



SAPIENZA
UNIVERSITÀ DI ROMA

Analysis and trigger approaches for New Physics searches exploiting Machine Learning at the ATLAS detector

Scuola di Scienze Matematiche Fisiche e Naturali
Scuola Dottorale Vito Volterra
Dottorato di Ricerca in Fisica (XXXVII cycle)

Graziella Russo

ID number 1743704

Advisor

Prof. Stefano Giagu

Co-Advisor

Dr. Valerio Ippolito

Academic Year 2024-2025

Analysis and trigger approaches for New Physics searches exploiting Machine Learning at the ATLAS detector

Sapienza University of Rome

© 2025 Graziella Russo. All rights reserved

This thesis has been typeset by L^AT_EX and the Sapthesis class.

Author's email: graziella.russo@uniroma1.it

*To Zorro IV,
which taught me how to be resilient*

Abstract

This thesis explores the application of Machine Learning (ML) techniques in two key areas of the ATLAS experiment: model-independent searches for New Physics and real-time triggering for the Muon Spectrometer upgrade. A Transformer Anomaly Detection (AD) model is developed to identify unexpected resonances in fully hadronic final states. The model is trained exclusively on QCD dijet background events and employs a feature transformation technique to ensure jet mass independence. The approach is validated on various Beyond the Standard Model benchmark signals and is integrated into an analytical fit over the background invariant mass spectrum for measuring upper limits on the signal production cross-sections. In parallel, ML methods are explored for the upgraded ATLAS Level 0 muon trigger system. Both Convolutional Neural Networks and Graph Neural Networks are studied for real-time muon reconstruction in the Resistive Plate Chambers trigger, with Quantization Aware Training and Knowledge Distillation applied to meet FPGA constraints.

These results demonstrate the potential of ML in enhancing both NP searches and real-time triggering, contributing to the optimization of the ATLAS experiment for the High-Luminosity LHC era.

Contents

Introduction	ix
1 To the Standard Model... and Beyond!	1
1.1 The Standard Model	1
1.1.1 Particles	2
1.1.2 Quantum Electrodynamics	3
1.1.3 Quantum Chromodynamics	3
1.1.4 Electroweak model	4
1.1.5 The Brout-Englert-Higgs mechanism	5
1.1.6 pp collisions and Pile-Up	7
1.2 Evidences beyond the Standard Model	8
1.3 What to expect	11
1.4 Theoretical models for New Physics candidates	13
1.4.1 Heavy Vector Triplets	13
1.4.2 Dark Jets	14
2 The ATLAS detector	17
2.1 CERN	17
2.1.1 The LHC injection chain	17
2.1.2 The LHC ring design	18
2.1.3 Beam parameters	19
2.1.4 Bunch filling scheme, LHC cycle and Run schedule	20
2.2 The ATLAS detector	22
2.2.1 Coordinate system	22
2.2.2 Particle Identification	24
2.2.3 Magnet system	25
2.2.4 Tracking	26

2.2.5	Calorimetry	28
2.2.6	Muon system	31
2.2.7	Trigger and Data Acquisition system	33
3	Object Reconstruction	37
3.1	ATLAS data preparation	37
3.2	Tracks and vertices	38
3.2.1	Track reconstruction	39
3.2.2	Primary vertex identification	40
3.3	Electrons and photons	40
3.3.1	EM topoclusters	40
3.3.2	Electron track and conversion vertex reconstruction	41
3.3.3	Superclusters	41
3.4	Jets	42
3.4.1	Jet algorithms	42
3.4.2	Jet grooming and PU mitigation algorithms	43
3.4.3	Jet reconstruction	44
3.4.4	Jet calibrations	48
3.4.5	Jet substructure	50
3.5	Taus	51
3.6	Muons	52
3.7	Missing Transverse Energy	53
4	Machine Learning	55
4.1	History and definitions	55
4.2	Machine Learning categorisation	56
4.3	Deep Neural Networks	57
4.3.1	Activation functions	58
4.3.2	Architecture of Deep Neural Networks	59
4.4	The training process	60
4.4.1	Data preparation	60
4.4.2	Splitting the dataset	61
4.4.3	Overtraining and underfitting	61
4.4.4	Training algorithms	62
4.4.5	Performance estimators	65

4.5	Machine Learning and High Energy Physics	67
5	Anomaly Detection R&D project	69
5.1	Anomaly Detection	69
5.1.1	Anomaly Detection for model-independent NP searches . . .	70
5.2	The LHC Olympics 2020 dataset	72
5.3	Dataset preprocessing	74
5.3.1	The jet transformation	75
5.3.2	Graph and vectorial datasets creation	78
5.4	Model architectures	84
5.4.1	Decision Trees, Random Forests and Boosted Decision Trees	85
5.4.2	Transformers	86
5.4.3	AutoEncoders	87
5.4.4	Graph Neural Networks	90
5.5	Training of the models	95
5.5.1	Classification trainings	95
5.5.2	Anomaly Detection trainings	95
5.5.3	Configuration of common hyperparameters	99
5.6	Results	99
5.6.1	GNN4AD	100
5.6.2	Classification performance	103
5.6.3	Anomaly Detection performance	105
5.6.4	Effects of the transformation	107
5.6.5	Anomaly Detection performance on LHC Olympics 2020 Black Boxes	110
5.6.6	Jet- and event-level Anomaly Score	112
5.6.7	Summary of the R&D results	114
6	Anomaly Detection for new resonances search within the ATLAS detector	117
6.1	Data and MC samples	118
6.1.1	Event preselection	118
6.1.2	Trigger strategy and event selection	120
6.2	ML trainings on MC dataset	128
6.2.1	Dataset strategies	128
6.2.2	Dataset creation	130

6.2.3	Transformer performance	135
6.2.4	Event-level Anomaly Score	139
6.3	Results for the MC analysis	142
6.3.1	Limits on signal production cross-sections	144
6.3.2	Towards a parametrisation of the observed QCD spectrum . .	154
6.4	Perspectives on the ATLAS analysis	158
7	New algorithms for the L0 barrel Muon Spectrometer trigger	161
7.1	ATLAS Phase II Upgrade	161
7.1.1	HL-LHC plan	161
7.1.2	ATLAS sub-detectors upgrade	163
7.1.3	The BI chambers	163
7.2	Level 0 RPC TDAQ system	166
7.2.1	Data Collector and Transmitter boards	166
7.2.2	Sector Logic boards	166
7.3	Level 0 RPC trigger algorithms	168
7.3.1	The PatFinder algorithm	168
7.3.2	The Patter-matching algorithm	169
7.3.3	Advantages and disadvantages of traditional approaches . . .	170
7.4	ML-based L0 RPC trigger algorithms	171
7.4.1	Convolutional Neural Networks	172
7.4.2	Experimental requirements	173
7.5	Dataset for the ML trigger algorithm	174
7.5.1	Noise modelling in the dataset	175
7.5.2	Images dataset creation	178
7.5.3	Graph dataset creation	178
7.6	Single and multi-muon reconstruction with CNNs	180
7.6.1	Compression techniques	180
7.6.2	CNN models	181
7.7	Single muon CNN performance	183
7.8	Multi-muon CNNs performance	185
7.9	GNN for single muon track identification	187
7.10	Perspectives	192
8	Conclusion	193

A Studies on RPCs timing calibration, detector and trigger efficiencies	197
A.1 Trigger timing calibration	197
A.2 L1 muon barrel trigger efficiencies	200
B Details on the jet transformation	203
B.1 The original procedure	203
B.2 The implemented procedure	204
List of Acronyms	207

Introduction

This thesis explores the use of advanced Machine Learning (ML) techniques in two distinct yet complementary areas within the ATLAS experiment: model-agnostic searches for Beyond the Standard Model (BSM) resonances and real-time event selection for the Phase II trigger upgrade. Both applications leverage modern deep learning architectures to address unique challenges, such as data-driven Anomaly Detection (AD) for discovering New Physics (NP) and fast, resource efficient ML models for real-time event selection in high-luminosity conditions.

Particle Physics is founded on the Standard Model (SM), a well-established theoretical framework that describes the fundamental particles and their interactions, excluding gravity. Over the second half of the 20th century, a series of discoveries provided experimental confirmation of the SM, culminating in 2012 in the detection of the Higgs boson, the final missing piece of the SM framework. Despite its remarkable success in describing fundamental particles and their interactions, the SM is not a complete theory. Several experimental observations strongly suggest the existence of NP beyond the Standard Model (BSM). Traditional searches for NP conducted by the ATLAS and CMS collaborations have focused on specific theoretical models, exploring well-defined regions of the parameter space and setting upper limits on production cross-sections. However, the absence of direct experimental evidence, combined with limited theoretical guidance, has motivated a shift towards more model-independent approaches in recent years. In this context, fully unsupervised Machine Learning (ML) has emerged as a promising tool for NP detection, enabling efficient background rejection and signal identification while maintaining independence from specific theoretical assumptions.

This work presents a model-agnostic search for BSM resonances decaying hadronically within the ATLAS detector, leveraging low-level information from the jet constituents. To ensure independence from any theoretical model, the search employs a fully unsupervised AD technique, trained and validated exclusively on background events. The AD models use loss functions that do not require prior knowledge of the event classification, allowing them to learn underlying structure of background processes and identify deviations as potentially associated with NP.

I am one of the two analysers of this search and I personally developed and optimised several of the ML architectures studied in this thesis, including the Transformer AD, the AutoEncoders and multiple Graph Neural Networks (GNN) architectures. To systematically compare different GNN layers, I designed and implemented the GNN4AD framework, enabling a structured and efficient evaluation of the best

graph approach.

A key component of this search is a feature transformation applied to the dataset, which I personally studied, removing dependencies on jet mass and enforcing invariance under translations and rotations. To develop and validate this strategy, I first tested it in a controlled environment, represented by the LHC Olympics 2020 research and development (R&D) dataset. This step allows for a thorough evaluation of models performance, as well as the identification of potential limitations and vulnerabilities in the architectures. A critical aspect of this work was the transition from jet-level to event-level Anomaly Score (AS). I played a key role in optimising this process, ensuring that the event-level AS preserves the underlying anomalies detected at the jet level while maximising discrimination power. Some of the results obtained in this first part of the project have been published in [1].

Once a robust configuration is established, I played a key role in extending the AD approach to a real ATLAS analysis. A dedicated data format has been developed to include detailed constituent information and I contributed to the main analysis framework and ntuple production. Furthermore, a tailored trigger strategy and event selection have been designed to optimise signal sensitivity. The AD technique is then tested on ATLAS official Monte Carlo (MC) samples, which simulate the ATLAS detector under Run 3 conditions. The full analysis workflow has been conducted using MC simulations and a training strategy that I designed. Although this study does not yet account for all experimental uncertainties, I integrated all components of a full data-driven search pipeline, concluding with a parametrisation fit for conducting a Bump Hunt search to assess the sensitivity to potential NP signals.

Beyond AD, ML methods can also enhance real-time event selection in the ATLAS trigger system, which will undergo a major upgrade for the High-Luminosity LHC (HL-LHC). By the end of Run 3, the LHC will increase its instantaneous luminosity to $5\text{--}7.5 \times 10^{34} \text{ cm}^{-2} \text{ s}^{-1}$, significantly raising the Pile-Up (PU) conditions and challenging the detector's capability to efficiently select interesting events. To cope with these conditions, the ATLAS Muon Spectrometer barrel region will be upgraded with an additional Resistive Plate Chamber (RPC) triplet layer, positioned in the innermost part of the detector. Additionally, all electronics will be replaced and the Level 0 Phase II trigger logic will be fully implemented on Field Programmable Gate Arrays (FPGA). This upgrade provides a unique opportunity to develop innovative ML algorithms capable of detecting muons that do not originate from the Interaction Point, addressing a limitation of the current trigger system. In fact, NP may be also hidden in events discarded by the trigger selection, as trigger algorithms are typically optimised for SM particles.

In this thesis, novel ML approaches based on Convolutional Neural Networks (CNNs) and GNNs are explored to enhance muon trigger efficiency. I personally developed the image dataset and investigated the incorporation of noise contribution to the muon samples to ensure a realistic training environment. Additionally, I extended the CNN model to achieve full single muon identification and contributed to multi-muon reconstruction.

However, the implementation of the Level 0 RPC trigger algorithm is subject to strict constraints on latency (approximately 390 ns), FPGA resource allocations

and overall efficiency, necessitating the use of compression techniques to meet experimental requirements. By using simultaneously compression techniques such as Quantization Aware Training and Knowledge Distillation, it was possible to satisfy all the requirements for the single muon reconstruction. However, transitioning to the multi-muon case posed additional challenges, as the same compression techniques proved insufficient. To address this, a new approach leveraging graph representations and GNNs was developed, a direction in which I played a key role.

My first year as a PhD student coincided with the start of ATLAS Run 3. During this period, a timing calibration was required for the RPC sub-detector and I produced the first timing distributions for its validation. However, despite timing resynchronisation and High Voltage recovery procedures, many sectors remained non-operational, resulting in a lower overall trigger efficiency compared to Run 2 benchmarks. To address this issue, I contributed to the automating of the generation of detector and trigger efficiency plots, which played a crucial role in developing and monitoring a recovery strategy that successfully mitigated efficiency losses.

This thesis is organised as follows. An introduction to the SM theory is briefly described in Chap. 1, alongside with the specific BSM theoretical models used for the search. Chap. 2 focuses on the ATLAS detector and the LHC accelerator complex. Particles are identified in the ATLAS detector following well-defined reconstruction algorithm, covered in Chap. 3. ML models and the description of the training process are presented in Chap. 4. Chap. 5 introduces the ATLAS analysis, providing an extensive overview of the AD architectures developed. The networks are tested on the public LHC Olympics 2020 R&D dataset as preliminary benchmark, to assess the general validity of the proposed approach in a controlled environment. The ATLAS Run 3 AD search for BSM resonances decaying fully hadronically is presented in Chap. 6. Given the novelty of the approach, the analysis required several validations and dedicated studies and it is not yet terminated. However, a full study on Monte Carlo samples is outlined, showing the entire analysis pipeline and expected results. Finally, Chap. 7 covers the future ATLAS upgrades for Phase II, focusing on the barrel of the Muon Spectrometer. New ML algorithms developed for muon track reconstruction are described. App. A provides a chronological overview of the first year of Run 3 data-taking from the RPC point of view.

Chapter 1

To the Standard Model... and Beyond!

Particle Physics, or High Energy Physics (HEP), is the field of Physics which studies elementary particles and their behaviour under three of the four known fundamental interactions: electromagnetic (EM), Weak and Strong interactions¹. The Standard Model (SM) is an umbrella term which refers to many different theories developed from the second half of the 20th century to mid '70s and represents the state of the art of human understanding of Particle Physics. In the years, the SM has predicted many new particles that were actually observed at collider experiments, the last of which is the Higgs boson, discovered at CERN in 2012 [2, 3]. Despite the impressive accuracy in the predictions validated by experimental evidences, there are phenomena that the SM is not able to justify yet, paving the way to extensions to overcome its limitations.

1.1 The Standard Model

The SM of Particle Physics is a relativistic Quantum Field Theory (QFT) in which each particle is represented by a quantum field $\psi(x)$ and its dynamics is described in terms of Lagrangian densities $\mathcal{L}(\psi, \partial_\mu \psi)$ [4, 5]. The theory is constructed to be invariant under the Poincaré group, which encompasses Lorentz transformations and translations in Minkowski space, and under the local $SU(3)_C \otimes SU(2)_W \otimes U(1)_Y$ gauge symmetry. The $SU(3)_C$ non-abelian group is associated with strong interactions, while the $SU(2)_W \otimes U(1)_Y$ group describes Electroweak (EW) interactions.

¹Many attempts of merging all the four interactions, including also gravity, have been made, but none has been validated yet by experimental observations. However, at collider energy scale the gravitational interaction can be considered negligible.

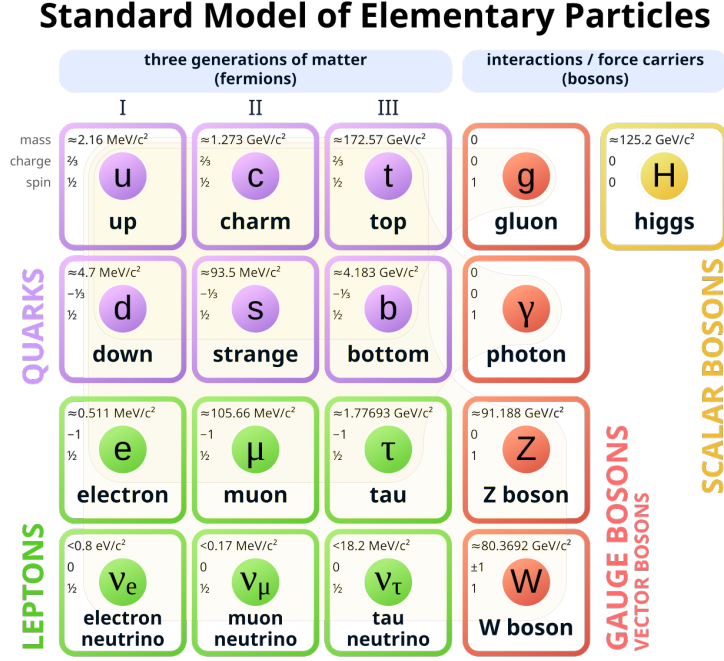


Figure 1.1. The SM of elementary particles. Figure from Ref. [6].

1.1.1 Particles

The SM fundamental components, the elementary particles, can be grouped into *fermions*, with half-integer spins and obeying Fermi-Dirac statistics, and *bosons*, with integer spins and adhering to Bose-Einstein statistics.

Fermion particles are also called *matter* particles and are organised in three generations of increasing proper mass. Each generation consists of two leptons, one electrically charged and one neutral, and two quarks, as represented by the columns in Fig 1.1.

There are three electrically charged leptons, one for each *lepton flavour*: the electron e , the muon μ and the tau τ . Additionally, there are three neutral leptons referred to as the electron neutrino ν_e , muon neutrino ν_μ and tau neutrino ν_τ . In the SM, leptons have their own anti-particles², having opposite quantum numbers but the same mass, as described in Quantum Mechanics (QM). All leptons interact weakly, while only charged leptons can interact electromagnetically.

Quarks exist in six different flavours, up u , charm c , top t , down d , strange s and bottom b . The first three quarks, the so-called *up-type* quarks, have a $\frac{2}{3}e$ electric charge; the last three, the *down-type* quarks, have a $-\frac{1}{3}e$ electric charge. As for the leptons, quarks have their own anti-particles. Quarks can interact electromagnetically, as well as through weak and strong interactions. In fact, quarks are also charged in *colour*, that is the charge of strong interactions described in the Quantum

²Actually, it has not been validated whether neutrinos have distinct anti-particles or are Majorana fermions, but from a theoretical point of view there is nothing that prevents neutrinos to have their anti-particle counterparts.

Chromodynamics (QCD) theory. Due to colour confinement, which will be better described in Sec. 1.1.3, quarks and colour charged particles cannot be observed in nature. Indeed, quarks can manifest only as composite colour-neutral states, the *hadrons*. Hadrons consisting of one quark and one anti-quark are called *mesons*, hadrons with three quarks are called *baryons*.

The bosons present in the SM of elementary particles have either spin $s = 0$ or $s = 1$. Spin-1 bosons, that correspond to vector fields, are also called *gauge bosons*. They are the mediators of three of the four known interactions: the massless photon γ is the mediator of the electromagnetic interaction, the 8 massless gluons g mediate the strong interaction and the Z and $W^\pm$ bosons are the weak mediators. The only spin-0 boson of the SM is the Higgs boson H which is responsible of giving mass to all the other elementary massive particles through the BEH mechanism, described in Sec. 1.1.5.

1.1.2 Quantum Electrodynamics

Quantum Electrodynamics (QED) is the relativistic QFT that describes electromagnetic interactions between fermions (with mass m , represented by the field ψ) and the electromagnetic vector field A_μ of the photon [7]. Starting from the Lorentz-invariant Lagrangian density of a free fermion, the dynamics is completely defined by:

$$\mathcal{L}_{\text{QED}} = \bar{\psi}(i\gamma^\mu D_\mu - m)\psi - \frac{1}{4}F^{\mu\nu}F_{\mu\nu} \quad (1.1)$$

Here, $F_{\mu\nu} = \partial_\mu A_\nu - \partial_\nu A_\mu$ is the electromagnetic tensor; $D_\mu = \partial_\mu - ieA_\mu$ is the covariant derivative; γ^μ are the Dirac matrices, while $\bar{\psi} = \psi^\dagger \gamma^0$ is the adjoint of the ψ field. The Dirac matrices are derived from Pauli matrices σ_i (with $i = 1, 2, 3$) as:

$$\gamma^0 = \begin{pmatrix} 0 & \mathbb{I} \\ \mathbb{I} & 0 \end{pmatrix} \quad \gamma^i = \begin{pmatrix} 0 & \sigma_i \\ -\sigma_i & 0 \end{pmatrix} \quad (1.2)$$

Since the Lagrangian in Eq. 1.2 is invariant under the local transformation of the $U(1)$ symmetry group, Noether's theorem [8] implies the existence of a conserved current, $J^\mu = \bar{\psi}\gamma^\mu\psi$, which corresponds to the conserved electric charge.

1.1.3 Quantum Chromodynamics

QCD is a non-abelian gauge theory based on the $SU(3)$ group representation [9, 10]. It describes the strong interaction between quarks, represented as fermionic fields $q(x)$, and 8 gluons, represented as 8 massless gauge fields $G_{\mu\nu}^a$ with $a = 1, \dots, 8$. The conserved charge of the theory is the colour, which can assume three values; unlike QED, in QCD the mediators of the interaction, the gluons, are charged in colour and therefore can interact with each other. Under a transformation of the $SU(3)_C$ group, the quark field changes as:

$$q(x) \longrightarrow q'(x) = e^{-ig_s T_a} q(x) \quad (1.3)$$

where g_s is the gauge coupling constant of strong interactions and T_a are the 8 generators of $SU(3)$ in the fundamental representation. The covariant derivative of

the theory is therefore defined as:

$$D_\mu = \partial_\mu + ig_s T_a G_\mu^a \quad (1.4)$$

The generators $T_a = \lambda_a/2$ are constructed from the Gell-Mann matrices λ_a [11] and follow the non-commutative rule $[T_a, T_b] = if_{abc}T_c$, where f_{abc} are the structure constants of $SU(3)$. The QCD tensor field is:

$$F_{\mu\nu}^a = \partial_\mu G_\nu^a - \partial_\nu G_\mu^a - g_s f^{abc} G_\mu^b G_\nu^c \quad (1.5)$$

The QCD Lagrangian density can be written as:

$$\mathcal{L}_{\text{QCD}} = \bar{q}(i\gamma^\mu D_\mu - m)q - \frac{1}{4}F^{a,\mu\nu}F_{\mu\nu}^a \quad (1.6)$$

The strong coupling constant $\alpha_s = \frac{g_s^2}{4\pi}$ is the only fundamental parameter of QCD. In perturbative calculations, observable predictions are expressed in terms of the renormalised coupling $\alpha_s(\mu_R^2)$, which depends on the renormalisation scale μ_R under consideration. The strong coupling constant must follow the Renormalisation Group Equation:

$$\mu_R^2 \frac{d\alpha_s}{d\mu_R^2} = \beta(\alpha_s) = -\alpha_s^2 \sum_{i=0}^{\infty} b_i \alpha_s^i \quad (1.7)$$

The first parameter b_i of the beta function is expressed as $b_i = -\frac{33-2n_f}{12\pi}$, where n_f are the number of flavours, and it is responsible for the *asymptotic freedom*. In fact, the strong coupling becomes smaller hence weaker for processes involving large momentum transfers, the so-called hard scattering processes. At low energies, instead, the coupling constant increases and leads to the formation of hadrons.

1.1.4 Electroweak model

The EW model is a gauge theory based on the $SU(2) \otimes U(1)$ symmetry group and seeks to unify electromagnetic and weak interactions [12, 13, 14]. The gauge bosons of the model are W_μ^i , with $i = 1, 2, 3$, for $SU(2)$ and B_μ for $U(1)$, while the corresponding coupling constants are g and g' . The fermions that interact with the gauge bosons are chirality eigenstates and therefore can be classified as *left-handed* and *right-handed* fermions:

$$\begin{aligned} \psi_L &= \frac{1}{2}(1 - \gamma^5)\psi \\ \psi_R &= \frac{1}{2}(1 + \gamma^5)\psi \end{aligned} \quad (1.8)$$

where $\gamma^5 = i\gamma^0\gamma^1\gamma^2\gamma^3$. In the EW model the fermions involved can be either leptons or quarks belonging to one of the three generations ($i = 1, 2, 3$):

$$L_i = \left\{ \begin{pmatrix} \nu_i \\ l_i \end{pmatrix}_L \right\}, \quad e_i = \{(l)_R\} \quad (1.9)$$

for the leptons and

$$Q_i = \left\{ \begin{pmatrix} u_i \\ d'_i \end{pmatrix}_L \right\}, \quad u_i = \{(u)_R\}, \quad d'_i = \{(d')_R\} \quad (1.10)$$

for the quarks. Note that right-handed neutrinos are not involved in electroweak interactions and that the down quarks in Eq. 1.10 are linked to the ones presented in Sec. 1.1.1 by $d'_i \equiv \sum_j V_{ij} d_j$, with V_{ij} being the Cabibbo-Kobayashi-Maskawa mixing matrix [15, 16].

The generators of the $SU(2)$ symmetry group are the weak isospin $\vec{\tau} = \frac{1}{2}\vec{\sigma}$ where $\vec{\sigma}$ are Pauli matrices; the generator of the $U(1)$ symmetry group is the hypercharge Y . Left-handed fermion fields transform like doublets under $SU(2)$, while right-handed fermions transform like singlets:

$$\begin{aligned} \psi_L(x) &\longrightarrow \psi'_L(x) = e^{i\vec{\alpha}(x) \cdot \vec{\tau} + i\beta(x)Y} \psi_L(x), \\ \psi_R(x) &\longrightarrow \psi'_R(x) = e^{i\beta(x)Y} \psi_R(x) \end{aligned} \quad (1.11)$$

It is possible now to introduce the EW covariant derivative as:

$$D_\mu = \partial_\mu + ig' \frac{Y}{2} B_\mu(x) + ig \frac{\tau^a}{2} W_\mu^a(x) \quad (1.12)$$

The observable vector fields, the photon, the $W^\pm$ the Z bosons, are obtained by linear combinations of the gauge bosons W_μ^i and B_μ of the EW model:

$$\begin{aligned} A_\mu &= W_\mu^3 \sin(\theta_W) + B_\mu \cos(\theta_W), \\ Z_\mu &= W_\mu^3 \cos(\theta_W) + B_\mu \sin(\theta_W), \\ W_\mu^\pm &= \frac{1}{\sqrt{2}} (W_\mu^1 \mp W_\mu^2) \end{aligned} \quad (1.13)$$

where θ_W is the Weinberg angle [17] that is linked to the electric charge e and the coupling constants g and g' through the formula $e = g \sin(\theta_W) = g' \cos(\theta_W)$. The Lagrangian density of EW interactions is expressed as:

$$\mathcal{L}_{EW} = -\frac{1}{4} W^{a,\mu\nu} W_{\mu\nu}^a - \frac{1}{4} B^{\mu\nu} B_{\mu\nu} + i\bar{\psi} \gamma^\mu D_\mu \psi \quad (1.14)$$

where $W_{\mu\nu}^a = \partial_\mu W_\nu^a - \partial_\nu W_\mu^a - g\epsilon^{abc} W_\mu^b W_\nu^c$ and $B_{\mu\nu} = \partial_\mu B_\nu - \partial_\nu B_\mu$. Despite being incredibly accurate in describing experimental observations, the EW model's Lagrangian lacks explicit terms for the mass of the $W^\pm$ and Z bosons, as well as fermions. In 1964 Higgs proposed a mechanism [18], first theorised by Englert and Brout to describe ferromagnetic spin waves [19, 20] to address the problem of the origin of the mass of elementary particles.

1.1.5 The Brout-Englert-Higgs mechanism

The Brout-Englert-Higgs (BEH) mechanism involves the spontaneous symmetry breaking of $SU(2)_W \otimes U(1)_Y$ group into $U(1)_{EM}$ group, through the existence of a

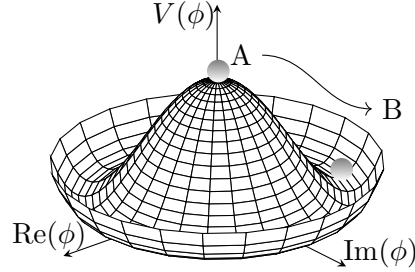
complex scalar boson, the Higgs boson [21]. Let's consider a complex scalar doublet ϕ of isospin $Y(\phi) = +\frac{1}{2}$:

$$\phi = \begin{pmatrix} \phi^+ \\ \phi^0 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} \phi_1 + i\phi_2 \\ \phi_3 + i\phi_4 \end{pmatrix} \quad (1.15)$$

The potential associated to the scalar field $V(\phi)$ appears in the form of:

$$V(\phi) = -\mu^2 \phi^\dagger \phi + \frac{\lambda}{2} (\phi^\dagger \phi)^2 \quad (1.16)$$

where μ^2 and λ are constants. The λ term of the potential must be positive to ensure the existence of an equilibrium state. When the term μ^2 is less than 0, the potential takes the shape of a mexican hat, with the scalar field free to assume any of the infinite minima that satisfy the condition $|\phi_1|^2 + |\phi_2|^2 + |\phi_3|^2 + |\phi_4|^2 = \mu^2/\lambda$.



When μ^2 is greater than 0, the field acquires a vacuum expectation value (VEV) v that spontaneously breaks the global $U(1)_Y$ gauge symmetry:

$$\langle \phi \rangle = \sqrt{\frac{\mu^2}{2\lambda}} \equiv \frac{v}{\sqrt{2}} \quad (1.17)$$

It is convenient to apply a gauge transformation to the Higgs field so that it can be written as:

$$\phi = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ v + h(x) \end{pmatrix} \quad (1.18)$$

where $h(x)$ is the Higgs boson scalar field. By including the Higgs scalar field, the EW SM Lagrangian density becomes:

$$\begin{aligned} \mathcal{L}_{\text{SM}} = & -\frac{1}{4} W^{a,\mu\nu} W_{\mu\nu}^a - \frac{1}{4} B^{\mu\nu} B_{\mu\nu} + i\bar{\psi}\gamma^\mu D_\mu \psi + \\ & -Y_{ij}^l \bar{L}_i \phi e_j - Y_{ij}^\nu \bar{e}_i \bar{\phi} L_j - Y_{ij}^d \bar{Q}_i \phi d'_j - Y_{ij}^u \bar{Q}_i \phi u_j + h.c. \\ & + (D^\mu \phi)^\dagger (D_\mu \phi) - V(\phi) \end{aligned} \quad (1.19)$$

where D_μ is the one expressed in Eq. 1.12 and the Y_{ij} are matrices describing the so called Yukawa interaction ($i, j = 1, 2, 3$ represent the three fermion generations). The first line of Eq. 1.19 represents the $SU(2)_W$ and $U(1)_Y$ gauge bosons and their interactions with fermionic particles. The second line is the Yukawa term, in which the Higgs boson field couples with the fermions; these terms are responsible for the fermions having mass. The last line describes the interaction of the Higgs boson with the W_μ^a and B_μ vector fields, the mass of the Higgs boson and its self-coupling.

1.1.6 pp collisions and Pile-Up

The SM is validated at collider experiments, such as the Large Hadron Collider (LHC) at CERN, described in Sec. 2.1.1. In particular, at LHC proton bunches collide at energies of the order of tens of TeV and the underlying theory of proton-proton, or pp , collisions is QCD [22, 23]. Depending on the energy that takes part in the interaction, it is possible to distinguish among *hard* and *soft* high-energy scattering. Perturbation theory describes processes resulting from hard scattering with good precision, but fails in predicting properties of soft processes. The calculation of hadronic cross section is, however, possible thanks to the QCD factorization theorem, through the convolution of two parts: the short-distance parton cross section and universal long-distance Parton Distribution Functions (PDF). PDFs encode the probability of finding a parton (quark or gluon) carrying a given fraction of the proton's momentum at a specific energy scale. Since these functions cannot be calculated from first principles, they are extracted from experimental data and evolved using the Dokshitzer-Gribov-Lipatov-Altarelli-Parisi equations [24]. Given the complexity of the problem, algorithms such as PYTHIA [25], HERWIG [26], POWHEG [27] and SHERPA are used to simulate the scatterings. Once the collision has been simulated, additional steps are required, especially if composite particles are present in the final state. In particular, the hadronisation (Fig. 1.2) is the process through which coloured partons, such as quarks and gluons, transform into colour-neutral hadrons due to colour confinement. This non-perturbative phenomenon is not yet understood from first principles and is instead modelled using phenomenological approaches. Nonetheless, popular theoretical models, such as *cluster fragmentation model* and *string fragmentation model* implemented in HERWIG and PYTHIA, respectively, are used to obtain reliable theory predictions.

In hadron-hadron collisions, all the processes not resulting directly from the hard scattering are referred as *underlying events*. The variety of processes from the underlying event may arise from hard scattering with initial- and final-state radiation, from hard process and second scattering, with or without initial- or final-state radiation, and so on. All the parasitic collisions originated either by the underlying events or by soft scattering of other partons among the protons in the beams are known as Pile-Up (PU). Physicists try to mitigate PU processes in the LHC experiments, as they interfere in the event reconstruction of the collision. A regular monitoring of the PU is therefore required; in Fig. 2.3b the average number of collisions not coming from the hard scattering, $\langle\mu\rangle$, is shown for the first three Run 3 years.

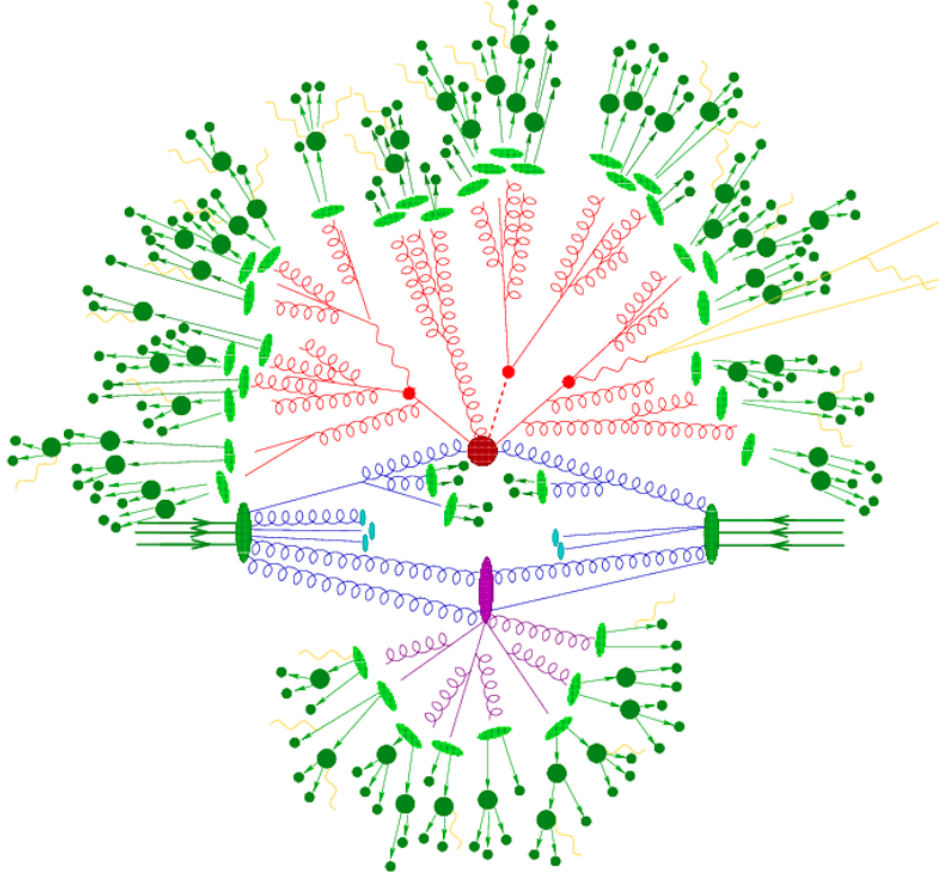


Figure 1.2. Visual representation of LHC pp collisions. The hard process is shown in red, the multiple partons interaction are represented in purple, the initial- and final-state radiation is depicted in blue and the hadronisation is in green. Figure from Ref. [28].

1.2 Evidences beyond the Standard Model

The SM is the theoretical model that provides the best description of the interactions between elementary particles, but there are still many unexplainable phenomena. In this section, I will provide a brief overview of the most significant open questions in the SM: the matter-antimatter asymmetry, the nature of Dark Matter, the neutrino mass and the strong CP problem.

Matter-antimatter asymmetry problem

It was clear from the very beginning of the SM formulation that there is an imbalance between matter and antimatter in the observable Universe. Although it may seem purely a cosmological problem, cosmologists start from the assumption that the theoretical models that describe the Universe *now* (thus the SM) are also able to describe the Universe *at its origin*. It is largely accepted that in the early Universe particles and antiparticles were present in exactly the same number, as requested from the SM, and when the Universe cooled down the annihilation processes between

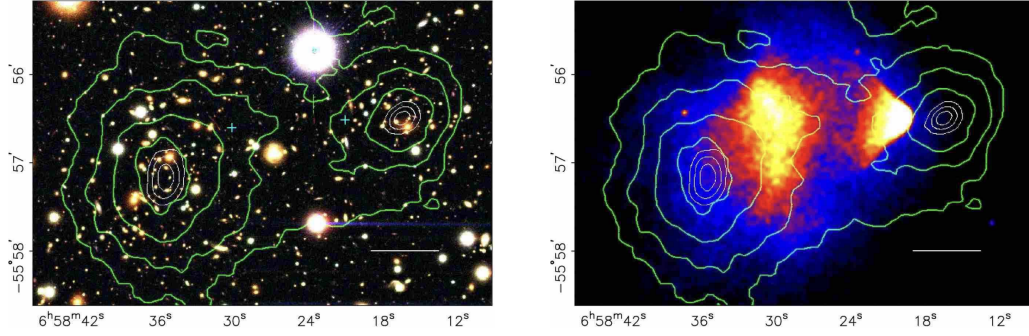


Figure 1.3. The two images show the same 1E0657-558 galaxy cluster in the visible range (on the left) and through x-ray detection (on the right). The green contours represent the profile of the gravitational potential obtained by measuring gravitational weak lensing effects. Fig. from Ref. [39].

matter and antimatter should have left the Universe completely empty [29]. During the cooling period, some physical processes contributed to a small imbalance in favour of baryonic matter. In 1967 Sakharov proposed three necessary conditions for such processes [30], which suggest that a possible explanation can come from the violation of the conjugation-parity (CP) symmetry, but there is still no evidence that proves this theory.

Nature of Dark Matter

The Universe is composed of 4.9% of ordinary matter, for 26.4% of Dark Matter (DM) and for 68.7% of Dark Energy [31]. DM is a type of matter that interacts with ordinary SM matter through gravitational interactions, without evidence of other interactions. There are many evidences of existence of DM, all coming from astronomical observations. In 1931 Hubble and Humason first measured the rotation curves of the spiral galaxies in the COMA clusters from the Doppler shift of atomic lines emitted by the stars [32]. Two years later, Zwicky using the virial theorem evaluated the expected velocity dispersion of the galaxies and noted a large discrepancy suggesting the presence of much more matter than that observed [33]. After that, many other measurements of galaxies rotation curves have corroborated the presence of DM [34, 35, 36]. Moreover, it was calculated in the '70s that DM is distributed in dark spherical halo in spiral galaxies, to ensure the stability of the disk, and this is actually observed in galaxy-galaxy gravitational lensing [37, 38]. This phenomenon occurs when an EM source is positioned behind a gravitationally interacting mass, which is not visible through the other interactions. The mass acts as an optical lens, completely or partially distorting and magnifying the image of the source. One of the best evidences of the existence of DM is the 1E0657-558 cluster, also known as Bullet Cluster (Fig. 1.3), which resulted from the collision of two clusters around 150 million years ago [39]. Through independent measurements of visible matter and gravitational effects of DM, it was possible to observe that the two types of matter are spatially separated in the cluster, suggesting a small self-interaction cross-section of DM but also constraining the DM/ordinary matter

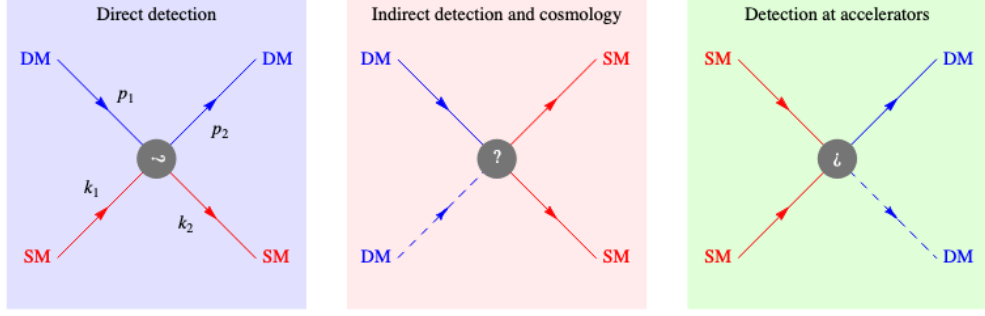


Figure 1.4. The three DM search strategies: direct, indirect detection and production at accelerators. Figure from Ref. [31].

cross-section. Another strong indication of DM comes from the observed large scale structure of the Universe. Today, the Universe appears highly inhomogeneous on scales smaller than its present horizon, but at the time of the Cosmic Microwave Background (CMB) radiation formation only tiny primordial density fluctuations were present, as shown from measurements provided by WMAP and Planck [40, 41]. By adding the presence of DM in the Universe’s evolution equations, it is possible to obtain the structures which constitute the Universe today.

Although the existence of DM is universally accepted within the scientific community³, its nature is still an unsolved puzzle in Particle and Astroparticle Physics. The main paradigms range from macroscopic objects, like Primordial Black Holes [44] or Massive Astrophysical Compact Halo Objects [45], to particle DM, like sterile neutrinos [46], axions [47], Weakly Interacting Massive Particles [48] or Supersymmetric particles [49, 50]. Among all these candidates there is not one that stands out as more promising with respect to the others and the last word is left, as always in physics, to experimental observations. There are three ways of DM detection: direct and indirect detection and production at the accelerators, as shown in Fig. 1.4. Direct detection searches for the recoil event of DM particles hitting one of the nuclei or electrons of the material of the detectors. Indirect detection searches for DM annihilations or decays in the Universe, by looking at the signatures in cosmic rays. At accelerators, the aim is to directly produce DM pairs and to detect their decay.

Neutrino mass

In the SM neutrinos are neutral, massless particles that interact only via weak interactions and appear only as left-handed chiral eigenstates. In 1957 Pontecorvo first proposed the possibility of neutrino oscillation [51], similar to $K^0 \rightleftharpoons \bar{K}^0$ oscillation. Since at that time neutrino flavour had not yet been discovered, Pontecorvo hypothesised the oscillation between right-handed (sterile) neutrinos that do not interact

³There are some Modified Newtonian dynamics [42] and Modified Gravity [43] theories that are able to explain the rotation curve of galaxies or the structure formation of the Universe without taking into account DM, but they struggle to adequately explain other phenomena that can be described accepting the DM hypothesis, thus they will not be treated in this thesis.

weakly. Ten years later in another paper [52] Pontecorvo proposed the flavour neutrino oscillation $\nu_\mu \rightleftharpoons \nu_e$ and pointed out that because of the oscillation the flux of solar ν_e should be reduced considerably. In the '70s and '80s the neutrino oscillation theory was further refined and many experiments in later years with solar, atmospheric and reactor neutrinos confirmed the oscillation phenomenon. Noting that there are no symmetry constraints in the SM that limit the neutrinos to have mass, the oscillation can be fully described as an interference phenomenon in which the unitary mixing matrix, also called Pontecorvo-Maki-Nakagawa-Sakata matrix [53], has the mass terms on the diagonal and at least one non-zero term out of the diagonal. Oscillations are sensitive to mass-squared differences, therefore the absolute value of neutrinos' mass is still unknown, as well as the hierarchy between the masses of the three neutrinos. In addition to these open questions, it is still not clear why neutrino should have mass, even if no first principle in the SM explicitly prevents neutrinos to have one.

Strong CP problem

The QCD theory, in principle, should allow for the violation of CP symmetry, as the Lagrangian density in Eq. 1.6 can be expanded to contain the term:

$$\mathcal{L}_{\text{QCD}} \supset \theta \frac{g_S^2}{32\pi^2} F_{\mu\nu}^a \tilde{F}^{a,\mu\nu} \quad (1.20)$$

where θ is the parameter controlling the strength of CP violation, $F_{\mu\nu}^a$ is the gluon field and $\tilde{F}^{a,\mu\nu}$ its dual. The θ parameter can naturally range between 0 and 2π , but the measurement of the electric dipole moment of the neutron [54] constraints its value at $\theta < 10^{-10}$. The fact that the θ parameter takes such small value appears unnatural, requiring extreme fine-tuning of the theory. Among several proposed solutions, the *axion* hypothesis introduced by Peccei and Quinn [55] is the most popular. The theory involves promoting θ to a dynamic field, the axion. Despite the numerous theories proposed, none has been experimentally verified, suggesting that our understanding of QCD may be incomplete.

1.3 What to expect

The SM is a very elegant theory and describes an incredible amount of processes with great precision; however, if there is a single phenomenon for which it fails, it means that there should be a more complete Beyond the Standard Model (BSM) theory for which the SM is a special case. The majority of the SM limitations comes from astrophysics or cosmology, while very few tensions have appeared at high-energy scales [56]. Nonetheless, New Physics (NP) must exist and should also be visible at collider experiments. When the Large Hadron Collider at CERN was approved in 1994, only few particles were missing from the SM puzzle (see Fig. 1.5) and it was believed that the experiments at the accelerator would complete the picture and give hints, if not even evidences, of NP in the following years. Unfortunately, apart from the Higgs boson, no NP particles have been discovered and

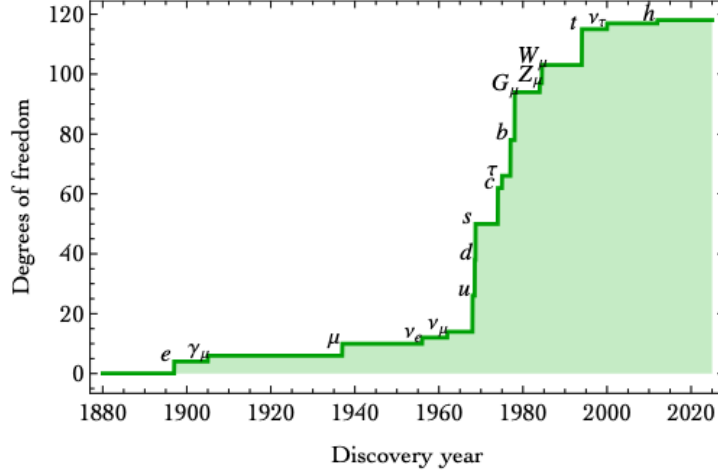


Figure 1.5. Discovery year of the elementary particles of SM. Figure from Ref. [31].

some theories popular in those years, like Supersymmetry, have not received any experimental confirmation. Physicists at collider experiments have always searched for hints of NP following two main directions: *precision measurements* in which small deviations from SM predictions are investigated and scrutinised, and *searches* which look for completely new processes that go beyond SM. Traditionally, searches for NP have focused on specific BSM hypotheses, analysing whether the data aligns with theoretical predictions. While this strategy has led to strong constraints on many models, no clear evidence of NP has been found so far. Given the lack of a clear theoretical guide for BSM physics, the most efficient approach is to adopt *model-independent*, or *model-agnostic*, strategies that can explore a broad phase space and a wide range of possible signatures. By relaxing the assumptions of traditional searches, these analyses maximize discovery potential and provide a more comprehensive exploration of what is still unknown.

Another possible reason why NP has not been observed in collider experiments may lie in the way data is collected. Given the extremely high collision rate, with hundreds of millions of interactions per second, it is impossible to store and process all events. Instead, a *trigger system* (see Sec. 2.2.7) is used to select and retain only a fraction of the data considered relevant for analysis. These trigger algorithms are optimised for SM processes, which means that NP signatures, if significantly different from known physics, could be partially or entirely rejected. As a result, certain NP processes might remain hidden within the collected data, making their detection statistically challenging.

The current LHC program for Run 3 and for the years to come consists of precision measurements, like the Higgs self-couplings⁴ and of more model independent searches, in addition to traditional theory-specific searches. Moreover, interest is rising in new trigger algorithms able to select “anomalous” events at the source that

⁴Many consider the Higgs self-couplings analyses as searches, but given the precision achieved in controlling the systematics and in Higgs tagging and being the Higgs boson a SM particle, for the sake of this thesis I will consider them as precision measurements.

can be later analysed with a dedicated approach [57].

1.4 Theoretical models for New Physics candidates

In this thesis, which focuses on searching for NP, some of the possible BSM theoretical models have been investigated, particularly Heavy Vector Triplets, which can explain new narrow resonances in the SM, and Dark Jets, which emerge from QCD-like scenarios in dark sectors.

1.4.1 Heavy Vector Triplets

General and complete theoretical BSM models, which are able to expand the SM and predict new phenomena that can be validated at collider experiments, are usually very complex and require a large number of parameters. Often, these parameters are so sensitive that subtle changes in their value produce large differences in the observables, making a full scan over the parameters not feasible. Therefore, Simplified Models [58] that involve only few new particles and interactions are becoming more popular. The idea behind is very simple: NP should manifest in the production of some new narrow resonances. Resonant searches typically reduce the set of parameters of the model, as they are not so sensitive to all the details of the underlying theoretical framework. Simplified Models are not supposed to be complete theories as they are defined to describe only on-shell productions and decays. By employing phenomenological Lagrangian densities, Simplified Models include only the essential terms, such as the mass of the resonance and the couplings governing its production and decay.

The simplest scenario is given by a Drell-Yan decay of an electroweak charged spin-1 boson, such as a Z' or W' , corresponding to the application of a Heavy Vector Triplet (HVT) to the SM [59]. Basically, a real vector V_μ^a ($a = 1, 2, 3$) in the adjoint representation of $SU(2)_W$ with vanishing hypercharge is added to the SM Lagrangian density. The charge eigenstates of the fields are combinations of the vector V_μ^a , as in the formula:

$$V_\mu^\pm = \frac{V_\mu^1 \mp iV_\mu^2}{\sqrt{2}}, \quad V_\mu^0 = V_\mu^3 \quad (1.21)$$

The phenomenological Lagrangian density becomes:

$$\begin{aligned} \mathcal{L}_V = & -\frac{1}{4}V^{a,\mu\nu}V_{\mu\nu}^a + \frac{m_V^2}{2}V^{a,\mu}V_\mu^a \\ & + ig_V c_H V^{a,\mu}[\phi^\dagger \tau^a D^\mu \phi - D^\mu \phi^\dagger \tau^a \phi] + \frac{g^2}{g_V} c_f V_\mu^a \bar{\psi} \gamma^\mu \tau^a \psi \\ & + \frac{g_V}{2} c_{VVV} \epsilon_{abc} V_\mu^a V_\nu^b D^\mu V^{\nu,c} + g_V^2 c_{VVHH} V_\mu^a V^{\mu a} \phi^\dagger \phi - \frac{g}{2} c_{VW} \epsilon_{abc} W^{a,\mu\nu} V_\mu^b V_\nu^c \end{aligned} \quad (1.22)$$

The first line of Eq. 1.22 contains the free-field dynamics of the HVT, with $V_{\mu\nu}^a = D_\mu V_\nu^a - D_\nu V_\mu^a$ and $D_\mu V_\nu^a = \partial_\mu V_\nu^a + g\epsilon^{abc}W_\mu^b V_\nu^c$, and the explicit mass term, where

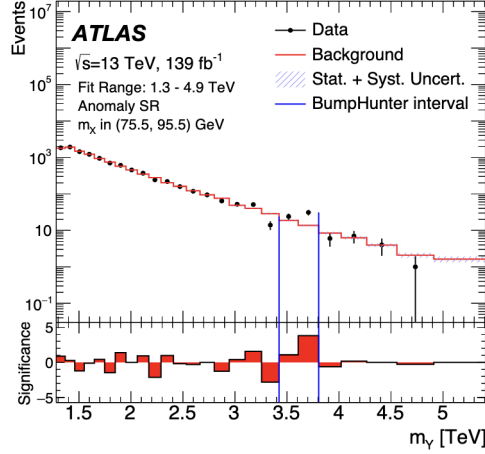


Figure 1.6. Invariant mass distribution of the Y boson from the HVT model in the ATLAS [60] analysis. The p-value in the ration plot shows that the excess in the [3608, 3805]GeV range has a global significance of 1.43σ .

the mass of the HVT is m_V . The second and third lines refer to the interaction of the HVT with coupling constant g_V and the SM fields: the Higgs boson ϕ , the $SU(2)_W$ generators W_μ^a and the fermions ψ . The constants are responsible for the interaction of the HVT with the Higgs boson c_H , with the fermions c_f , with itself trilinearly c_{VVV} and with the W_μ^a bosons c_{VW} . The newly introduced HVT can decay into di-lepton, di-quark or di-boson final states⁵. Experimental analyses that consider HVT models search for a resonant peak with a low mass tail in the invariant mass distribution. In case the HVT boson is decaying into di-quarks with quark mass $m_q \ll M_V$, the peak should follow a Breit-Wigner distribution with:

$$\Gamma(V_\pm \rightarrow q\bar{q}) \approx \frac{g_V^2 c_f^2}{24\pi} M_V \quad (1.23)$$

Several ATLAS and CMS Run 2 analyses have searched for HVT signatures, in particular in the fully hadronic final state [60, 61, 62]. In all the three approaches, events are selected if containing two large R jets (see Sec. 3.4.1) and passing some broad kinematic cuts. Jet tagging ML algorithms, such as Variational Recurrent Neural Networks (VRNNs), are employed to better discriminate between HVT and background mainly consisting of QCD dijets. The background estimation is performed with data-driven approaches. These searches have excluded at 95% of confidence level HVT signals in the mass range [1.3, 3.1] TeV and, additionally, the CMS analysis also excluded the [3.3, 3.5] TeV mass range.

1.4.2 Dark Jets

A compelling class of DM models involves dark sectors, which introduce new BSM interactions. Among these, dark QCD, which is a strongly coupled hidden sector

⁵HVT can decay also into $\gamma + W$ boson states, but with tiny branching ratio.

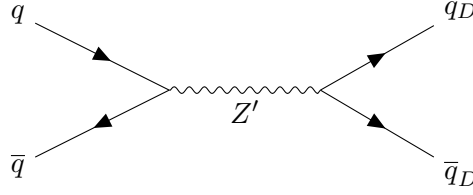


Figure 1.7. Feynman diagram of the pair production of dark quarks via the Z' portal in the s -channel.

with its own non-abelian gauge group, remains largely unexplored. These theories predict new matter fields, analogous to SM fermions and gauge bosons, which can lead to complex phenomena and potentially exotic signatures in collider experiments [63]. In this representation, new matter fields analogous to SM fermionic and gauge fields emerge. From a theoretical point of view and in the absence of experimental confirmations, there are no a priori limits to the group dimension N_d , thus the number of dark colours, or the number of flavours n_f of the matter fields involved. At energies below a certain confinement scale, denoted as Λ_D , the dark sector undergoes confinement, giving rise to composite particles known as dark hadrons. In order to be accessible at collider experiments, $\Lambda_D \ll \sqrt{s}$ where $\sqrt{s}$ is the centre of mass energy. Under this condition, dark quarks can be produced via a portal interaction, subsequently undergoing rapid hadronisation within the dark sector. Depending on the specific symmetries of the theory, some of these dark hadrons may be stable or may decay, at least partially and possibly with significant lifetimes, into SM particles.

The interaction between the dark sector and the SM is facilitated by a mediator, commonly known as the *portal*, enabling the production of dark sector states, such as dark quarks or dark hadrons. The portal can either be a SM particle, for example the hypercharge gauge boson $B_{\mu\nu}$ [64], the Higgs bilinear $|H|^2$ [65] or neutrinos via a new operator [66], or a new BSM mediator. In particular, the new mediator can be a heavy Z' boson decaying in the s -channel [67, 68] through the process represented in Fig. 1.7. This assumption will add the following term to the SM Lagrangian density:

$$\mathcal{L}_{Z' \text{ portal}} = -\frac{1}{4}Z'^{\mu\nu}Z'_{\mu\nu} - \frac{1}{2}M_{Z'}^2 Z'^\mu Z'_\mu + Z'_\mu (\bar{q}_i \gamma^\mu q_i + \bar{q}_{D\alpha} \gamma^\mu q_{D\alpha}) \quad (1.24)$$

where q and q_D are the SM and dark sector quarks, respectively, and $M_{Z'}$ is the Z' boson mass. The signatures at collider experiments arising from this scenario will be jets (see Sec. 3.4) with non-QCD-like features and missing energy (see Sec. 3.7) likely aligned with the jets. Searches for these type of jets, also called semi-visible jets, or for jets with displaced tracks, emerging jets, have been conducted in the ATLAS and CMS experiments at Large Hadron Collider using Run 2 data, and set limits on these processes [69, 70, 71]. The decay of dark hadrons depends on their spin, mass and the mediator to the visible sector. In the Model A, two dark mesons are produced, the dark pion π_d and the dark rho ρ_d . Dark pions are spin-0 mesons, much lighter than the other hadrons; they tend to decay into heavy SM quarks due to chiral flipping suppression. The dark rho, by contrast, is a spin-1 bound state of

dark quarks and, in the case where $m_{q_D} \ll \Lambda_D$, it decays promptly into two dark pions. The parameters concerning Model A are summarised in Tab. 1.1.

Parameter	Value
N_d	3
n_f	2
Λ_D	15 GeV
m_{q_D}	20 GeV
m_{π_d}	10 GeV
m_{ρ_d}	50 GeV

Table 1.1. Parameter values considered in Model A.

A recent ATLAS analysis has set the limits for the resonant production of DJ signatures within the Model A framework [72]. They have searched for a $Z' \rightarrow q_D \bar{q}_D$ resonance with Z' masses in the range [1.5, 3.5] TeV. A BumpHunter test showed a small discrepant region between 1.5–1.7 TeV (Fig. 1.8a), but the global p-value of 0.63 indicates no significant deviation from the background-only hypothesis. For $g_q = 0.05$ and $g_{q_D} = 0.2$, the coupling of the Z' to SM and dark quarks respectively, the observed limit on the Z' mass is set up to 3 TeV for Model A, as shown in Fig. 1.8b.

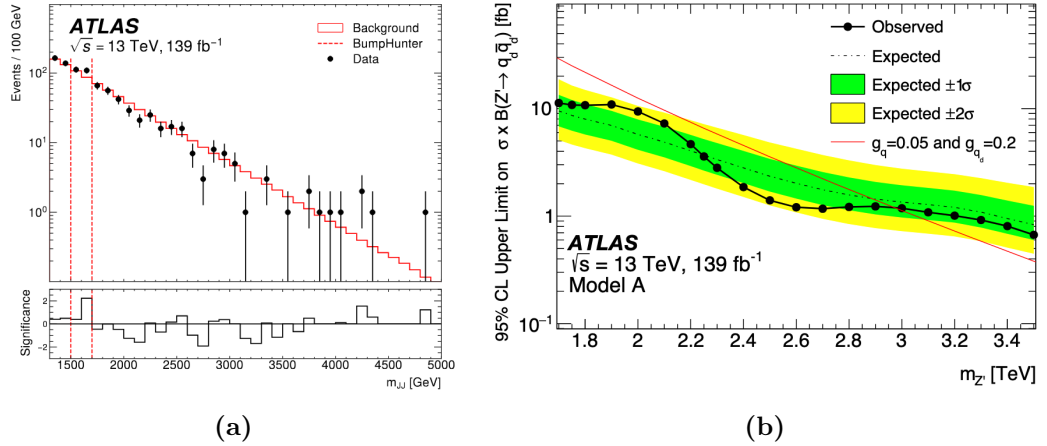


Figure 1.8. In Fig. 1.8a, the BumpHunter scan in the dijet invariant mass for the resonant DJ ATLAS search. A small deviation has been found at 1.5-1.7 TeV, but the global p-value indicates the result as compatible with the background-only hypothesis. In Fig. 1.8b, the observed and expected limits at 95% CL on the cross-section times the branching ratio for the signal production for Model A are shown. Figures from Ref. [72].

Chapter 2

The ATLAS detector

In this chapter, an overview of the ATLAS detector at the LHC is presented, focusing on its core components and their roles in particle detection. The discussion includes the detector's general structure, its key subsystems and the essential technologies that enable its operation. This foundation will support the detailed exploration of specific measurements and analyses in the subsequent chapters.

2.1 CERN

After the end of the Second World War, the absence of international agreement between the USA and the URSS and, at the same time, the awareness that many European countries did not have the capability to build, on a national level, the large apparatus required by the emerging high energy physics field led to the discussion of a scientific collaboration on a European level [73]. Many politicians and physicists of the stature of Raoul Dautry, Pierre Auger, Lew Kowarski, Edoardo Amaldi and Niels Bohr advocated for the construction of a laboratory complex aimed at purely scientific purposes, rejecting military and industrial applications. In July 1953, the Convention for the establishment of a European organization for nuclear research [74] was ratified by 12 founding Member States and, in September 1954, the European Organization for Nuclear Research (CERN) was born.

Today, CERN hosts a wide range of experiments and accelerator facilities and holds the largest and most energetic accelerator complex in the world, the Large Hadron Collider (LHC) depicted in Fig. 2.1. LHC has four interaction points where the ATLAS, CMS, LHCb and ALICE experiments are placed.

2.1.1 The LHC injection chain

The LHC, with a circumference of 26.7 km, is the largest circular collider and superconducting accelerator ever built, managing to achieve the record energy of 13.6 TeV in the interaction point for proton-proton collisions. To achieve this, a complex injection chain is required.

Protons are obtained from hydrogen through ionisation of the gas, then accelerated

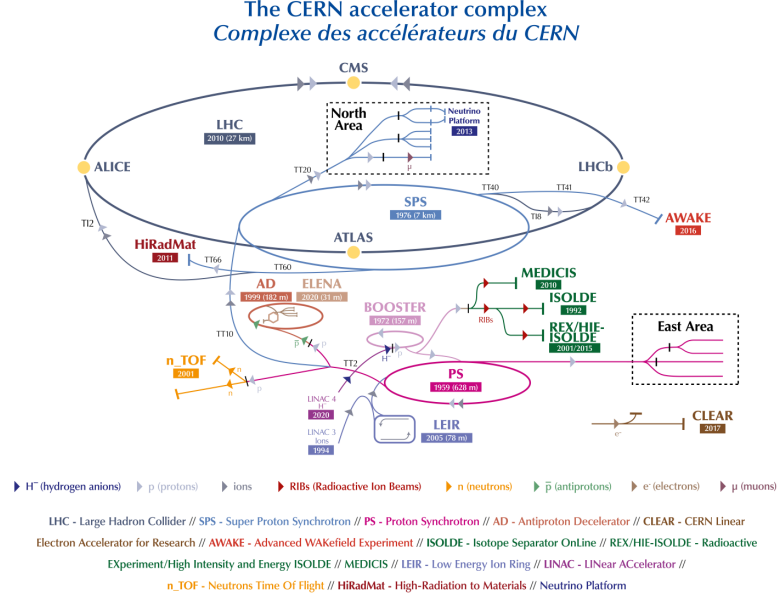


Figure 2.1. Schematic of all the accelerators and connection lines at CERN site. Figure from Ref. [75].

in the LINear ACcelerator-4 (Linac4) up to 160 MeV [76]. The protons, grouped in bunches (for more details see Sec. 2.1.2), are collected in the Proton Synchrotron Booster (PSB) achieving an energy of 2 GeV [77]. From here, the proton bunches reach the Proton Synchrotron (PS) and are accelerated up to 26 GeV [78]. After the PS, the protons arrive in the Super Proton Synchrotron (SPS) where they reach 450 GeV [79], then are injected into the LHC.

In addition to the proton-proton physics programme, the LHC can also operate with lead ions bunches: in this case the first two steps of the injection chain are replaced by dedicated accelerators, LINear ACcelerator-3 (Linac3) and Low Energy Ion Ring (LEIR). For further details about ion physics at LHC, see Ref. [80].

2.1.2 The LHC ring design

The LHC is a two-ring superconducting accelerator and collider at about 100 m below ground level in the Large Electron-Positron (LEP) tunnel, operational from 2008 and designed to reach a maximum energy of 14 TeV in the centre of mass [81]. It hosts two high-luminosity experiments, ATLAS and CMS, with an instantaneous luminosity (for a proper definition of the variable, consult Sec. 2.1.3) of $10^{34} \text{ cm}^{-2} \text{ s}^{-1}$ and two low-luminosity experiments, LHCb with an instantaneous luminosity of $10^{32} \text{ cm}^{-2} \text{ s}^{-1}$ and ALICE which operates with ions and reaches an instantaneous luminosity of $10^{27} \text{ cm}^{-2} \text{ s}^{-1}$. The LHC has an eight-fold symmetry, with eight straight sections called Interaction Regions and eight arcs; the beams circulating clockwise and anti-clockwise are referred as Beam1 and Beam2 respectively, as depicted in Fig. 2.2.

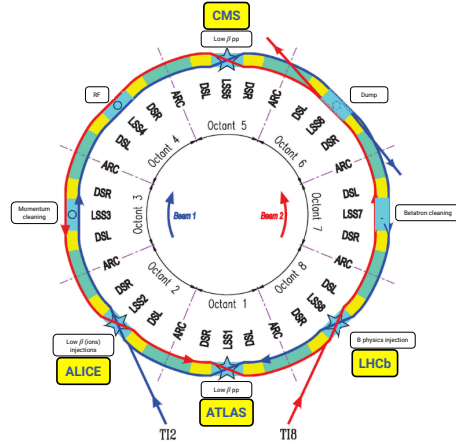


Figure 2.2. Schematic of the LHC ring; Beam1 (Beam2) is shown in blue (red). Figure from Ref. [81].

The main components of a particle accelerator are Radio Frequency (RF) cavities. An RF cavity is a metallic chamber that contains an electromagnetic field whose waves become resonant inside the cavity [82]. The electromagnetic force generated pushes forward the beam's charged particles and accelerate them. The LHC cavities are made of superconducting niobium film layered over a copper sphere and operate at 4.5 K; each cavity can reach a maximum of 5 MV m^{-1} accelerating field at 400 MHz.

Magnets help to optimise charged particles' trajectory inside the collider. Dipoles, quadrupoles, sextupoles, octupoles and decapoles are used to bend and focus the particle beams. LHC dipoles use superconducting magnets to generate a peak 8.33 T magnetic field [83]. The 1232 dipoles are made of niobium titanium (NbTi) cables which have a critical temperature of 10 K, but are operated at 1.9 K to ensure a 11 600 A electric current needed for the magnetic field.

In order to avoid collisions between the beam particles and the gas or dust molecules inside the beam pipe, an ultrahigh vacuum with a pressure of 10^{-10} mbar is required.

2.1.3 Beam parameters

Protons circulate inside the LHC ring grouped in bunches to best take advantage of RF acceleration, which occurs in a specific cycle. A maximum of 2808 bunches can circulate inside the collider, with approximately 1.15×10^{11} protons per bunch and resulting in a 0.582 A beam current. The bunch size is not constant around the ring due to the focusing and squeezing effects of the magnets, nonetheless, they arrive 25 ns (or 7.5 m) spaced in the four interaction points where they can be approximated to have ellipsoidal shape.

The number of events associated with the process i generated per second in the LHC collisions (N_i) is directly related to the cross section of the process (σ_i) and to the machine's instantaneous luminosity ($\mathcal{L}$), through the formula:

$$dN_i = \sigma_i \mathcal{L} dt \implies N_i = \sigma_i \int \mathcal{L} dt \quad (2.1)$$

The machine's instantaneous luminosity is a quantity that depends only on the beam parameters. For bunches with transversal profile following a Gaussian distribution, assuming equal beam parameters for both circulating beams, it can be written as:

$$\mathcal{L} = \frac{N_b^2 n_b f_{rev} \gamma}{4\pi \epsilon_n \beta^*} F \quad (2.2)$$

N_b is the number of particles per bunch, n_b the number of bunches per beam, f_{rev} is the revolution frequency of the beam around the LHC ring, γ is the Lorentz factor of the particles in the beam¹, ϵ_n is the normalised transverse beam emittance, β^* is the beta function at the collision point arising from the solution to Hill's equation, which governs the transverse motion of particles in a focusing lattice [84]. The beta function is linked to the transverse beam size σ through the formula $\sigma = \sqrt{\epsilon_n \beta^* / \gamma}$. Finally, F the geometric luminosity reduction factor due to the crossing angle θ_c in the Interaction Point (IP), the latter given by:

$$F = \frac{1}{\sqrt{1 + \left(\frac{\theta_c \sigma_z}{2\sigma^*}\right)^2}} \quad (2.3)$$

Here, σ_z and σ^* are the Root Mean Squared (RMS) bunch length and transverse RMS beam size, respectively. The values of the beam parameters used by LHC are shown in Tab.2.1.

Beam parameters		Value
N_b	Number of particles per bunch	$1.4 - 2.3 \cdot 10^{11}$
n_b	Maximum number of bunches	2808
f_{rev}	Revolution frequency	11.245 kHz
γ	Lorentz factor	7247
ϵ_n	Normalised transverse beam emittance	1.30-1.70 μm
β^*	Beta function at Point 1 and 5	0.55 m
F	Geometric luminosity reduction factor	0.863
θ_c	Crossing angle	285 μrad
σ_z	RMS bunch length	7.5 cm
σ^*	Transverse RMS beam size	16.7 μm

Table 2.1. Nominal values of the beam parameters relevant to calculating the machine's instantaneous luminosity for 2022-24 data-taking. The Lorentz factor γ is computed assuming the actual proton beam energy of 6.8 TeV, not the nominal (7 TeV). Values from Ref. [85].

2.1.4 Bunch filling scheme, LHC cycle and Run schedule

To achieve this tremendous amount of luminosity, a well-structured bunch filling strategy has to be defined. In principle, given the RF and revolution frequency,

¹The Lorentz factor, or gamma factor, of a particle is defined as $\gamma = \frac{1}{\sqrt{1 - \frac{v^2}{c^2}}}$, where v is the speed of the particle and c is the speed of light in the vacuum.

there are 35640 available buckets around the machine; as the operational bunch spacing has been set to 25 ns, the available slots are reduced to 3564 [86]. The numbering of the bunches is consistent with the slot number, so that each bunch in each ring can be identified by its Bunch Crossing Identifier (BCID).

The physics data-taking involves the four experiments and the entire accelerator complex and therefore a very strict long-term schedule is provided. Periods of data taking that can last 2-4 years, that are called Runs, follow periods of Technical stop, or Shutdown. Run 1 covered a period from 2010 to 2013, after that there was the first Technical Stop, the Long Shutdown (LS) 1; Run 2 started in 2015 and finished 2018 and was followed by another Technical Stop, LS2. Currently, LHC is taking data for its Run 3, started in 2022 and expected to end in 2026 (for the long term LHC schedule, see Chap.7.1.1).

During a Run, the data taking is interrupted by small technical stops due to incidents with the machine or detectors, as well as socioeconomic reasons. Typically, after the Year-End Technical Stop (YETS) there is one month of commissioning with beams, then the proton physics runs until October, month dedicated to ion collisions.

The performance in terms of energy and luminosity delivered by LHC depends greatly on the machine conditions; therefore, each Run has its own features. In particular, during Run 1, the energy in the centre of mass ($\sqrt{s}$) was 7 TeV in 2010 and 2011, while in 2012 it was 8 TeV. The total integrated luminosity recorded in the ATLAS detector was 5.25 fb^{-1} in the first two years and 21.7 fb^{-1} in the last year. Run 2 managed to achieve the higher energy of 13 TeV in the centre of mass, with a total integrated luminosity of 147 fb^{-1} . Run 3 has improved the centre of mass energy to 13.6 TeV with an expected integrated luminosity around 200 fb^{-1} . Up to October 2024, the end of proton-proton collisions, the integrated luminosity recorded by ATLAS was 183 fb^{-1} , as shown in Fig. 2.3a.

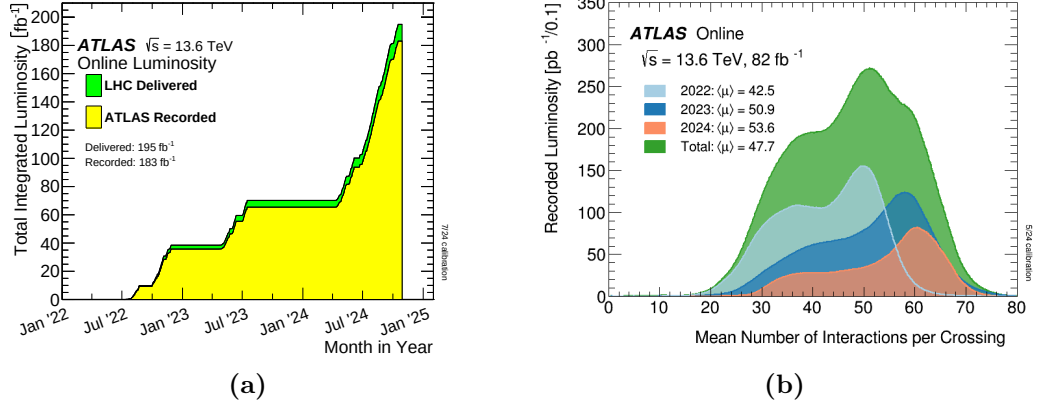


Figure 2.3. Performance of the LHC accelerator during Run 3. In Fig. 2.3a, the integrated luminosity for stable beams pp collisions at $\sqrt{s} = 13.6$ TeV. The delta between LHC delivered (in green) and ATLAS recorded luminosity (in yellow) reflects data acquisition inefficiencies and the so-called *warm start*, which is the temporal delay between the stable beam declaration and the actual recording of the events due to the ramp of high-voltage in specific sub-detectors. Figure from Ref. [87]. In Fig. 2.3b, the luminosity-weighted distribution of the mean number of interactions and the PU profile for the first three Run 3 years. Figure from Ref. [88].

2.2 The ATLAS detector

A Toroidal LHC ApparatuS (ATLAS) is a general-purpose particle detector, designed to exploit the full discovery potential of the LHC. With its cylindrical shape, 46 m of length and 26 m of diameter, ATLAS is the largest detector ever built for a particle collider. The feature that distinguishes ATLAS from other particle detectors, both past and present, is the presence of two magnetic fields. The magnet system consists of a solenoid placed around the inner detector cavity and superconducting air-core toroids positioned externally. The detector is structured in layers to facilitate the particle identification and event reconstruction. Starting from the IP and moving radially outward, the overall detector layout includes the Inner Detector, the Electromagnetic and Hadronic Calorimeters and the Muon Spectrometer. A general illustration of the ATLAS detector is portrayed in Fig. 2.4.

2.2.1 Coordinate system

The ATLAS coordinate system is right-handed, with its centre at the IP. The z axis is aligned along the beam line; the x - y plane lies transverse to the beam direction. The positive x coordinate is defined as pointing from the IP to the centre of LHC ring, while the positive y coordinate points upwards. The detector can be considered as divided in two halves along the z coordinate: the side A corresponds to the positive z values, the side C to negative z values; side B is the plane with $z = 0$. Given ATLAS cylindrical symmetry, it is important to define the azimuthal and polar angle in this coordinate system. The angle extending from the beam axis toward the x - y plane is the polar angle θ . The azimuthal angle ϕ is measured

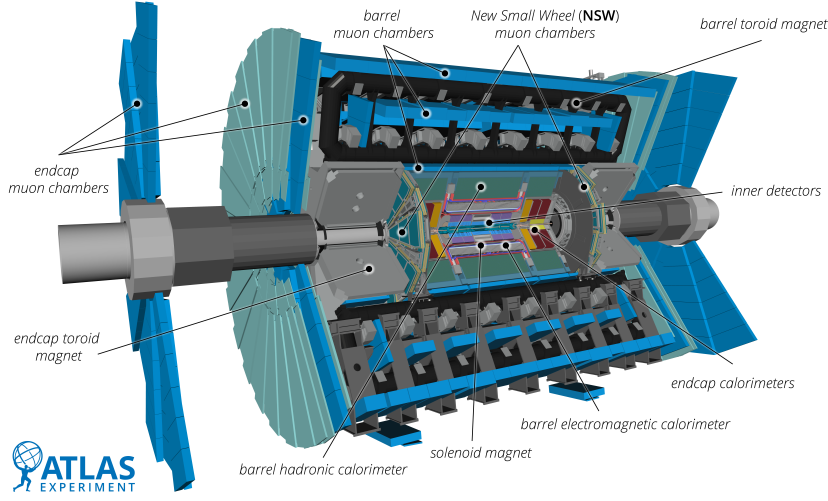


Figure 2.4. Overview of the ATLAS detector layout and its main components. Figure from Ref. [89].

around the beam axis. The θ angle lies in the $[0, \pi]$ range, with $\theta = 0$ lying on the positive z axis; the ϕ angle covers the $[-\pi, \pi]$ range.

Other important variables which derive naturally from the chosen coordinate system and cylindrical symmetry of the detector are described below. In accelerator physics, particles share the same direction, the z axis. It can be useful, then, to study physics in the centre of mass frame which is moving at some velocity along the z axis. The rapidity boost y , or simply rapidity, is the boost which takes an observer from the lab frame to the frame in which the particle moves only parallel to the beam and it is defined as:

$$y \equiv \frac{1}{2} \ln \left(\frac{E + p_z}{E - p_z} \right) = \operatorname{arctanh} \left(\frac{p_z}{E} \right) \quad (2.4)$$

where p_z is the z component of the particle three-momentum and E its energy. The difference of rapidities of two particles is invariant with respect to Lorentz boosts along the z axis.

Rapidity is crucial for the above-mentioned reasons, but in collider experiments it can be difficult to compute since the energy and total momentum of a particle are challenging quantities to measure. This introduces the concept of a related quantity, the pseudorapidity η , which can be defined based on geometrical considerations:

$$\eta \equiv -\ln \left[\tan \left(\frac{\theta}{2} \right) \right] = \operatorname{arctanh} \left(\frac{p_z}{|\mathbf{p}|} \right) \quad (2.5)$$

where $\mathbf{p}$ is the particle three-momentum. In the limit of particles travelling at the speed of light, rapidity and pseudorapidity are equivalent. The pseudorapidity is 0 when $z = 0$ and $\theta = \pi/2$, it tends to $\pm\infty$ as $\theta = 0$ or $\theta = \pi$. The regions of the detector at high values of pseudorapidity are referred to as *forward regions*.

The distance in the pseudorapidity-azimuthal angle plane is defined as:

$$\Delta R = \sqrt{(\Delta\eta)^2 + (\Delta\phi)^2} \quad (2.6)$$

In pp collisions, partons inside the protons are the real interacting particles. The initial state of the partons immediately before the collision is unknown, but we know that collisions happen along the beam axis. Energy and momentum must be conserved before and after the interaction and this consideration leads to other useful quantity definitions. In fact, the transverse component of momentum p_T is a conserved quantity and it is defined from the x and y component of the three-momentum as:

$$p_T = \sqrt{p_x^2 + p_y^2} \quad (2.7)$$

The coordinate system here provided will be adopted throughout this thesis to uniquely describe the topology of events occurring in the ATLAS detector.

2.2.2 Particle Identification

The cylindrical shape and layered arrangement in the ATLAS detector have been designed to exploit the different responses of particles when interacting with detector materials. The magnet system is necessary for bending the trajectories of charged particles to distinguish them from neutral particles. Particles moving perpendicularly to a constant magnetic field $\mathbf{B}$ with charge q and velocity $\mathbf{v}$ are subject to the Lorentz force $\mathbf{F}$:

$$\mathbf{F} = q \mathbf{v} \times \mathbf{B} \quad (2.8)$$

In the perpendicular plane the equation becomes:

$$\frac{mv^2}{r} = qvB \implies p_\perp = qrB \quad (2.9)$$

where m is the mass and r the bending radius. Eq. 2.9 is particularly relevant for particle detectors as it establishes a relationship between the conserved and otherwise difficult-to-measure perpendicular momentum $p_\perp$ and easier-to-determine quantities, such as the magnetic field intensity B and the bending radius r . However, measuring the bending radius r can be challenging and calculations can be simplified using some trigonometric adjustments in the formula:

$$p_\perp = \frac{0.3 BL^2}{8s} \quad (2.10)$$

where s is the sagitta and L the chord of the circumference, as shown in Fig. 2.5. The sagitta s and the chord L are much easier to measure within detectors that identify particle tracks.

Tracking detectors are positioned closest to the IP and are expected to detect all charged particles by measuring their transverse momenta as described by Eq. 2.10. Usually, trackers are made of light materials so that particles do not lose all their energy by passing through them. The typical technologies used for trackers are gaseous detectors or silicon-based detectors.

Outer to the tracking detectors, there are the electromagnetic and hadronic calorimeters. The concept behind calorimeters in general is to fully contain the particles passing through from the release of the particles' energy. For this reason, they are

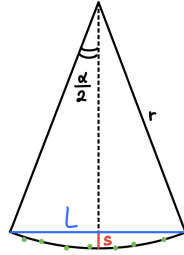


Figure 2.5. Sketch of the determination of the bending radius r from the sagitta s and the chord L of the circumference. Calculations in Eq. 2.10 are obtained under the assumption that the angle α is small, thus $\alpha = L/r$. The green dots represent the position of the tracks left by the charged particle while interacting in the detector under the presence of a perpendicular magnetic field.

designed to be several interaction-lengths deep (the actual thickness depends on the particles calorimeters are supposed to detect). The electromagnetic calorimeter is intended for electrons and photons, the hadronic calorimeter for hadrons. Typical technologies for the calorimeters involve the use of two materials, one heavier functioning as absorber and the other lighter as active material which produce an electric current that is later collected and measured.

The outermost component of particle detectors is the muon spectrometer. In fact, muons interact very little with matter and are able to arrive up to the spectrometer almost without energy losses. The muon system tracks the muons' trajectory and reconstruct their momentum. Muon detectors need to cover large areas, therefore the materials used here are usually cheaper with respect to those of the tracker. The choice of the technology typically falls into gaseous detectors.

2.2.3 Magnet system

Unlike other particle detectors both past and present, ATLAS magnet system is composed not only of an internal solenoid, but also of one barrel and two end-cap air-core toroids, with an overall dimension of 25.3 m in length and 20.1 m in diameter [90], as shown in Fig. 2.6a. The unusual configuration and size are intended to provide better momentum resolution while simultaneously reducing particle interactions with inactive material. To achieve an intense magnetic field, superconducting magnets made of aluminium-stabilised niobium titanium (NbTi) are cooled by pumping of liquid helium at 4.5 K, well below the NbTi critical temperature of approximately 10 K [91].

The central solenoid is positioned in front of the Electromagnetic Calorimeter and shares the cryostat and supportive cylinder with it; this pairing helps minimise overall material thickness. The central solenoid is 5.3 m long, 19 cm thick and has a 2.44 m internal bore. Inside the magnet a 7.73 kA electrical current flows so that it achieves a peak field of 2.6 T in the windings, resulting in an almost uniform 2 T field centrally.

The toroidal field is generated by the barrel and end-cap toroids. The barrel toroid

surrounds the Calorimeters and encapsulates the barrel of the Muon System; it is 25.3 m long, has an inner diameter of 9.4 m, while the outer diameter is 20.1 m long. The two end-cap toroids seal the magnetic field at the ATLAS edges; each end-cap toroid is 5 m long, with an inner diameter of 1.65 m and an outer diameter of 10.7 m. Each of the three toroids is composed of 8 coils assembled radially and symmetrically around the beam axis; the coils of the end-cap toroids are rotated by $\pi/8$ relative to the barrel coils. The barrel and end-cap peak fields reach 3.9 T and 4.1 T, respectively, resulting from a 20.5 kA current flow.

The overall magnetic field provided by the ATLAS magnet system, shown in Fig. 2.6b and 2.6c, causes charged particles to bend in the transverse plane within the solenoid region and in the longitudinal plane within the toroidal regions.

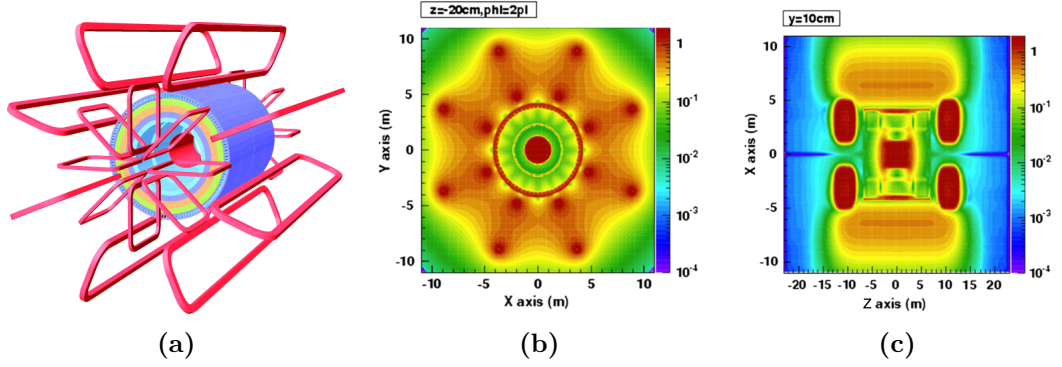


Figure 2.6. Overview of the ATLAS magnet system. Fig. 2.6a shows the schematic view of the magnet system. Figure from Ref. [92]. Fig. 2.6b depicts the transverse magnetic field in the centre of the experiment, thus the end-cap toroidal field is not present. The eight toroidal barrel coils are clearly visible, as well as solenoid magnet (not to be confused with the thick red iron shell, part of the calorimeter, which acts as return yoke for the field) [93]. Fig. 2.6c represents the longitudinal view of the magnetic field at $y = 10\text{ cm}$. This condition corresponds at $\phi \approx 0$, which translate in the end-cap regions having their maximal magnetic field and the barrel, on the contrary, their minimal [93].

2.2.4 Tracking

In the ATLAS detector, the Inner Detector (ID) [94, 95] is responsible for the tracking of charged particles. It consists of three different sub-detectors: the Pixel Detector, the SemiConductor Tracker (SCT) and the Transition Radiation Tracker (TRT), as shown in Fig. 2.7. Fine-granularity detectors, such as the Pixel Detector and the SCT, are needed to provide for excellent momentum and track resolution; thus, semiconductor and silicon technologies are used. The ID is contained in a cylindrical envelope 7.02 m long and 2.30 m in diameter. The silicon detector is surrounded by straw tube trackers, which constitute the TRT. The gaseous TRT also provides a good solution to the minimisation of material thickness in a region so close to the collision site. The ID performance on momentum resolution is robust and corresponds to 94% at 1 GeV, the threshold used in pattern recognition, and reaches a 99% plateau at approximately 5 GeV [96] during Run 3. The proximity to the IP (the closest silicon layer is 5.05 cm away from the IP) requires that all

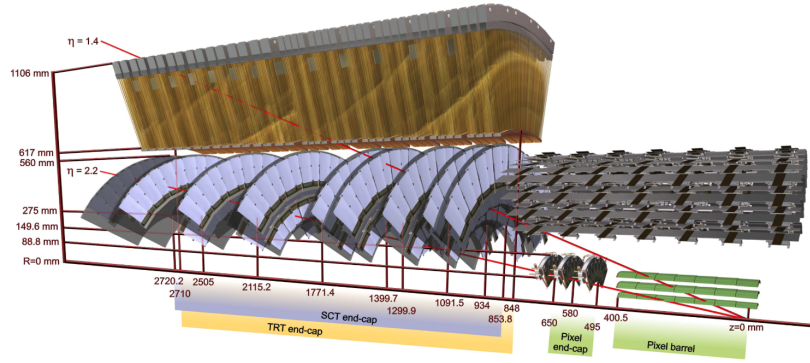


Figure 2.7. Quarter longitudinal section of the ATLAS Inner Detector; all length values are expressed in mm. Figure from Ref. [92]

the sub-detectors' materials and electronics should be radiationally hard. These requirements loosen the further the detector is from the IP, nonetheless the Pixel Detector is designed to stand 300 kGy of ionising radiation and 5×10^{14} 1 MeV neutrons per cm^2 . The overall ID coverage ranges in $|\eta| \leq 2.5$ and provides for robust pattern recognition in both primary and secondary vertex measurements.

Pixel Detector

The Pixel Detector has a barrel plus two end-caps structure, each consisting of three layers of pixel sensors. The barrel layers are placed in concentric cylinders around beam axis with radii of 50.5 mm, 88.5 mm and 122.5 mm and they are all 400.5 mm long. This provides coverage of $|\eta| \leq 2.5$ for the innermost layer and $|\eta| \leq 1.7$ for the outer two. The end-cap layers consist of 3 disks on each side, covering the $1.7 \leq |\eta| \leq 2.5$ regions. The disks are positioned at $z \pm 495$, 580 and 650 mm respectively. Each layer consists of 250 μm thick oxygenated n-type wafers (the sensors) with readout pixels. Each sensor covers a $19 \times 63 \text{ mm}^2$ area and contains 46080 readout pixel channels, each pixel being $50 \times 400 \mu\text{m}^2$ (or $50 \times 600 \mu\text{m}^2$ for the pixels near to the Front-End (FE) electronics). The overall resolution is 10 μm in the r - ϕ plane and 115 μm in the z direction.

During LS2 an additional 3D pixel sensor layer, the so-called Insertable B-Layer (IBL), has been inserted at $r = 33.25$ mm to provide additional hit information crucial for b -tagging performance [97].

SemiConductor Tracker

The SCT consists of 4 barrel layers covering the $|\eta| \leq 1.4$ region and 9 end-cap wheels on each side, providing further coverage in $1.4 \leq |\eta| \leq 2.5$. Each silicon detector covers an area of $6.36 \times 6.40 \text{ cm}^2$ and contains 768 readout strips spaced with an 80 μm pitch. Two adjacent silicon detectors are wire-bonded together to form 12.8 cm long strips; these pairs are glued back-to-back together with other pairs tilted with a stereo angle of 40 mrad. The combination of four silicon detectors

arranged in such a way composes a barrel module; forward modules have very similar displacement, but make use of tapered strips with one set aligned radially. Forward modules come in two lengths: 12 and 7 cm long. Barrel modules are mounted on carbon-fibre cylinders with radii 299 mm, 371 mm, 443 mm and 514 mm. The forward modules are placed concentric rings (up to 3) to form 9 wheels. The spatial resolution in the r - ϕ plane is $16\,\mu\text{m}$ and in the z direction is $580\,\mu\text{m}$.

Transition Radiation Tracker

The TRT is made of 4 mm diameter polyimide drift tubes, or straws; inside each tube a $31\,\mu\text{m}$ diameter tungsten wire, plated with 0.5 - $0.7\,\mu\text{m}$ gold, is placed. Each straw contains a mixture of gases composed by 70% xenon, 27% carbon-dioxide and 3% oxygen. Barrel tubes are 144 cm long, while the forward tubes are 37 cm long. Under normal operating conditions, the electron collection time is around 48 ns, while the drift-time accuracy is $130\,\mu\text{m}$. In the barrel, 50k straws are arranged axially around the beam direction starting from 554 mm radius up to 1082 mm, thus covering the $|\eta| \leq 0.7$ region. In the end-caps, 320k radial tubes cover the $0.7 \leq |\eta| \leq 2.0$ region, with the first tubes placed at $|z| = 848\text{ mm}$ and the last ones at $|z| = 2710\text{ mm}$.

The TRT assures a good pattern recognition thanks to continuous tracking; it also provides a set of measurements roughly corresponding to a single point with $50\,\mu\text{m}$ precision and which contributes to the momentum measurement. These measurements are used for simple and fast track triggers, as better discussed in Sec. 2.2.7.

2.2.5 Calorimetry

ATLAS calorimetry is based on sampling detectors, composed of alternating layers of active material and absorber [98, 99, 100]. Calorimeters are designed to fully contain the electromagnetic and hadronic showers produced by particles passing through the absorber material and to collect the energy deposited in the active medium. Their length can be expressed as a function of radiation length X_0 or interaction length λ , defined as the average distance a particle must travel in the material to reduce its energy by a factor $1/e$ due to electromagnetic or strong interaction, respectively. Three cryostats are used to cool the liquid argon, which serves as active medium: one for the barrel of the Electromagnetic Calorimeter (ECal) and two for the end-caps, which house the Electromagnetic End-cap Calorimeter (EMEC), the Hadronic End-cap Calorimeter (HEC) and the Forward Calorimeter (FCal). A liquid argon presampler detector is placed between the ID and the ECal to measure the energy lost before particles enter the calorimeter system. The ECal consists of accordion-shape lead absorbers and electrodes interleaved with liquid argon active layers. The HCal is composed of scintillator tiles and steel absorbers, the HEC of liquid argon and copper absorbers, while the FCal of liquid argon and copper-tungsten absorbers. The ECal barrel covers the region $0 \leq |\eta| \leq 1.475$, while the EMEC covers $1.35 \leq |\eta| \leq 3.2$. The region in the range $1.37 \leq |\eta| \leq 1.52$ is called the *crack-region* due to the presence of large amount of inactive material. The reconstruction of electrons and photons in this region requires attention and

will be described in details in Sec. 3.3. The HCal consists of three barrel regions: one central and two extended barrels; the overall coverage of this sub-detector is $0 \leq |\eta| \leq 1.7$. The HEC is situated in the $1.5 \leq |\eta| \leq 3.2$ region while the FCal covers the region $3.1 \leq |\eta| \leq 4.9$. A representative picture of the complete geometry of ATLAS calorimeters is shown in Fig. 2.8.

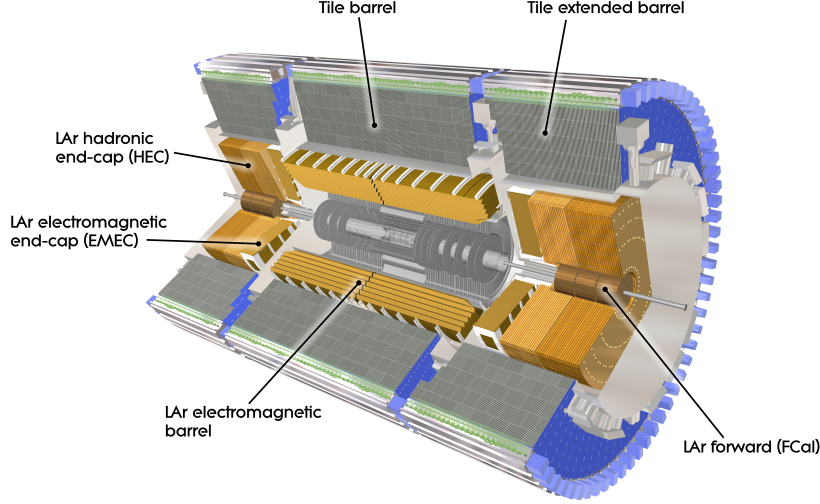


Figure 2.8. Representation of ATLAS calorimeter system. Figure from Ref. [89].

Electromagnetic Calorimeter

The ECal is composed of the ECal barrel and the EMEC detectors. It provides a uniform performance in ϕ thanks to its geometry and a fast signal extraction of 450 ns. In the barrel, the accordion waves are arranged axially, and radially in the end-caps; the folding angle varies depending on the distance to the IP, causing the gap between two consecutive absorber plates to be smaller near the IP and larger farther away. The lead plates have different thickness which ranges from 1.13 mm in the barrel to 2.2 mm in the end-caps. The barrel is divided in two halves, each 3.2 m long, with an inner diameter of 2.8 m and an outer diameter of 4.0 m. The end-caps consists of two coaxial wheels, with inner diameter of 66 cm and outer diameter of 420 cm. There is no η discontinuity in the barrel, nonetheless each half is divided into 16 ϕ -segmented modules for the readout, each of the modules spanning a $\Delta\phi = \pi/8$ angle. The elementary cells have different η and ϕ segmentations, ranging from $\Delta\eta = 0.003125$ -0.075 to $\Delta\phi = 0.025$ -0.1. Readout electrodes are used to extract the signal from the liquid argon between two consecutive lead plates; in the barrel and in the outer wheel of the end-caps the electrodes are read in three layers and in two layers in the inner wheel. The segmentation in the readout electrodes is denser in the first layers and coarser in the others, as they are intended to contain the tail of the electromagnetic shower, as shown in Fig. 2.9a. In fact, depending on the η region the ECal detector spans from 22 to 36 X_0 , enough to fully contain the whole

shower. The overall ECal energy resolution is:

$$\frac{\sigma_E}{E} = \frac{10\%}{\sqrt{E[\text{GeV}]}} \oplus 0.7\% \quad (2.11)$$

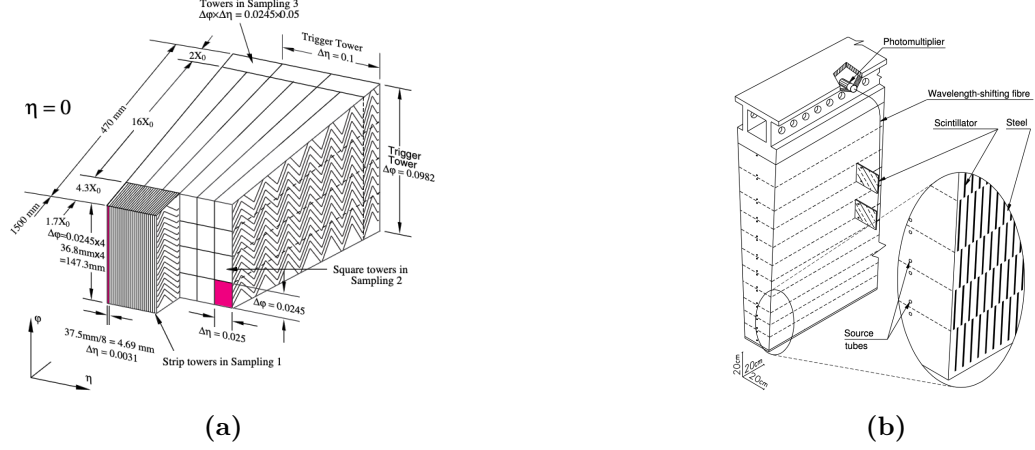


Figure 2.9. Fig. 2.9a represents the accordion shape of the ECal detector at $\eta = 0$; the segmentation in the η - ϕ plane for the three layers is highlighted. Figure from Ref. [99]. Fig. 2.9b shows the tile calorimeter design. Figure from Ref. [100].

Hadronic Calorimeter

The HCal consists of the tile barrel and extended barrels, the HEC and the FCal. The tile calorimeter has an inner and outer diameter of 4.46 m and 8.50 m, respectively. It is 5.8 m long in the central barrel and 2.6 m long in each of the extended barrels. The tile calorimeter covers at least 7.4λ and it is segmented into 64 modules, each spanning a $\Delta\phi \sim 0.1$. The structure is almost periodic, alternating steel and polystyrene tiles with a ratio 4.7:1. The scintillating tiles are 3 mm thick, have a radial length which ranges between 97 mm and 187 mm, the azimuthal length is variable too, between 200 mm and 400 mm, as shown in Fig. 2.9b. The towers cover a $\Delta\eta \times \Delta\phi$ region of 0.1×0.1 when $|\eta| \leq 2.5$, 0.2×0.2 when $2.5 \leq |\eta| \leq 3.2$. The size of $\Delta\eta$ tower coverage in the FCal depends on the specific pseudorapidity region, ranging from 0.1 to 0.5; the $\Delta\phi$ tower coverage, instead, is 0.4. When ionising particles pass through the polystyrene, the material emits in the ultraviolet, thus a fluorine doping is used to shift into visible wavelength. The photon emission is then collected by 1 mm in diameter optical fibres, then grouped into plastic channels and finally read by a PhotoMultiplier Tube (PMT). The fibre grouping defines a three-dimensional cell structure, forming three radial layers with sampling depth approximately corresponding to 1.5 , 4.1 and 1.8λ . The cells have $\Delta\eta \times \Delta\phi = 0.1 \times 0.1$ dimension and $\Delta\eta \times \Delta\phi = 0.1 \times 0.2$ in the other two.

The HEC is made of two cylindrical wheels on each side, with outer diameter 4.06 m. Front and rear wheels both contain 32 identical wedge shaped modules all around ϕ and the readout cells have dimension $\Delta\eta \times \Delta\phi = 0.1 \times 0.1$ when $1.5 \leq |\eta| < 2.5$ and $\Delta\eta \times \Delta\phi = 0.1 \times 0.2$ when $2.5 \leq |\eta| \leq 3.2$.

The FCal is exposed to a large particles flux, covering a very high η region and being 4.7 m far from the IP. It is split into 3 layers, each of them is deep 45 cm. The innermost layer is in reality part of the ECal and the absorber material is copper; the other two layers are part of the HCal and are made of tungsten.

The overall energy resolution of the HCal is given by a slightly modified version of Eq. 2.11:

$$\frac{\sigma_E}{E} = \frac{a}{\sqrt{E[\text{GeV}]}} \oplus b \oplus \frac{1\%}{E} \quad (2.12)$$

where the a and b terms are reported in Tab. 2.2.

	Statistical term a	Constant term b
pions: total energy	$(43.0 \pm 1.3)\%$	$(3.6 \pm 0.2)\%$
jets: total energy	$(46.0 \pm 1.5)\%$	$(2.3 \pm 0.5)\%$
jets: cone size $\Delta R = 0.7$	$(59.8 \pm 1.8)\%$	$(1.8 \pm 0.5)\%$
jets: cone size $\Delta R = 0.4$	$(69.6 \pm 2.4)\%$	$(1.2 \pm 1.0)\%$

Table 2.2. Nominal values of the statistical and constant terms of the energy resolution formula for the HCal.

2.2.6 Muon system

The muon system, also called Muon Spectrometer (MS), consists of different sub-detectors that measure the momenta of particles able to arrive up to the outermost part of the ATLAS detector with $p_T > 3 \text{ GeV}$, which are mostly muons. Similarly to the other ATLAS sub-detectors, the MS is divided into barrel and end-caps and, altogether, it can detect muons up to $|\eta| \leq 2.7$ [101]. Both the barrel and the end-caps have a 16-fold segmentation in ϕ , to follow the symmetry of the magnet system. Chambers are divided into large and small sectors, large sectors covering regions between two toroidal coils, small sectors ranging around the coils. The barrel covers the region $|\eta| \leq 1.05$, while the end-cap covers the region $1.05 \leq |\eta| \leq 2.7$ region for tracking and $1.05 \leq |\eta| \leq 2.4$ for the Level 1 trigger. At the centre of the ATLAS detector, at $|\eta| \sim 0$, a gap in the chambers was left to allow for services to the solenoid magnet, to the calorimeters and to the inner detector. The barrel sub-detectors of the MS consist of three cylindrical shells located between and on the barrel toroid situated at 5 m, 7.5 m and 10 m. Two different technologies are used: Resistive Plate Chambers (RPC) for fast trigger response and Monitored Drift Tubes (MDT) for precision tracking and momentum measurements, as reported in Fig. 2.10a. The end-cap sub-detectors are located in front (the Small Wheel) or behind (the Big Wheel) the end-cap toroids; each consists of three disks placed at $|z| = 7.4 \text{ m}$, 10.8 m and 21.5 m . The Small Wheels, made of MDT, Thin-Gap Chambers (TGC) and Cathode Strip Chambers (CSC), have been affected by ageing effects; therefore the detectors have been replaced by the New Small Wheel (NSW) [102], operative from Run 3 and shown in Fig. 2.10b. The NSW is composed of two multilayers, each comprising four small-strip Thin-Gap

Chambers (sTGC) and four MicroMegas (MM) planes arranged as follows: sTGC - MM - MM - sTGC. The big wheels consist of a complex of 8 movable and 2 fixed (the outermost) disks with an outer diameter of 25 m. The inner wheels can slide longitudinally along the beam direction to access the central part of the detector. The big wheels are composed of four TGC and two MDT layers for each side.

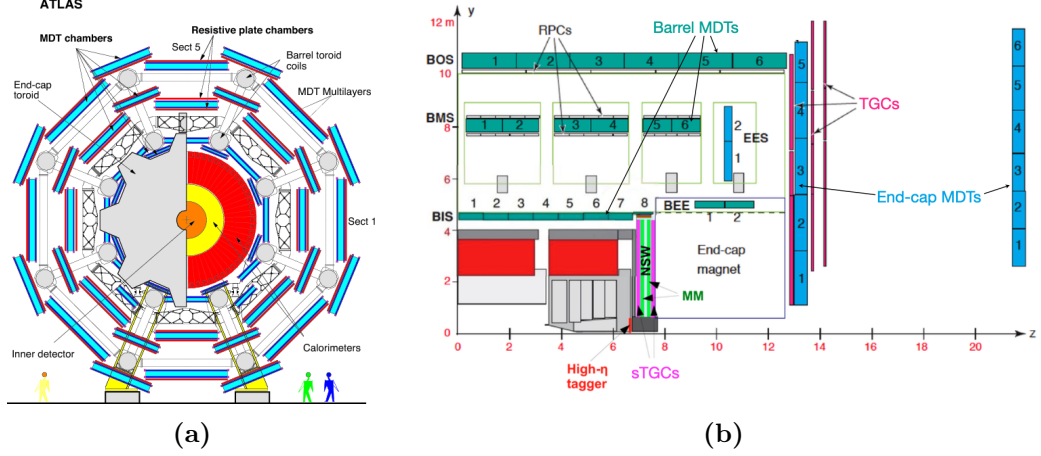


Figure 2.10. Fig. 2.10a represents the transverse view of the ATLAS Muon Spectrometer. Figure from Ref. [103]. Fig. 2.10b shows the longitudinal view of the MS, with the NSW. Figure adapted from Ref. [104].

Monitored Drift Tubes

MDTs are aluminium tubes with a diameter of 30 mm and 400 μm wall thickness. Inside each tube, there is a 50 μm diameter central wire made of tungsten-rhenium alloy and a mixture of argon, methane and nitrogen (91%-5%-4% respectively) operated at 3 bar. The maximum drift-time is 700 ns from the wall to the wire and the MDTs provide an average resolution per tube of 80 μm . Each chamber consists of multi-layers of MDTs, with an overall rectangular shape for the barrel and trapezoidal shape for the end-cap petals. The multi-layer is composed by three or four drift tube layers and in each chamber, two multi-layers are separated by spacer bars with a height that varies between 6.5 mm and 317 mm, depending on the chamber type. To fulfil the requirements on momentum resolution, an accuracy of the relative positioning of the chambers with an accuracy of 10 μm is needed. In fact, every time the magnetic field is operative, the alignment of the chambers can vary and therefore should be monitored. Optical devices consisting in four light rays and sensors are positioned between the two multi-layer of each chamber and provide a constant monitoring of the deformation of the tubes. The overall resolution for a track segment in a MDT multi-layer is 50 μm or 40 μm depending whether the multi-layer has 3 or 4 drift-tube layers. This corresponds to a sagitta resolution of $\sigma_S = 45 \mu\text{m}$ and the momentum resolution is given by:

$$\frac{\sigma_p}{p} = \sigma_S \times \frac{p[\text{GeV}]}{500 \mu\text{m}} \quad (2.13)$$

Resistive Plate Chambers

RPCs' basic unit is the so-called gas-gap, which consists of two bakelite plates kept parallel at a distance of 2 mm, filled with a mixture of tetrafluoroethane (94.7%), isobutane (5%) and sulfur hexafluoride (0.3%). The bakelite plates are operated to generate an electric field of 4.9 kV mm^{-1} between the plates. When a particle passes through the gas-gap, it ionises the gases, creating an avalanche; ions and electrons are attracted to the cathode and anode, respectively. The generated signal is read out via metallic strips mounted on the both sides of the RPCs, allowing for orthogonal η and ϕ readout. The strips are arranged with a pitch ranging from 30.0 to 39.5 mm. Two gas-gaps are grouped together forming an RPC doublet layer; two doublets are situated above and below the middle barrel MDT station, also called Barrel Middle (BM), while a single doublet is placed at Barrel Outer (BO). RPCs provide an excellent time response, with a resolution of 1.5 ns, much faster than the MDTs; on the other hand MDTs measure the η coordinate with greater precision ($\sim 80 \mu\text{m}$ with respect to $\sim 1 \text{ cm}$ for the RPCs). For this reason, RPCs are used for the Level 1 trigger to provide a region where the muon should have passed and where the MDTs should look at, as discussed in Sec. 2.2.7.

Thin-Gap Chambers

TGCs are multi-wire proportional chambers composed of two graphite-epoxy layers containing a mixture of carbon dioxide (55%) and n-pentane (45%) gases. Gold-plated tungsten wires with a diameter of $50 \mu\text{m}$ and 1.8 mm pitch form the anode of the detector and are distant 1.4 mm from the cathode. The cathode consists of pick-up strips that run parallel to the wires. Signals arrive inside a time window of 25 ns with 99% of probability and therefore TGCs are used for the Level 1 trigger (cfr. Sec. 2.2.7). The timing resolution is approximately 4 ns, while the spatial resolutions are 2-3 mm in R and ϕ respectively.

MicroMegas

Micro mesh gaseous structure detectors, commonly known as MMs, consist of a planar drift electrode filled with a mixture of argon (93%) and carbon dioxide (7%). A steel mesh divides two regions: the 5 mm conversion/drift gap where the initial ions and electrons are generated by the passage of muons and a $128 \mu\text{m}$ amplification gap. Readout strips are placed on the anode plate with a pitch of 0.425-0.445 mm. An optical alignment of the MM chambers is performed similarly to the MDTs alignment.

2.2.7 Trigger and Data Acquisition system

The LHC proton bunches collide at a frequency of 40 MHz and with an average number of interaction of $\langle \mu \rangle \approx 50$ per crossing. This translates to a combined data volume of more than $60 \times 10^6 \text{ MB s}^{-1}$ and it would be extremely challenging to store and process all this data. However, since only a fraction of these events

contain useful information for both precision measurements and NP searches, the Trigger and Data Acquisition (TDAQ) system [105] is employed to efficiently select and store the most relevant data. The idea behind the trigger concept is to have a structure that can decide whether to store the event or not, based on its content. The time needed to take the decision, called *trigger latency*, helps also in delaying the readout time which, for many sub-detectors, is considerably longer than the time interval in which two consecutive pp interactions occur. Another important quantity associated with triggers is the *trigger rate*, which is the rate of events the TDAQ system can accept. The data acquisition system is responsible for the transport and assembly of the event information from the FE electronics up to the recording disks.

Given the ATLAS detector complex structure, the trigger system consists of two levels: the Level 1 (L1) and the High Level Trigger (HLT). The L1 is hardware-based and runs on custom-made electronics located on the detector or in the USA15 cavern, which is the cavern housing most of the electronics of the experiment and situated adjacent to the ATLAS experimental cavern. The L1 latency is $3\mu\text{s}$ and the maximum trigger rate is 100 kHz . The HLT is software-based and operates on a large farm of about 4×10^4 CPU cores. The HLT latency is 550 ms and the maximum trigger rate is 3 kHz during Run 3. An overview of the TDAQ system is shown in Fig. 2.11.

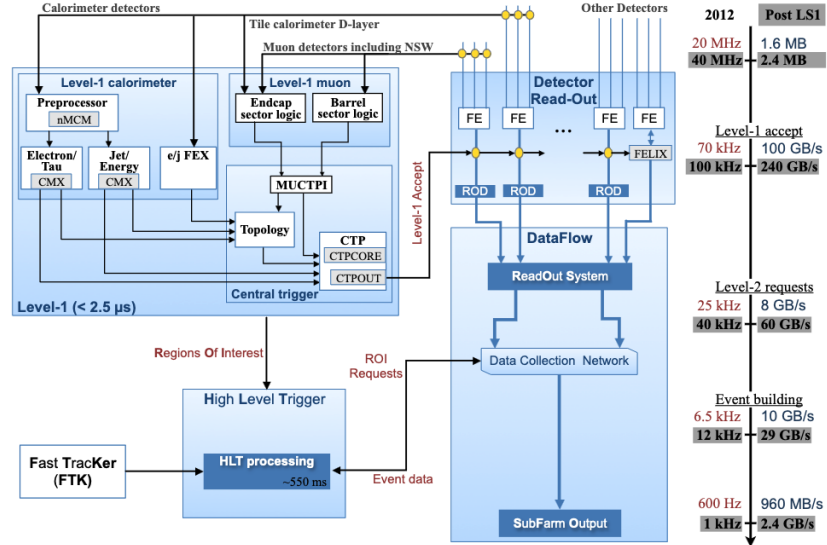


Figure 2.11. Block diagram of the ATLAS trigger system after the Phase I upgrade. The novelties with respect to Run 1 trigger configuration are highlighted in grey. Figure from Ref. [105].

Level 1 Trigger

The L1 trigger is a system of synchronous pipelined processors that receive input data from the fast muon trigger chambers, the RPCs for barrel and the TGCs for end-cap, and from the calorimeters [106]. Right after a collision, event fragments

are processed in parallel by the sub-detectors ReadOut Drivers (ROD); then the L1 trigger searches for large energy deposits or high- p_T muon tracks in the MS. The trigger-dedicated processors for the muons, the Muon Central Trigger Processor Interface (MUCTPI), and for the calorimeters, the Level-1 Topological Processor (L1Topo), send their data to the Central Trigger Processor (CTP). Meanwhile, data passes to the ReadOut System (ROS) which buffers the event and, if the L1-accept signal is given by the CTP, sends it to the HLT. Apart from the L1-accept signal, the L1 trigger identifies also some Regions of Interest (RoI) to pass to the HLT and to further examine the signals.

The L1 trigger is said to be clock-driven, as it runs at 40 MHz, the frequency of LHC collisions; the clock signal is directly derived from the LHC machine clock.

High Level Trigger and data flows

The HLT algorithms have access to larger event information, such as primary vertex positioning, the number of primary vertices and PU-dependent corrections. This information helps identifying whether the object that has triggered the L1-accept comes from the primary vertex hard interaction [107]. Moreover, at this level early track information is available, allowing for a better event and object selection with improved performance. If the event fires certain combinations of the L1 and HLT triggers, it is finally written to disk. These combinations, called *trigger chains*, also apply kinematic selections and are used by multiple ATLAS analyses. The list of all the trigger chains available in a certain period of time constitutes the *trigger menu*. The rates of events of a certain trigger chain can vary significantly during a Run, due to the increase of instantaneous luminosity. In these cases, a *prescale* of factor N is introduced, which means that only $1/N$ events passing the specific trigger chain will be stored. Physics analyses are usually conducted on *lowest un-prescaled* triggers, if possible, where no prescale is applied.

Data is organised into *streams*, each stream is a set of defined trigger selections and prescales that contains all the events recorded to disk after satisfying the selections. The stream typically used for physics analyses is called *physics_Main*, but there are several others specific for monitoring, calibrating and debugging the different sub-detectors. The events selected by the trigger chain are processed and stored at the Tier-0 facility at CERN; then they are archived and copied to Tier-1 around the world and finally derived datasets produced by the physics groups for the analyses are produced and copied at Tier-2 facilities.

Chapter 3

Object Reconstruction

The goal of physics analyses is to compare theoretically predicted quantities to those measured experimentally. Monte Carlo (MC) generators are used both to produce simulated events and to emulate the response of the real detector to the passage of the artificial particles. In this chapter, an overview of the steps required to produce data in the format used for ATLAS analyses is given.

3.1 ATLAS data preparation

ATLAS data undergoes different steps before being used in physics analyses; some of these steps are different for MC and physics events [108]. In Fig. 3.1, the whole data preparation process is shown. The first step for MC events is the event gen-

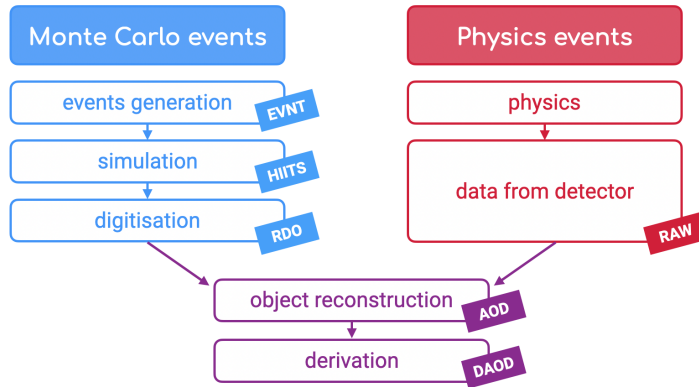


Figure 3.1. Block diagram of the ATLAS data preparation, in the cases of MC and physics data. On the side of every step, the name of the data format used is shown in the coloured box.

eration. The Feynman rules, which describe particle interactions, are derived from the Lagrangian of a theory, whether within the SM or BSM. These rules are then used to compute matrix elements, which are essential for predicting physical processes. The calculations are typically performed using *event generator programs*,

which simulate collisions and particle interactions. Event generators play the role of the LHC machine as they simulate the collisions between elementary particles. They are extremely important in HEP since they provide a way to estimate signals and backgrounds of complex processes and make comparisons with experimental data [109]. Generators, however, are far from perfect since they have to describe vast range of physics, some understood from first principles (mainly EW calculations) and some have to be calculated up to a certain order in perturbation theory (mainly QCD). ATLAS uses numerous event generators, the most common for Run 3 are: POWHEG [27], MADGRAPH5_AMC@NLO [110], SHERPA [111], PYTHIA [25] and HERWIG [26]. The last three can produce complete final states, while others, especially those implementing higher-order calculations, produce final states with few partons. Many of the generators used also implement estimations of systematic uncertainties that come from variations of the renormalisation and factorisation scales.

The steps required to build the event structure consists of several steps, can be performed almost sequentially in a MC generation. First, two hadrons collide and two of the partons composing the hadrons undertake the hard scattering of interest, being it SM or BSM. As particles involved are ultra-relativistic colour and/or electromagnetic charges, the particle emissions, or Bremsstrahlung, can occur. If the emission implicates particles that were initially colliding, it is called Initial State Radiation (ISR); if instead emission is associated with outgoing particles, it is Final State Radiation (FSR). If some short-lived resonances, such as the $W^\pm$ or Z^0 bosons, are produced, their decay is considered part of the hard process itself since some of their properties, like spin correlations, are passed to decay product particles. At this stage the pp interaction, the physics process of interest and their decay products have been generated, but they have not propagated into the detector.

The next step is, therefore, the *detector simulation* that emulates the detector response at the passage of these particles, including any decay and interaction with detector material. ATLAS includes two types of simulation: the *full simulation*, performed with the GEANT4 toolkit [112], and the *fast simulation* with ATLFAT3 [113] which parametrise the calorimeter response.

After the simulation, the energy deposits and particle tracks are read and the real detector response is emulated; this step is called *digitisation*. Each sub-detector is simulated in GEANT4, the PU is added and the event passes through a trigger simulation stage.

At this point the amount of information in the MC events is analogous to the Raw data coming from the real detector. The following steps, in fact, are common to both physics and MC data. First detector signals are converted into physics objects in what is called *reconstruction* step; then loose selection and calibration algorithms are applied, and finally the events are stored in the final format which includes useful information to perform an analysis, the *derivation*.

3.2 Tracks and vertices

Charged particles that transverse the ID lose part of their energy due to ionisation and radiative loss. The energy deposits are clustered in three-dimensional spatial

points, the *hits*; hits can be grouped to form *tracks* which represent the particle trajectories.

3.2.1 Track reconstruction

The track reconstruction is the most resource expensive, because the ID is the closest to the IP among the sub-detectors [114, 115]. The strategy used for track reconstruction starts with the formation of *track seeds*, by grouping together triplets of hits in the Pixel and SCT sub-detectors compatible with a single particle passage. The seeds are used to search for track roads by means of a combinatorial Kalman filter and track candidates are built. Track candidates are then fitted with a global χ^2 method; this procedure runs for several iterations, until either it aborts and the track candidate is discarded or achieves the precision required. Hits in the TRT are now added and the track candidates are fitted again using a Gaussian Sum Filter tool to improve momentum resolution and particle identification. This strategy is called *inside-out tracking* since it moves from the innermost ID sub-detectors to the outermost and it is outlined in Fig. 3.2a. Inside-out reconstructed tracks have a sequence of at least 8 hits and $p_T > 500$ MeV.

The inside-out tracking reconstruction is optimised for particles produced at the location of pp interactions; to recover particles originated at larger distance, mainly electrons from photon conversion, a *back-tracking* strategy is used. In this case the seed comes from RoIs determined by deposits in the ECal; the road search is conducted with hits in the TRT compatible with the electromagnetic deposits. The TRT segments are then extended towards the IP.

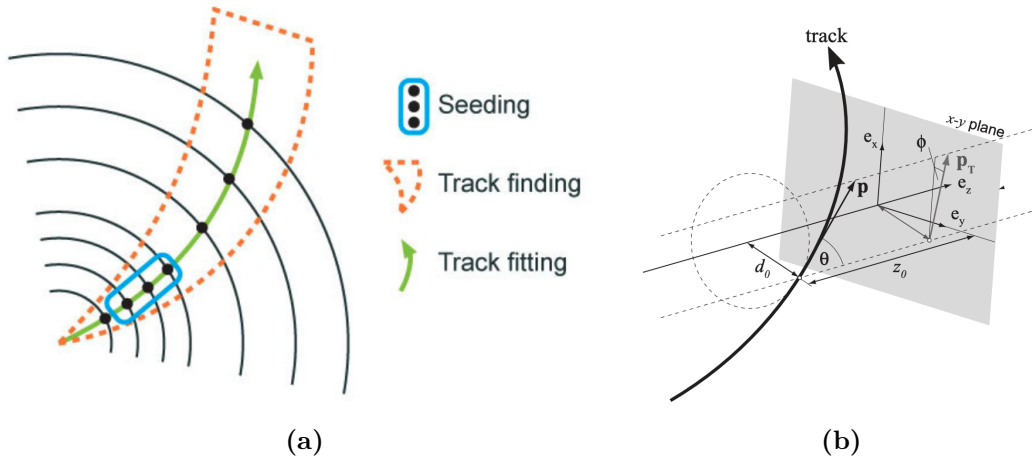


Figure 3.2. ATLAS track reconstruction process. In Fig. 3.2a the steps of inside-out tracking are shown. Fig 3.2b represents the illustration of the five global track parameters d_0 , z_0 , ϕ , θ , q/p . Figure from Ref. [114].

Once the track has been identified, it is possible to associate with it five important parameters, defined from a reference point called the perigee. The perigee is the point of closest approach to the beam line; the parameters, shown in Fig. 3.2b, are d_0 and z_0 , the transverse and longitudinal impact parameters, ϕ and θ , the

azimuthal and polar angles, and q/p , the charge divide by the momentum.

3.2.2 Primary vertex identification

In order to discard tracks not originated from the hard scattering, the position of the *primary vertex* have to be reconstructed. Once the tracks have been reconstructed, the collision points can be estimated as well from the distributions of the z_0 of all the tracks. A fit on the vertex location is then performed to identify the vertex candidates; the vertex with associated the largest sum of tracks p_T^2 is determined to be the primary vertex. The position of the so identified primary vertex is used as reference point for the reconstruction of all the other objects. The vertex position resolution along the beam line is estimated to be $18\text{ }\mu\text{m}$ for $t\bar{t}$ events and a bit higher for lower multiplicity processes.

3.3 Electrons and photons

Given ATLAS sub-detector design, electrons are supposed to leave tracks inside the ID and energy deposits in the ECal, while photons should pass almost completely untracked in the ID but loose all their energy in the ECal. The electrons and photons reconstruction comes naturally from this consideration and the procedure consists of five steps: the identification of energy deposits in the ECal, the *topoclusters*; the identification of ID tracks compatible with electrons; the track-cluster matching for the electrons; the identification of vertices in the ID consistent with photon-to-electron-positron-pairs conversion; the creation of the final object through *superclusters*. A reconstructed electron is a supercluster of an EM topocluster and a matching track. A reconstructed photon can originate from a *converted* photon, which consists of an EM topocluster and a matched conversion vertex, or an *unconverted* photon, made only by an EM cluster [116, 117].

3.3.1 EM topoclusters

EM topoclusters are topological cell clusters in the ECal sub-detector. The EM cell significance, ζ_{cell}^{EM} , is defined as:

$$\zeta_{cell}^{\text{EM}} = \left| \frac{E_{cell}^{\text{EM}}}{\sigma_{noise, cell}^{\text{EM}}} \right| \quad (3.1)$$

where E_{cell}^{EM} is the cell energy and $\sigma_{noise, cell}^{\text{EM}}$ the cell expected noise. To initiate a cluster, the cell must have a significance $\zeta_{cell}^{\text{EM}} \geq 4$; if a cell satisfies this requirement, it becomes a topocluster seed. Once the seed is identified, growth cells are searched among the seed neighbouring cells if they satisfy the condition $\zeta_{cell}^{\text{EM}} \geq 2$. Each of the growth cells becomes a new seed for searching other growth cells. If two or more topoclusters share the same cell, they are merged. If a topocluster contains more than one local minimum, defined as a cell with $E_{cell}^{\text{EM}} > 500\text{ MeV}$, it is split, cells are assigned to one topocluster or the other based on their distance from the

barycentre of the two. The EM topocluster procedure here described is used also for jets and taus, as described in Sec. 3.4. Therefore, to distinguish the EM component of the shower from PU and hadronic shower, a new quantity, the EM fraction f_{EM} , is calculated as the energy fraction in ECal. Applying a cut on the EM fraction so that $f_{\text{EM}} > 0.5$ results in a PU rejection of $\sim 60\%$ without affecting the true electrons topocluster efficiency.

3.3.2 Electron track and conversion vertex reconstruction

The procedure to find electron tracks in the ID is very similar to that previously described for standard tracks in Sec. 3.2.1, with few differences. First, the tracks are searched within the RoI of the topocluster position and the usual global χ^2 fit is performed.

The loosely-matched electron tracks require now a proper cluster matching. The track parameters are calculated again, extrapolating from the perigee to the second layer of the calorimeter. Tracks are matched to a topocluster if $|\eta_{\text{track}} - \eta_{\text{topoc}}| < 0.05$ and $-0.10 < q \cdot (\phi_{\text{track}} - \phi_{\text{topoc}}) < 0.05$. The asymmetry in the $\Delta\phi$ requirement arises from track missing energy due to photon emission which may be measured in the cluster. If multiple tracks are matched with the same cluster, a ranking is performed based on the sub-detector hits and the least ΔR distance from the topocluster barycentre.

Photons may leave in the ID some tracks coming from electron-positron pairs production. In these cases the electron and positron may be reconstructed as two tracks pointing to the so-called *conversion vertex*, compatible with a massless particle decay. To increase the converted photon purity, tracks reconstructed with the back-tracking procedure are preferred.

Conversion vertices found so far need a proper cluster-matching with a process similar to the electron track-cluster-matching. In general, track topologies compatible with photon decays are preferred, such as close-by, parallel tracks or tracks with shared hits. The vertex is matched if tracks satisfy tighter RoI conditions: $|\eta_{\text{track } 1,2} - \eta_{\text{topoc}}| < 0.05$ and $|\phi_{\text{track } 1,2} - \phi_{\text{topoc}}| < 0.05$; if the vertex has been reconstructed in the TRT, slightly looser conditions are requested.

3.3.3 Superclusters

Superclusters are the combination of a topocluster, or more than one, and matched tracks or conversion vertices that lead to the object definition. Topoclusters identified in Sec. 3.3.1 are listed in p_{T} descendent order and used as seeds for supercluster candidates. Electron supercluster candidates consist of one topocluster with energy $E \geq 1$ GeV and a matched track with more than 4 hits in the Pixel and SCT sub-detectors. Photon supercluster candidates require one topocluster with energy $E \geq 1.5$ GeV, with a matched conversion vertex (converted photon) or without (unconverted photon). Other topoclusters, called *satellites*, can be associated with the supercluster candidate, as they may be originated from Bremsstrahlung or topocluster splitting. If a satellite is assigned to a specific supercluster, it is removed from

the list of supercluster seeds to avoid double-counting. The supercluster iteration ends when there are no more candidates. The energy calibration is then applied to electron and photon superclusters [114].

3.4 Jets

Hadrons are composite particles, but made up of quarks and gluons as described in QCD. At LHC energy, one might expect to resolve the internal structure of hadrons; however, due to QCD colour confinement, quarks and gluons cannot be observed as free particles. Instead, in collider experiments, when an elementary particle carrying color charge is produced, it gives rise to a collimated spray of hadrons, known as a *jet* [118]. During the hadronisation, quarks and gluons undergo successive branchings at small angles into other partons, until they reach a non-perturbative scale at around 1 GeV. At this point, partons start forming hadrons due to colour confinement and these hadrons are measurable, as they leave a track in the ID and energy deposits in both the calorimeters.

The *jet definition*, however, is far from being simple since the decision whether two particles belong to the same jet or are part of two separate jets does not undergo a strict rule. Therefore, the jet definition is determined by the rigorous procedure that states how to reconstruct the jet in a process that mimics the inverse of parton shower.

3.4.1 Jet algorithms

The jet definition consists of a *jet algorithm* and a set of parameters associated with it. Given a jet algorithm J_{n+1} that cluster together a set of $n + 1$ partons $k_1, \dots, k_{n+1}$, the algorithm should follow two properties, the *collinear safety*, defined as:

$$J_{n+1}(\dots, k_i, k_j, \dots) \longrightarrow J_n(\dots, (k_i + k_j), \dots) \quad \text{if } k_i \parallel k_j \quad (3.2)$$

and *infrared safety*, which is:

$$J_{n+1}(\dots, k_i, \dots) \longrightarrow J_n(\dots, k_{i-1}, k_{i+1}), \dots) \quad \text{if } k_i \rightarrow 0 \quad (3.3)$$

These two conditions prevent the jet to change if a parton splits into two parallel partons, the collinear safety, or if an infinitesimally soft parton is added, the infrared safety.

Jet algorithms can be divide into two categories: cone and sequential recombination algorithms.

Cone algorithms

Cone algorithms first appeared in 1977 [119] and are based on the concept of *stable cone*: if the four-momenta sum of all the particles contained in the cone of centre (y_c, ϕ_c) and radius R gives a rapidity exactly equal to y_c , then the cone is stable. The procedure begins with a properly defined set of seeds and for each seed it is

assessed if the cone with centre in the seed is stable or not; this step is iterated until a stable cone, the jet, is found. The first cone algorithms were not infrared and collinear safe; it was only with the SIScone algorithm [120] that this problem was finally addressed.

Sequential recombination algorithms

The concept behind these type of algorithms is that since jets are the output of successive parton branchings, it is possible to reconstruct the jet by sequentially recombining two entities into one. Given a list of elementary constituents of the jet (for a proper definition of the constituents see Sec. 3.4.3), it is possible to calculate two quantities, the inter-particle distance between particle i and particle j defined as:

$$d_{ij} = \min(p_{Ti}^{2p}, p_{Tj}^{2p}) \cdot \Delta R_{i,j}^2 \quad (3.4)$$

and the beam distance of particle i from the beam axis B expressed as:

$$d_{iB} = p_{Ti}^{2p} \cdot R^2 \quad (3.5)$$

p is a free parameter, while R is the jet radius. For each particle i the algorithm calculates d_{ij} and d_{iB} for all the other j particles:

- if $d_{ij} < d_{iB}$, particles i and j are removed from the list and are recombined to form a new constituent k which is part of the list;
- if $d_{ij} > d_{iB}$, the particle i is a jet and it is removed from the list.

The jet radius denotes the typical size of the jet, while the p parameter defines different algorithms: the k_t algorithm [121] uses $p = 1$, the Cambridge/Aachen algorithm [122] sets $p = 0$ and the anti- k_t algorithm [123] uses $p = -1$. Jets identified with the anti- k_t algorithm have a conical shape, with the extreme case in which if a particle has no hard neighbours within a distance R , all the neighbours will cluster within a circumference of radius R . A comparison between the cone and sequential recombination algorithms is shown in Fig. 3.3. In ATLAS *small R jets* have $R = 0.4$ and *large R jets* have $R = 1$, both reconstructed with parameter $p = -1$.

3.4.2 Jet grooming and PU mitigation algorithms

Although it is fundamental that the jet algorithm remains unchanged if soft emission is added, it is nonetheless important that the soft radiation from the underlying events are removed from the jet and this is the task of jet grooming algorithms.

- *modified Mass Drop Tagger (mMDT)*: it de-clusters the jet and at each step it applies a mass drop condition to identify hard substructure.
- *Soft Drop (SD)*: it is similar to the mMDT algorithm, but it uses a broader approach by applying a condition on the momentum balance between the subjects during the de-clustering process.

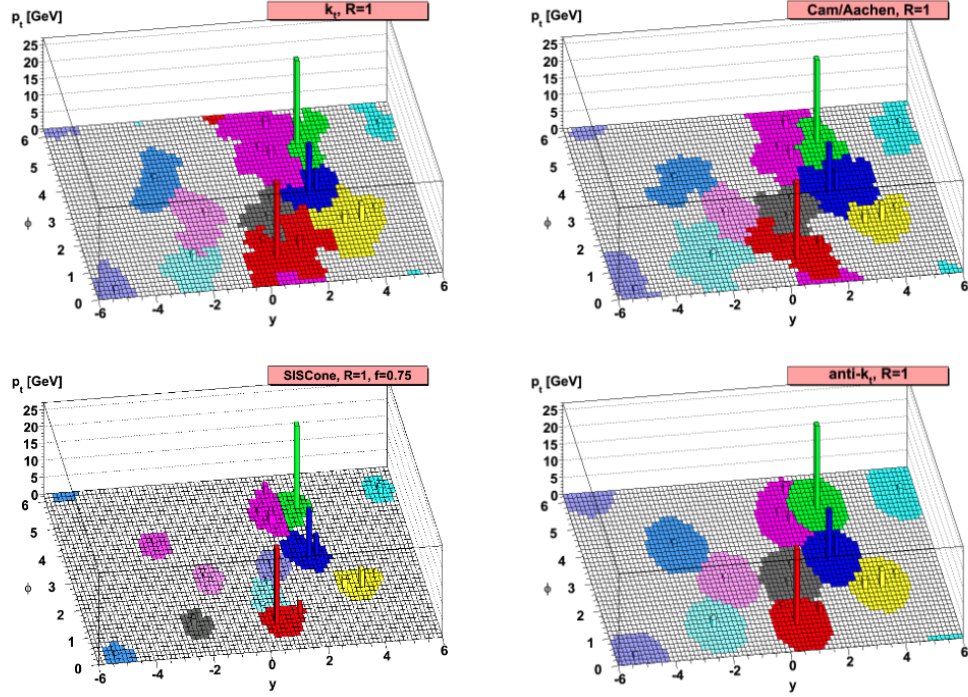


Figure 3.3. Behaviour of different jet algorithms to events containing partons generated with HERWIG together with soft emissions. The jets are correctly reconstructed in all the four cases; the shape of the jets, however, tend to vary significantly. The conical jet shape that arises from the anti- k_t algorithm (bottom right plot) is clearly visible. Figure from Ref. [123].

- *trimming*: it involves breaking down the jet into smaller subjets using a clustering algorithm like k_t or Cambridge/Aachen; the trimming technique removes the subjets that carry less transverse momentum than a predefined fraction of the original jet's transverse momentum, typically 5-10%.
- *pruning*: it works during the jet clustering process; as the jet is clustered, each pairwise combination of particles is examined. If the pair fails to meet certain criteria based on momentum balance and angular separation, the softer of the two particles is discarded rather than combined.

Prior to jet reconstruction, the energy deposits are given as input to PU mitigation algorithms, like *Constituent Subtraction* [124], *SoftKiller* [125] or *PUPPI* [126].

3.4.3 Jet reconstruction

The actual particles of which a jet is composed are: $\sim 65\%$ charged hadrons, such as $\pi^\pm$, $p/\bar{p}$; $\sim 10\%$ neutral hadrons, such as π^0 , $n/\bar{n}$, K_L^0 and $\sim 25\%$ lighter particles originated from the decay of unstable hadrons, mainly photons from $\pi^0 \rightarrow \gamma\gamma$. Particles, however, cannot experimentally be used as input for jet algorithms; ATLAS uses different approach to reconstruct the so-called *jet constituents*.

Topoclusters

Jet topological clusters, or simply topoclusters, are reconstructed in the same way of purely EM topoclusters [127], as described in Sec. 3.3.1. Topoclusters are searched in both the calorimeters with the ECal providing a higher granular response than the HCal. The significance of each cell is measured, clusters are seeded by cells with $\zeta_{cell} \geq 4$, all neighbouring cells with $\zeta_{cell} \geq 2$ are added and then all neighbouring cells, regardless of the significance, are added to the protocluster. A final cluster splitting step splits large protoclusters with multiple local minima; the stages of protocluster formation are shown in Fig. 3.4. In the case of electrons and pho-

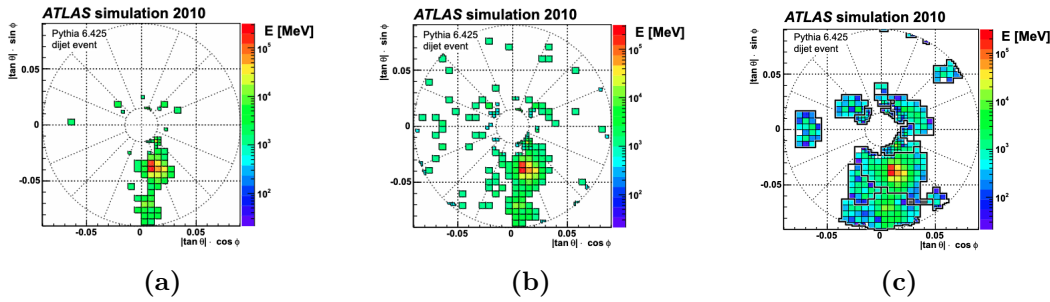


Figure 3.4. Stages of topocluster formation in the first module of FCal detector with simulated dijet events with at least one jet entering the calorimeter. Fig. 3.4a shows the seed cells with $\zeta_{cell} \geq 4$, in Fig. 3.4b the cell growth by adding cells with $\zeta_{cell} \geq 2$ and finally in Fig. 3.4c the protocluster is finalised with all the cells for which $\zeta_{cell} > 0$. Figures from Ref. [127].

tons this procedure is sufficient to guarantee an efficient object reconstruction; for hadronic jets this is no more valid, as hadronic showers are subject to much larger intrinsic fluctuations. The protoclusters here described contain only a fraction of the hadronic shower and this can be used to distinguish jets from electrons and photons. The cluster multiplicity, defined as the number of cells associated to a certain protocluster, can vary considerably depending on the particles which primarily have generated the shower. The kinematic distributions of the topoclusters are also excellent discriminant between hadronic and EM showers; in fact, it is possible to associate to each protocluster a four-momentum P_{clus}^{EM} . Topoclusters are considered massless pseudo-particles since there is no physical meaning in associating a mass to an object without having specified and validated an hypothesis on the nature of the particle that has originated the energy deposit. The protocluster coordinates η_{clus} and ϕ_{clus} are calculated as the sum of the coordinate of each single cluster constituent weighted with the distance from the barycentre of the cluster. Topocluster Local Cell Weighted (LCW) calibrations aim to fully reconstruct the calorimeter signal on its appropriate hadronic or EM scale so that a final classification is possible. After the calibration, it is possible to associate the EM probability of the cluster $\mathcal{P}_{clus}^{EM}$: if $\mathcal{P}_{clus}^{EM} > 0.5$ the topocluster is classified as EM, otherwise it is a hadronic cluster.

Particle Flow

The jet reconstruction with topoclusters represents a baseline that is robust to detector operation, but it is not fully exploiting ATLAS potential, as the information in the ID is not used at all. In fact, putting together the ID momentum resolution in Eq. ?? with the calorimeter energy resolution in Eq. 2.12, it is clear that at low p_T the tracker performs better, while at high p_T the calorimeter has a better resolution, as shown in Fig. 3.5. The granularity of the calorimeter does not allow resolving the angular separation of highly boosted particle decays; in contrast, the ID has an excellent angular resolution especially when evaluated at the perigee. However, as the tracks become more energetic, they bend less in the ID and their momentum resolution worsen; at the same time, their extrapolation to the calorimeter is increasingly precise. The ATLAS jets reconstructed with this strategy are called Particle Flow (PFlow) objects [128].

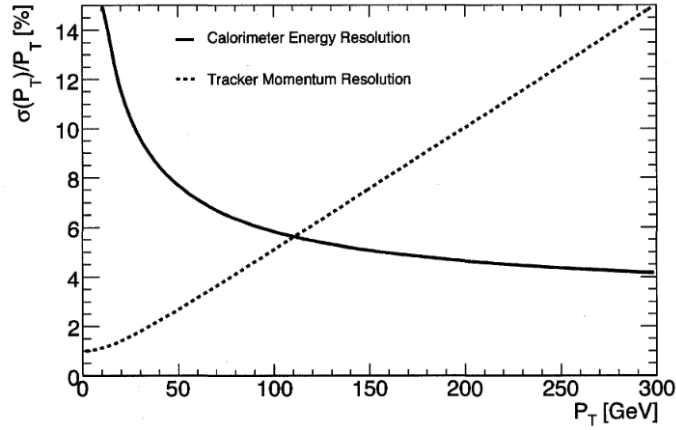


Figure 3.5. Comparison between momentum resolution of the ID and the calorimeter energy resolution. At low p_T the tracking performs better, at high p_T it is the calorimeter that has a better response. Figure from Ref. [129].

Tracks satisfying stringent quality criteria are selected and listed in p_T descent order; the tracks must have at least 9 hits in the silicon sub-detectors and no missing hits in the Pixel sub-detector. Tracks should also have $|\eta| < 2.5$ and $0.5 \text{ GeV} \leq p_T \leq 40 \text{ GeV}$, the upper momentum limit is set because tracks with higher p_T are often poorly isolated from nearby activity and the majority of tracks in low energy jets, the most affected by tracking resolution, have p_T in the given range. For each selected track, topoclusters are ranked according to a distance metric, which is rescaled based on their angular width and computed relative to the topocluster barycentre. Tracks are preliminary matched with topocluster if $E_{clus}/p_{trk} > 0.1$, where E_{clus} is the energy of the topocluster and p_{trk} is the track momentum; this cut rejects less than 1% of the correct particles. Among the topoclusters passing the cut, those with shorter rescaled distance from the tracks are matched. For each matched track-cluster pair, the algorithm estimates the energy that the track would have deposited, $\langle E_{dep} \rangle$, in the calorimeter based on its momentum. This is necessary to determine the amount of energy to subtract from the calorimeter measurement.

It is quite common for hadrons to deposit energy in cells that are reconstructed as more than one topocluster. After the cell-by-cell energy subtraction, the remnants of the topocluster are removed, if their energy is consistent with the expected fluctuations of a single particle's signal. The additional topoclusters consistent with the shower splitting hypothesis are removed from the list and the iteration is repeated.

Track Calo-Clusters

Track Calo-Clusters (TCC) are object built to reconstruct the jets with an approach which follows the PFlow procedure of combining both the tracking and calorimeter information, but improving the angular resolution of high p_T particles [130]. TCC's building blocks are tracks and topocluster, like for the PFlow. The track-topocluster matching consists of two steps: first the uncertainty of the track extrapolated to the calorimeter is compared to the width of the topocluster and, if larger, the track is discarded, otherwise it passes to the next step. The track is matched to the topocluster if its angular separation:

$$\Delta R < \sqrt{\sigma_{clus}^2 + \sigma_{trk}^2} \quad (3.6)$$

where σ_{clus} is the topocluster width and σ_{trk} is the track extrapolation uncertainty. The matching procedure uses any track in the ID, not removing lepton nor underlying event tracks, but removing PU. At this point a TCC candidate can be either a primary vertex track matched with a topocluster, called *combined TCC*, or a primary vertex track, called *charged TCC*, or a topocluster not matched with any track, called *neutral TCC*. Fig. 3.6 shows the different cases of TCC objects. After

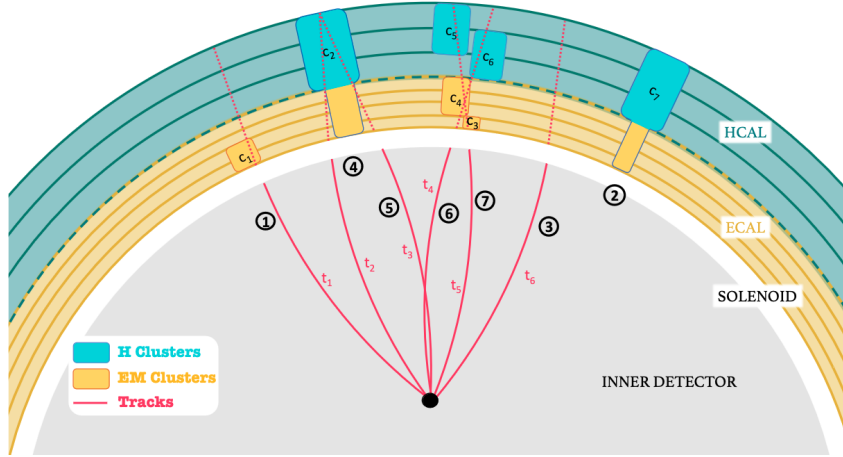


Figure 3.6. Possible cases of TCC candidates: candidates 1, 4, 5, 6 and 7 are combined TCC, candidate 3 is a charged TCC and candidate 2 is a neutral TCC. Figure from Ref. [130].

the matching, it is possible to build the TCC four-momentum. In case a single track has been matched with a single topocluster, the TCC four-momentum will have the track's η and ϕ and the topocluster's p_T . In case of charged or neutral

TCCs, the four-momentum will assume, respectively, the track's or topocluster's four-momentum. When multiple tracks are matched to multiple topoclusters, the directions are always taken from the tracks, while the p_T is assigned taking into consideration that the topocluster p_T is distributed among matched tracks in proportion to their p_T contributions.

Unified Flow Objects

So far, PFlow objects have been used in ATLAS Run 2 analyses for lower p_T small R jets, while TCC for boosted environments, with high p_T large R jets. Unified Flow Objects (UFO) have been designed to combine the excellent performance of PFlow objects at low p_T and the resolution power of TCCs at high p_T [131]. The UFO reconstruction process starts applying the standard PFlow algorithm; charged PFlow objects matched with PU are removed (Charged Hadron Subtraction or CHS). The remaining objects are divided into neutral and charged PFlows. Standard PU mitigation algorithms, like Constituent Subtraction and SoftKiller, are applied to the neutral PFlow objects. At this stage, the objects are passed to the standard TCC algorithm, but only tracks from the hard scattering vertices not used for PFlow subtraction are passed as input. The output objects of this process are UFO objects: they maximise the advantages of PFlow subtraction at low p_T , while optimising the TCC cluster splitting at high p_T . The integration of PU mitigation techniques helps maintain the stability of the jet measurements even in high PU environments.

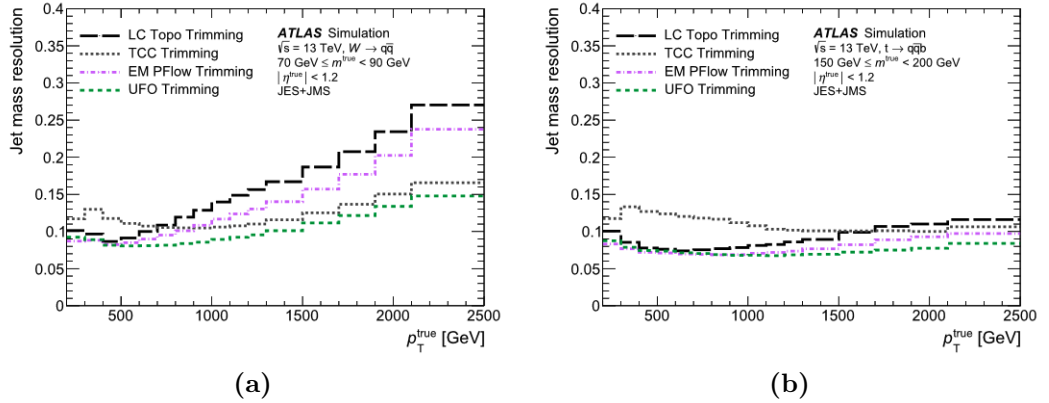


Figure 3.7. Jet mass resolution of the different objects, topoclusters (referred as “LC Topo”), TCCs, PFlows and UFOs, after trimming. In Fig. 3.7a the events come from a simulated W boson decaying into $q\bar{q}$, with true mass in the range $70 \text{ GeV} \leq m^{\text{true}} \leq 90 \text{ GeV}$; in Fig. 3.7b from a simulated t quark decaying into $q\bar{q}b$, with true mass in the range $150 \text{ GeV} \leq m^{\text{true}} \leq 200 \text{ GeV}$. Figures from Ref. [131].

3.4.4 Jet calibrations

After having built the jets, they have to be calibrated to account for several effects; the calibration procedures are slightly different for small and large R jets. In general

it is possible to distinguish two main parts of the calibration process: absolute MC-based and *in situ* calibrations [132, 133, 134]. Absolute MC-based calibrations aim to restore the scale of reconstructed jets to particle level in energy, direction and mass (the latter only for large R jets). In situ calibrations correct the remaining data/MC differences by using well-measured reference objects, such as the Z boson.

Absolute MC-based calibrations

Given a quantity X , absolute MC-based calibrations are a list of procedures whose goal is to reconstruct jets for which:

$$\langle X_{reco}^{MC} \rangle = \langle X_{truth} \rangle \quad (3.7)$$

A correction in Jet Energy Response (JER) and η is applied to small R jets [132]. The calibration starts by building the jet energy response $\mathcal{R}$, defined as a Gaussian with mean E_{reco}/E_{true} . The η correction, needed especially for the end-cap and transition regions, consists in the subtraction of $\eta_{bias} = \langle \eta_{reco} - \eta_{true} \rangle$; η_{bias} is determined by an iterative fit to a Gaussian function and it is applied jet-by-jet. There are many factors that may contribute to the JER, such as the distribution of energy deposits across different calorimeter layers and whether a gluon or a quark initiated the jet. This calibration step, known as the Global Sequential Calibration, applies corrections sequentially for different observables to improve the resolution and reduce uncertainties. For Run 3 a Neural Network is used for this task. Large R

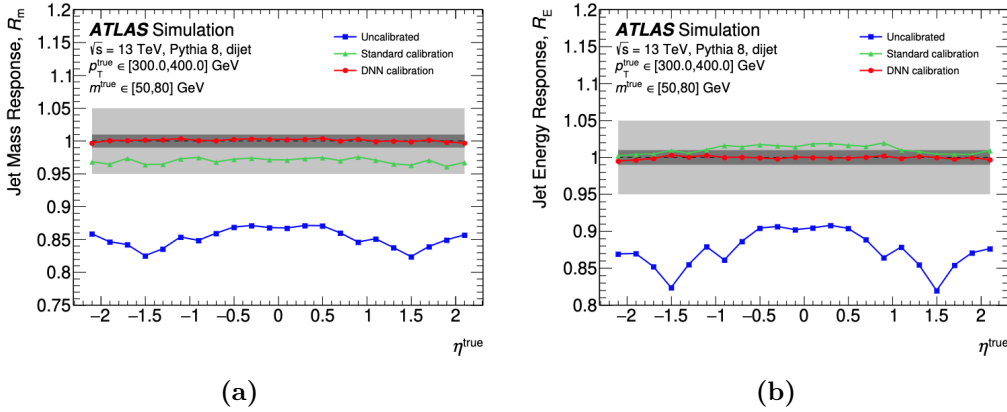


Figure 3.8. The simultaneous JER (Fig. 3.8a) and jet mass resolution (Fig. 3.8b) for large R jets obtained with the help of a Deep Neural Networks. Figures from Ref. [134]

jets also calibrate the jet mass and a Deep Neural Networks (for details see Sec. 4.3) is used to calibrate simultaneously the energy and the mass and the results shown in Fig. 3.8a and 3.8b. The algorithm uses 21 jet variables divided into four categories: jet-level, substructure, detector-level and event level variables. The large R jet η calibration is analogous to the one described for small R jets [134].

In situ calibrations

In situ calibrations aim to correct data jets so that they correspond to the already calibrated MC jets; this mean that given a jet quantity X :

$$\langle X_{reco}^{MC} \rangle = \langle X^{data} \rangle \quad (3.8)$$

In situ calibrations are tested on *candles*, that are processes for which the real value of a given quantity is known experimentally. As for MC-based calibrations, the small R jets are corrected for the energy and η , while large R jets are corrected for energy, η and mass. For each of the calibrations, the jet response is measured in data and in simulation as:

$$\mathcal{R}_{in\ situ}^{MC, data} = \left\langle \frac{X^{jet}}{X^{ref}} \right\rangle \quad (3.9)$$

so that the correction factor, defined as $c = \mathcal{R}_{in\ situ}^{MC} / \mathcal{R}_{in\ situ}^{data}$ can be applied directly on data.

Large R jets in situ calibrations are composed of two steps. The first is analogous to small R jets JES corrections by using measurements from Z/γ +jet or multijet events in the central region of the detector and vetoing small R jets. The Z boson is selected by its decay into leptons, typically muons, ensuring a clean and precise measurement; the photon are well reconstructed by themselves. The p_T of the Z boson or photon is compared to the p_T of the associated jet and any discrepancies are used to adjust the JES. In the second step the jet mass response is measured in lepton+jets $t\bar{t}$ production.

3.4.5 Jet substructure

Jets are generated by various processes but for HEP analyses it is important to disentangle different kind of jets. Jet substructure variables are constructed from low-level kinematic variables to make the distinction easier. In this Section, only the substructure variables actually used in this thesis will be described.

Jet mass

Since the mass of topocluster is set to zero, the total mass of the jet J can be calculated from its constituents as:

$$M_{jet}^2 = \left(\sum_{i \in J} E_i \right)^2 - \left(\sum_{i \in J} |\mathbf{p}_i| \right)^2 \quad (3.10)$$

N -subjettiness

The N -subjettiness describes to what degree the substructure of a given jet is compatible with being composed of N subjets, or to be N -prong [135]. For example, a W boson decaying into two quarks should be composed of two distinct, but not necessarily easy to resolve hard subjets, therefore W boson jets are said to be two-prong. A boosted QCD jet usually has been generated by a single hard parton,

thus is one-prong. Given a jet, first the jet is reclustered with a k_t algorithm that returns exactly N subjets with jet-axes¹ $a_1, \dots, a_N$. Then, the N -subjettiness, τ_N , can be calculated as:

$$\tau_N = \frac{\sum_i p_{T,i} \min\{\Delta R_{a_1,i}, \Delta R_{a_2,i}, \dots, \Delta R_{a_N,i}\}}{\sum_i R \cdot p_{T,i}} \quad (3.11)$$

where ΔR refers to the distance between the constituent i and the subjet axis, while R is the jet radius. The subjet axis can be defined in several ways, the most common of which is the winner-takes-all (WTA) k_t axes: the jet is re-clustered with the k_t algorithm combining two particles with the rapidity, azimuth and mass of the particle with larger transverse momentum and final p_T equals to the sum of the two's p_T . Jets with $\tau_N \approx 0$ have N or fewer subjets, since all the radiation is aligned with the candidate subjet direction. Usually, it is preferred to consider the ratio $\tau_{N+1,N} = \tau_{N+1}/\tau_N$ to better discriminate the jet prongness.

Energy Correlation Functions

Energy Correlation Functions (ECF) are similar to N -subjettiness, aiming to describe a jet substructure without requiring the selection of N reference axes [136]. Given a jet J the ECFs are defined as:

$$\begin{aligned} ECF_0(\beta) &= 1 \\ ECF_1(\beta) &= \sum_{i \in J} p_{T,i} \\ ECF_2(\beta) &= \sum_{i < j \in J} p_{T,i} p_{T,j} (\Delta R_{ij})^\beta \\ ECF_3(\beta) &= \sum_{i < j < k \in J} p_{T,i} p_{T,j} p_{T,k} (\Delta R_{ij} \Delta R_{ik} \Delta R_{jk})^\beta \end{aligned} \quad (3.12)$$

where the parameter β gives weight to the angular separation of the jet constituents and it is typically set to 1. By combining the ECFs, it is possible to build dimensionless variables, the most discriminating for two-prong jets is D_2 calculated as:

$$D_2(\beta) = \left[\frac{ECF_1(\beta)}{ECF_2(\beta)} \right]^3 \cdot ECF_3(\beta) \quad (3.13)$$

3.5 Taus

Tau leptons decaying into hadrons and a tau neutrino form a narrow jet and can be reconstructed independently from standard jets [137]. The tau reconstruction begins anti- k_t small R jets seeded with topoclusters; to qualify as tau candidates, the jets must satisfy $p_T \geq 10$ GeV and $|\eta| \leq 2.5$. A tau vertex association algorithm refines the candidate's vertex assignment by summing the p_T of tracks within $\Delta R < 0.2$ of the jet axis; the vertex with the highest contribution from these tracks is chosen

¹The determination of the jet-axes is not univocal, the most simple is the one here described.

as the tau vertex. Then, tracks are associated with the tau candidate if they lie within $\Delta R < 0.2$ of the tau direction and meet quality criteria that $p_T \geq 1$ GeV and have at least 7 hits in the Pixel and SCT sub-detectors. The identification of tau candidates relies on Boosted Decision Trees (for further details see Sec. 5.4.1) trained on simulated signal and background samples, to distinguish among one-prong and three-prong tau decays.

3.6 Muons

Muons are Minimum Ionising Particles (MIP) at LHC energies and therefore are expected to leave tracks in the ID and in the MS. The pattern recognition is performed independently in the two sub-detectors and later combined in different muon *types*. In the ID the track reconstruction is the same described in Sec. 3.2.1; in the MS tracks are built from patterns of hits, the segments, found in each muon chamber. The Hough transform is used to reconstruct pattern fragments from the hits in the MDTs through a straight-line fit. The fragments are selected using requirements on hit multiplicity and fit quality, in addition to a position matching. Segments are then used to form the tracks in a procedure which is seeded by fragments in the middle layers, where more hits are available, and extended to inner and outer layers. An algorithm is later performed to select the best aligned track, allowing also the sharing of segments among tracks due to pattern crossings. Once the tracks in the MS have been identified, a global χ^2 fit is performed, with the possibility to remove hits with large contribution to the sum.

Muon types

Depending on the different algorithms used to perform the muon reconstruction and identification it is possible to distinguish four different muon types, listed below.

- *Combined Muons (CM)*: the muon track is reconstructed from the combination of both ID and MS independent tracks. The most common approach, the *outside-in* approach, starts from MS muon candidates, opens a cone around the extrapolated tracks and searches for matching tracks in the ID. CM cover the region $|\eta| < 2.5$.
- *Segment-Tagged Muons (ST)*: low p_T muons which do not have enough energy to arrive up to the MS, or muons that fall in MS regions with low acceptance, may fail the CM reconstruction. ST muons consist of a full ID track but, when it comes to extrapolation to MS, an association with at least one local track segment is required.
- *Calorimeter-Tagged Muons (CT)*: sometimes low p_T muons or muons passing through sub-detector holes leave no track in the MS and cannot be reconstructed as ST muons. In these cases, they can be recovered by the CT algorithm which requires an ID track and an energy deposit in the calorimeters compatible with minimum ionising particles.

- *Extrapolated Muons (ME)*: standalone MS tracks are used for this type of muon, in addition to loose requirements on compatibility with IP. These muons are preferred in the $2.5 \leq |\eta| \leq 2.7$ region, which is not covered by the ID.

3.7 Missing Transverse Energy

Missing Transverse Energy (MET) serves as an experimental proxy for the transverse momentum of particles that go undetected in pp collisions [138]. It can be reconstructed in its vectorial or scalar form; the vector is calculated from the sum of $\mathbf{p}_T = (p_x, p_y)$ of all the objects reconstructed in the event, that is:

$$\mathbf{p}_T^{miss} = - \left(\sum_{\substack{\text{selected} \\ \text{electrons}}} \mathbf{p}_T^e + \sum_{\substack{\text{accepted} \\ \text{photons}}} \mathbf{p}_T^\gamma + \sum_{\substack{\text{accepted} \\ \tau\text{-leptons}}} \mathbf{p}_T^\tau + \sum_{\substack{\text{selected} \\ \mu}} \mathbf{p}_T^\mu + \sum_{\substack{\text{accepted} \\ \text{jets}}} \mathbf{p}_T^{jet} + \sum_{\substack{\text{unused} \\ \text{tracks}}} \mathbf{p}_T^{track} \right) \quad (3.14)$$

In some occasions, it is useful to deal with the scalar MET, which is simply the absolute value of the Eq. 3.14. MET reconstruction naturally depends on the WP and the algorithms of all reconstructed particles; it is also important to avoid objects double-counting, that is solved explicitly by the signal ambiguity resolution algorithm.

Chapter 4

Machine Learning

Machine Learning (ML) is the process of programming a system to improve a performance criterion using example data or past experience. ML algorithms are generally exploited to achieve results and solve problem traditional algorithms fail to cope with, due to their versatility and ability to handle unseen situations. In this chapter, an introduction to ML algorithms will be provided, starting from the definitions of *training* and all its steps, to an overview of different ML architectures.

4.1 History and definitions

Human brain is a biological device capable of process data coming from the environment and from the experience acquired over time, in order to refine its knowledge and to improve its own conditions; this process is called *learning*. To achieve this, our brain consumes only around 20 W [139]. It should be not surprising, then, that the inspirations for letting human-made devices, the machines, learn came from human brain. In 1943, years before the first transistors were manufactured, a neurophysiologist and a logician published an article that laid the foundations for ML [140]. In the article, McCulloch and Pitts theorised a formalism in which the activity of brain neurons can be explained in terms of logical operations and proposed that any logical function can be implemented by a network of neurons. Despite the first theoretical formulations of modern ML date back to eighty years ago, it is not until early 2000s that ML algorithms became powerful tools which can overtake traditional algorithms in not obvious tasks, as shown in Fig. 4.1. In fact, ML algorithms require significant computational power and are suited for a parallel computation typical of modern GPUs, which in fact started to flourish in the 2000s. Here I will provide the most commonly accepted definitions [142, 143, 144] that will be used throughout this thesis.

- *Learning*: improving with experience at some task with respect to a performance metric.
- *Artificial Intelligence (AI)*: is, broadly, any technique that can mimic human learning, thinking and decision making.

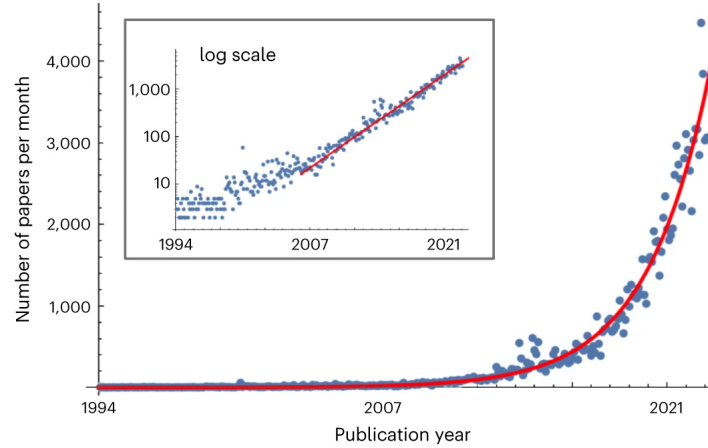


Figure 4.1. The rise of ML is evident from the number of papers published per month in the arXiv categories “AI” and “ML” that is growing exponentially. Figure from Ref. [141]

- *Machine Learning (ML)*: is the subset of AI that involves algorithms that learn to perform a task, improving their performance over time thanks to the use of past experience and/or vast dataset samples.
- *Deep Learning (DL)*: is a further subset of ML that focuses on neural networks with many layers, hence “deep”.

ML algorithms can be totally explained in terms of statistical and logical tools; the apparent unpredictability arises from the fact that they can be very deep and complex and it is not always trivial to understand why the model is behaving in a certain way. Mathematically speaking, a ML mode, given a dataset with input $\mathbf{x}$ and output $\mathbf{y}$, learns the function $\tilde{f}(\mathbf{x})$ so that:

$$\mathbf{x} \mapsto \tilde{f}(\mathbf{x}) \approx f(\mathbf{x}) = \mathbf{y} \quad (4.1)$$

where $f(\mathbf{x})$ is the unknown function that correctly assigns each input to its output. During the *training*, the model learns how to minimise the *error*, that measure the distance between $\tilde{f}(\mathbf{x})$ and $f(\mathbf{x})$.

4.2 Machine Learning categorisation

ML algorithms can perform different tasks that can be categorised into Classification, Regression, Clustering, Generation and Anomaly Detection; this list is non-exhaustive, but represents the majority of ML tasks. When the output $\mathbf{y}$ is discrete, the task is of Classification and the finite set of $\mathbf{y} = (y_1, \dots, y_N)$ represents the classes; cases in which the classes are only two are referred to as *Binary Classification* or *Concept Learning*. When $\mathbf{y}$ is continuous, the model is learning a Regression. If the task is to identify and group together elements of the same

kind in a heterogeneous dataset, that is a Clustering; if the elements of the same kind are rare with respect to the standard data, it is rather Anomaly Detection (for further details see Sec. 5.1). There are cases in which the task is to reproduce of the same kind of the input or, given an input that clarifies the instructions, create an appropriate output: this task is called Generative Learning.

ML algorithms can be subdivided into *supervised*, *weakly supervised* or *unsupervised* based on the way the training, that is the learning process, happens. In supervised trainings, the model learns not only from the huge amount of input data but also from the correct output, the *labels*. The algorithm learns by comparing its predictions against the true labels and adjusts its parameters to minimise the error resulting from this comparison. Unsupervised learning consists in training the model on the input data, without the possibility of knowing immediately the correct answer. Weakly supervised learning refers to all the methods adopting strategies that do not fully exploit the label information, as supervised methods, nor completely exclude the correct output, like unsupervised learning. There are many ways in which weakly supervised learning can be achieved, the most common passes to the model only a small subset of the training data is labeled, while the rest is unlabeled or labeled data that can contain noise, meaning that not always the output is correct. Additionally, there is another type of learning, that is *reinforcement learning*, where an *agent* learns to make decisions by interacting with an environment. The agent takes actions to maximise some notion of cumulative reward over time. Unlike supervised learning, where the model is trained on a fixed dataset, reinforcement learning involves a trial and error approach, where the agent receives feedback in the form of rewards or penalties based on the actions it takes. There are certain cases in which the input representation dictates the selection of specific architectures that can optimise the extraction of data features. For example, when the input consists of images, Convolutional Neural Networks (CNN) very often prove to be the most effective solution, as their architecture is particularly well-suited to capturing spatial hierarchies and local patterns within the data. Similarly, when dealing with time-sequence data, Recurrent Neural Networks (RNN) can be used due to their ability to maintain and utilise temporal dependencies across the sequence. In the case of textual data, Large Language Models (LLM) are typically employed, as they are capable of understanding and generating natural language with remarkable accuracy. Lastly, when the input is structured as graphs, Graph Neural Networks (GNN) are often the architecture of choice, as they can effectively model and analyse the relationships and interactions between nodes in a graph structure.

4.3 Deep Neural Networks

Deep Neural Networks (DNN), first introduced by Rosenblatt in 1958 [145], are multi-layer architectures based on simple building blocks, the *perceptrons*, that mimic the behaviour of human brain neurons understood at that time. The per-

ceptron can be mathematically formulated as:

$$z = \sum_{i=1}^n w_i x_i + b \quad (4.2)$$

$\mathbf{x} = (x_1, \dots, x_n)$ represents the input features or attributes of the data being processed; each input is associated with a corresponding weight $\mathbf{w} = (w_1, \dots, w_n)$; the bias b allows the neuron to shift the perceptron output. The perceptron output z , which is the weighted sum of the inputs and the bias, is then passed to a threshold function, called *activation function*, that introduces non-linearity in the model; the most simple activation is the step function, but others will be shown in Sec. 4.3.1. A visual representation of the perceptron is shown in Fig. 4.2. While the perceptron is a powerful tool for solving linearly separable problems, it fails in handling more complex, non-linear relationships within the data and this led to a temporary decline in interest in neural networks. In the 1980s, more sophisticated algorithms consisting in multiple layers of neurons, the Multi-Layer Perceptron (MLP), were developed to address the limitations and had the effect to revive the field. MLPs form the basis of modern DL, since Deep Neural Networks (DNN) models are models with many MLP layers.

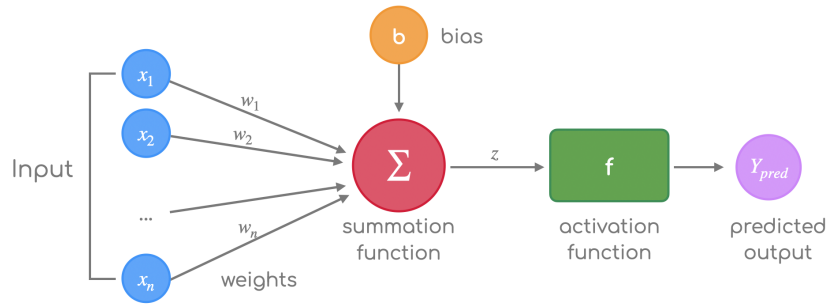


Figure 4.2. Sketch of a perceptron, the basic unit of DNNs.

4.3.1 Activation functions

Activation functions set the threshold to activate or disable a specific neuron, given the input. The choice of the activation function must be carefully handled depending on the task and on the input features.

- **Sigmoid function:** given the input to the neuron z , it maps any real-valued number to a range between 0 and 1, making it particularly useful for models where outputs are interpreted as probabilities in classification tasks. It is defined as:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (4.3)$$

- **Hyperbolic tangent function:** it maps input values to a range between -1 and 1, providing a symmetric output that is centred around zero. Mathemat-

ically, it can be formulated as:

$$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \quad (4.4)$$

- **Rectified Linear Unit function:** it has become one of the most popular activation functions since it introduces non-linearity while being computationally efficient to implement. It is defined as:

$$\text{ReLU}(z) = \max(0, z) \quad (4.5)$$

However, ReLU suffers of what is called the “dying ReLU” problem, where neurons can become inactive and stop learning if they consistently output zero.

- **Leaky ReLU and PReLU functions:** to address the ReLU limitations, variations such as the Leaky ReLU and the Parametric Rectified Linear Unit (PReLU) have been introduced, allowing for a small, non-zero gradient when the input is negative, thus preventing the neuron from becoming inactive. Given a small positive constant α , the Leaky ReLU is defined as:

$$\text{Leaky ReLU}(z) = \begin{cases} z & \text{if } z > 0 \\ \alpha z & \text{if } z \leq 0 \end{cases} \quad (4.6)$$

When the α parameter is not fixed but becomes a learnable hyperparameter of the model, the activation function is called PReLU.

- **Gaussian Error Linear Unit function:** is a smoother version of the ReLU function, given by the convolution with a Gaussian function centred in 0 [146]. Given a normal distribution $\mathcal{N}(0, 1)$, with mean 0 and standard deviation 1, the GELU is defined as:

$$\text{GELU}(z) = z \cdot \mathcal{N}(0, 1) \quad (4.7)$$

- **Softmax function:** Given the input of the i^{th} neuron z_i and K total number of classes, the softmax is primarily used for multi-class classification tasks and it is defined as:

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \quad (4.8)$$

4.3.2 Architecture of Deep Neural Networks

DNNs consist of stacked layers of neurons, where the output of each neuron in one layer serves as the input to the neurons in the subsequent layer. There are three primary types of layers: *input*, *hidden* and *output* layers. The input layer is the first layer of a DNN, where the raw data is introduced into the model. The output layer is the final layer of the network, responsible for producing the model’s predictions. Depending on the model’s task, the output layer has different designs, for example

in n -class classification, typically it consists in n neurons ending with softmax activation functions. The hidden layers are all the layers sandwiched between input and output layers and define the DNN's depth. Increasing the depth of a network generally enhances its ability to model more complex functions, but deeper networks are also more challenging to train.

Fully connected networks, also known as dense networks, are a type of NN architecture where each neuron in one layer is connected to every neuron in the subsequent layer. Information is allowed to flow freely between all parts of the network, but these models also tend to be computationally expensive and prone to overfitting. A common regularisation technique for DNNs is the *dropout*, which involves randomly “dropping out”, that means setting to zero, a proportion of the neurons during each training iteration. This prevents the network from becoming overly reliant on specific neurons and forces it to learn more robust features.

4.4 The training process

The training is the procedure through which the input data is passed to the model that extracts the relevant features to perform the assigned task. The training process is generally divided into three main phases: *training*, *validation* and *testing loops*. The training loop is a fundamental mechanism in which the model iteratively adjusts its parameters to minimise the error between its predictions and the actual outcomes. The validation loop, which occurs simultaneously with training, considers a separate portion of the data, known as the validation set, and uses it to monitor the model's performance. This step provides an early indication of how well the model is likely to generalise to unseen data. Training and validation typically involve multiple *epochs*, with each epoch representing a complete pass through the entire training and validation datasets. During each epoch, the model's weights are updated to gradually improve its accuracy. The training and validation loops continue until a predetermined criterion is met, such as a specific number of epochs, a convergence threshold or an early stopping condition based on validation performance. The testing phase is carried out after the training is completed, using a distinct test dataset that was not involved in either the training or validation processes. The aim of testing is to provide an unbiased evaluation of the model's final performance and its ability to make accurate predictions on new data.

4.4.1 Data preparation

Since ML algorithms learn from data, it is of fundamental importance that the dataset is representative and large “enough”. The size of the dataset ensures that the model has sufficient samples of all the possible outcomes it has to predict, while the representativity is needed otherwise the results obtained can be not consistent with the model performance when exposed to real data.

Once collected, the data typically undergoes *preprocessing* which consists of cleaning the data, selecting and engineering the relevant features and transforming the data into a format that is suitable for analysis, such as normalising numerical values or

encoding categorical variables. By normalising the data, each feature is scaled to a common range and this prevents poor performance or slow convergence during training. Given the input $\mathbf{x}$, the most common types of input normalisation are the following:

$$\begin{array}{ll}
 \text{min-max} & \mathbf{x} \rightarrow \mathbf{x}' = \frac{\mathbf{x} - \min(\mathbf{x})}{\max(\mathbf{x}) - \min(\mathbf{x})} \\
 \text{standardisation} & \mathbf{x} \rightarrow \mathbf{x}' = \frac{\mathbf{x} - \mu}{\sigma} \\
 \text{unit vector} & \mathbf{x} \rightarrow \mathbf{x}' = \frac{\mathbf{x}}{\|\mathbf{x}\|}
 \end{array} \tag{4.9}$$

With the min-max normalisation, the features are rescaled between 0 and 1; in the standardisation, the values are centred around 0 with a normal distribution; the unit vector normalisation does not affect the relations between data.

Sometimes, an additional step, *data augmentation*, is required when the available dataset contains a limited amount of data: through this technique it is possible to artificially expand the size and the diversity of a dataset. By applying various transformations to the existing data, new samples are generated, which help the model to generalise better and to become more robust to variations.

4.4.2 Splitting the dataset

Typically, the dataset is divided into three distinct subsets, reflecting the model training's different steps: the training set, the validation set and the test set. It is important that training, validation and test sets are disjoint sets of the original dataset: this ensures the logical independence of the three phases and the consistency of the results. A common split for the dataset is “80-10-10”: 80% is used for the training dataset, 10% for validation and 10% for testing. Sometimes the test set is not needed, as there exist additional datasets more similar to real data that can be used for this purpose.

It is common practise to divide training, validation and test datasets into smaller manageable subsets, known as *batches*, which are fed into the model sequentially during each iteration of training. Batches reduce the computational load and memory requirements, allow the model to learn more efficiently, help in stabilising the learning process and can lead to faster convergence.

4.4.3 Overtraining and underfitting

The ultimate goal of a ML algorithm is to achieve a model that generalises well to new, unseen data. However, two common pitfalls in the training process can hinder this goal: *overtraining* and *underfitting*.

Overtraining, also called overfitting, occurs when a model learns the training data too well so that it captures not only the underlying patterns but also the noise of the dataset. As a result, the model performs exceptionally well on the training data but fails to generalise to new data. Overtraining is often characterised by a model that has become overly complex, with too many parameters or layers that allow it to memorise the training data rather than learn the broader trends. To mitigate

overtraining, strategies that simplify the model, such as regularisation techniques, can be beneficial, as described in Sec. 4.4.4.

On the other hand, underfitting arises when a model is too simplistic and misses important patterns and relationships within the data. An underfit model performs poorly on both the training and validation datasets, as it is unable to accurately represent the target function. Underfitting can also occur if the model is not trained for enough epochs.

4.4.4 Training algorithms

During the training, ML networks iteratively minimise the *loss* function that quantifies the difference between the model's predictions and the true outputs. The optimisation is achieved by training algorithms that compute the gradient of the loss function, navigate its multi-dimensional surface and direct the model weights to move towards the minimum of the function. Additionally, regularisation techniques can be used to prevent overfitting by penalising overly complex models.

Loss functions

The choice of the loss function is critical as it directly influences the learning dynamics of the model, thus, the performance. Different tasks require different types of loss functions, each tailored to the specific nature of the problem.

For regression tasks, where the goal is to predict continuous values, the *Mean Squared Error* (MSE) is one of the most commonly used loss functions and is defined as:

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (\mathbf{y}_i - \hat{\mathbf{y}}_i)^2 \quad (4.10)$$

where $\mathbf{y}_i$ represents the vector of true values, $\hat{\mathbf{y}}_i$ the predicted value and N is the number of samples observed.

For classification tasks, where the objective is to assign inputs to discrete categories, the *Cross-Entropy* loss is widely used and is defined as:

$$\text{Cross-Entropy} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C \mathbf{y}_{i,c} \log(\hat{\mathbf{y}}_{i,c}) \quad (4.11)$$

where $\mathbf{y}_{i,c}$ is a binary indicator if class label c is the correct classification for sample i and assumes either the value 0 or 1; $\hat{\mathbf{y}}_{i,c}$ is the predicted probability of sample i belonging to class c and C is the number of classes.

Back-propagation and Gradient Descent

Gradient Descent (GD) is an optimisation algorithm used to find the local minimum of a differentiable function, in general, and of the loss function, in ML trainings by iteratively adjusting the model's parameters. It works applying efficiently the chain rule for computing iteratively the gradient of the function with respect to each

weight. The gradient is calculated backward, from the output layer to each of the preceding layers, therefore the procedure is called *back-propagation*. The gradient, then, indicates the direction and magnitude by which the weights should be adjusted to minimise the loss. The update rule for the weight θ is the following:

$$\theta_{t+1} = \theta - \eta \cdot \nabla_{\theta} L(\theta_t; \mathbf{y}, \hat{\mathbf{y}}) \quad (4.12)$$

where θ_t represents the parameter value at iteration t ; η is the *Learning Rate* (LR), a hyperparameter that controls the size of the steps taken towards the minimum and $\nabla_{\theta} L(\theta_t)$ is the gradient of the loss function $L(\theta)$ with respect to the weight θ . The choice of the LR is crucial and must be optimised: large values of LR could prevent the algorithm to reach the minimum or even make it diverge; on the contrary, small values can lead to a slow convergence and the gradient can get stuck in local minima, as depicted in Fig. 4.3.

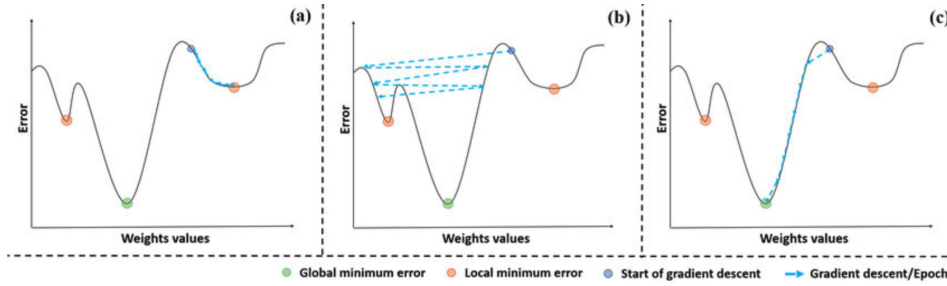


Figure 4.3. Three different cases of choice of LR. Case (a) shows a small LR value stuck in a local minimum; case (b) depicts the choice of a large LR that will never converge in the global minimum; case (c) represents the optimal LR. Figure from Ref. [147]

Stochastic Gradient Descent (SGD) is a variant of GD that updates the model's parameters using only a single training example, or a small subset of examples, at each iteration, rather than the entire dataset. In its basic form, SGD can be noisy and may result in a parameter update path that oscillates heavily. To address the oscillation problem, momentum can be introduced to the SGD update rule. Momentum helps to smooth the path by accumulating the gradients of previous steps and propagating it into the current update:

$$\begin{aligned} v_{t+1} &= \mu \cdot v_t + \eta \cdot \nabla_{\theta} L(\theta_t; \mathbf{y}, \hat{\mathbf{y}}) \\ \theta_{t+1} &= \theta_t - v_{t+1} \end{aligned} \quad (4.13)$$

v_t is the velocity vector and μ is the momentum coefficient, typically set to a value between 0 and 1.

Mini-batch Gradient Descent is an intermediate approach between the two, using only small, randomly selected subsets of the training data called a *mini-batch* to compute the gradient. Mini-batches are widely used especially when dealing with large datasets, case in which the loading of the entire dataset into memory for full-batch calculations is often impractical. Mini-batches lead also to an intrinsic parallelisation that can be exploited by modern hardware, such as GPUs. Since each mini-batch is a random sample of the data, the model does not see the exact same data distribution in every update step, which can prevent overfitting to the

training data.

The *Adaptive Moment Estimation* (Adam) optimiser is a more sophisticated variant of gradient descent, as it computes adaptive learning rates for each parameter by keeping track of both the first moment mean and the second moment uncentered variance of the gradients [148]. The update rules for Adam are:

$$\begin{aligned}
 m_t &= \beta_1 m_{t-1} + (1 - \beta_1) \cdot \nabla_{\theta} L(\theta_t; \mathbf{y}, \hat{\mathbf{y}}) \\
 v_t &= \beta_2 v_{t-1} + (1 - \beta_2) \cdot [\nabla_{\theta} L(\theta_t; \mathbf{y}, \hat{\mathbf{y}})]^2 \\
 \hat{m}_t &= \frac{m_t}{1 - \beta_1} \\
 \hat{v}_t &= \frac{v_t}{1 - \beta_2} \\
 \theta_{t+1} &= \theta_t - \eta \cdot \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}
 \end{aligned} \tag{4.14}$$

m_t and v_t are estimates of the first and second moments; β_1 and β_2 are decay rates for these moments and ϵ is a small constant added to prevent division by zero.

Regularisation techniques

To mitigate the risk of overtraining, regularisation techniques introduce additional constraints or modifications to the learning process, resulting in more robust models. *L1* and *L2* regularisation are among the most widely used techniques for adding penalties to the model's complexity. They are mathematically expressed as:

$$L_{L1} = L + \lambda \sum_{i=1}^N |\theta_i|; \quad L_{L2} = L + \lambda \sum_{i=1}^N \theta_i^2 \tag{4.15}$$

where L is the loss function, λ is the regularisation parameter that controls the strength of the penalty, θ_i represents the model parameters and N is the number of parameters. L1 regularisation encourages sparsity in the model by driving some of the coefficients to exactly zero; L2 tends to shrink the coefficients more evenly, preventing any one feature from dominating.

Early stopping is another powerful regularisation technique that monitors a performance metric and stops the training when this metric is no more improving for a predefined number of iterations. In fact, in the first epochs of a training the model improves its performance on both the training and validation sets. However, after a certain point, the model may begin to overfit the training data, leading to a decrease in validation performance and here is the moment in which early stopping intervenes.

Hyperparameter tuning

In a ML training, it is possible to distinguish between the *parameters* or weights, learned and tuned during the training by minimisation of the loss function, and the *hyperparameters* which are the external configurations that govern the training

process of a model. Hyperparameters include not only LR, regularisation coefficients, the number of layers or units in a NN but also the choice of a specific loss or regularisation technique rather than another. Given the potentially vast space of hyperparameter configurations, systematic approaches are required to identify the optimal set of values that yield the best model performance.

Grid search is very simple but effective, as it involves defining a discrete set of possible values for each hyperparameter and then exhaustively evaluating the model's performance across all possible combinations of these values. This method, however, can be computationally expensive, particularly when the number of hyperparameters and their potential values is large. In contrast to grid search, random search selects hyperparameter values randomly from a predefined range or distribution, allowing for more extensive exploration within the same computational budget.

4.4.5 Performance estimators

Performance estimators are typically derived from comparing the model's predictions with the true labels in a dataset. Depending on the nature of the problem, different metrics are employed to capture various aspects of the model's performance.

Model validation

Several performance metrics are commonly used in model validation, each offering insights into different aspects of the model's behaviour. *Accuracy* is used for binary classification tasks and can be expressed as:

$$\text{Accuracy} = \frac{\# \text{ correct predictions}}{\# \text{ total predictions}} = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.16)$$

TN/TP are the number of samples belonging to the class 0/1 correctly classified, respectively the *true negatives* and the *true positives*; FN/FP are the number of samples belonging to the class 0/1 wrongly identified, respectively the *false negatives* and the *false positives*. While accuracy provides a general sense of how well the model is performing, it can be misleading in cases of class imbalance, where the model might achieve high accuracy by predominantly predicting the majority class; in these cases metrics like *precision* and *recall* are often used.

Precision is particularly important in situations where the cost of false positives is high, while recall is crucial in contexts where missing a positive case has significant consequences. They are mathematically defined as:

$$\text{Precision} = \frac{TP}{TP + FP}; \quad \text{Recall} = \frac{TP}{TP + FN} \quad (4.17)$$

The *Receiver Operating Characteristic* (ROC) curve, shown in Fig. 4.4 plots the true positive rate, the recall, against the false positive rate, defined as $FPR = \frac{FP}{FP + TN}$ at various threshold settings. The *Area Under the Curve* (AUC) of the ROC curve quantifies the overall ability of the model to discriminate between positive and

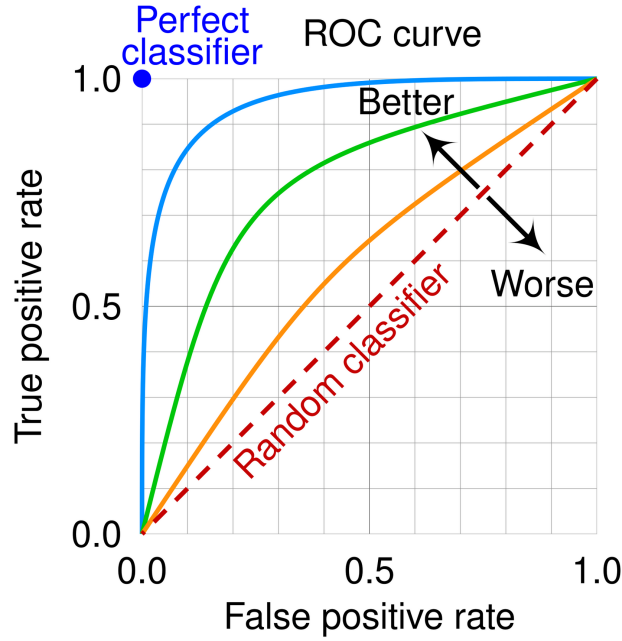


Figure 4.4. Plot of ROC curves representing different classifiers, from random to perfect one. Figure from Ref. [149]

negative classes:

$$\text{AUC} = \int_0^1 \text{ROC}(t) dt \quad (4.18)$$

A model with an AUC close to 1 indicates excellent discrimination, while an AUC near 0.5 suggests that the model's performance is no better than random guessing. Model validation is not limited to a single metric; sometimes other case-specific quantities can be calculated to monitor the performance.

Monitoring the training progress

While training a ML model, it is possible to monitor epoch-by-epoch performance through the *learning curve*. It consists in plotting the training and validation loss across epochs. During the early stages of training, both the training and validation losses should decrease as the model learns the underlying patterns in data. However, as training progresses, the model may begin to overfit the training data: at that point the training loss continues to decrease, while the validation loss starts to increase.

In addition to the learning curve, tracking the model's accuracy on both the training and validation datasets across epochs provides useful insights into the model's classification performance. A well-behaved model will show the accuracy increasing on both the training and validation sets, as the epochs increase. However, if the accuracy on the training set continues to rise while the validation accuracy plateaus or declines, it is another indication of overtraining.

The *Confusion Matrix* shows the number of correct and incorrect predictions for

each class of a classification problem. The matrix includes true positives, true negatives, false positives and false negatives for each class, offering a granular view of where the model is performing well and where it may be struggling. Additionally, the ROC curve itself can be used to evaluate the training and can be plotted for specific epochs in which the model should be monitored.

4.5 Machine Learning and High Energy Physics

A large number of events is produced at LHC due to the high luminosity and the 40 MHz of collision frequency. The choice of protons as colliding particles and the complexity of the sensors in the detectors to record the produced particles increases even more the amount of data stored by the experiments. The presence of high dimensionality and large volume data in HEP comes also from the quantum mechanical nature of collisions, that requires a statistical analysis approach reached by the creation of large MC datasets. HEP experiments, therefore, can be considered as generators of big data in all respects and this is a fundamental prerequisite for the use of ML algorithms [150, 151, 152, 153]. In fact, back to the ‘90s when LEP was active and in the early stages of LHC experiments constructions [154], CERN experiments were already employing simple ML architectures, such as Boosted Decision Trees, to be supported in the selection of interesting variables for the analysis. ML models can be exploited almost in every stage of a typical HEP analysis, from the MC simulation, to discrimination among variables, from trigger and tracking to flavour tagging algorithms. Fast generative models offer a promising alternative for traditional MC-based algorithms and ML optimisation techniques can help generators for global parameter optimisation. The development of tools that allow an easy and automatic implementation of ML algorithms in hardware or firmware based devices has led to a huge deployment of these models in the trigger pipelines, to predict physical quantities for track identification and momentum reconstruction. Object reconstruction and calibrations can benefit considerably from shallow networks such as DNNs based on higher-dimensional features, like in the works in [155, 156]. From Run 2, the ATLAS b -tagging algorithm is based on RNNs and DNNs [157]; for Run 3, instead, new GNN-based architectures have shown incredibly promising results improving the light and c -jet rejection by a factor ~ 2 [158, 159]. Traditional data analyses, which use a sequence of boolean decisions followed by a statistical analysis on the selected data, are progressively replaced by analyses employing one or even more ML algorithms to streamline the process, increase sensitivity and reduce human bias.

Every passing year, at CERN we are breaking record after record in terms of centre-of-mass energy reached or luminosity collected, thus of data produced and stored. Using ML techniques address the challenges posed by the ever-growing complexity and volume of data and enable more efficient, precise and automated analyses.

Chapter 5

Anomaly Detection R&D project

Anomaly Detection is a technique used to identify data instances that significantly deviate from the majority of data. This is analogous to what an experimental particle physicist does when analysing data: searching for extremely rare events, the *signals*, being them SM or BSM processes, among a large amount of not interesting events that constitute the *background*. The commonly adopted strategy is to apply some data selections and variable cuts to best discriminate signal from background. Recently, an increasing number of analyses have shifted towards ML-based event selection criteria, as these algorithms can learn underlying features and data structures that traditional, human-defined cuts may overlook. Anomaly Detection algorithms represent a useful tool for refining this process and, thanks to their generality, can also be employed in model-agnostic analyses.

This Chapter presents the research and development (R&D) studies conducted as a preliminary step to the ATLAS analysis, discussed in Chap. 6, to detect NP candidates in fully hadronic final states by using Anomaly Detection techniques. In fact, among all the objects detectable in collider experiments jets are the least obvious and most elusive. It is, therefore, possible that NP signals lie subtly within the jet substructures, remaining undetected by traditional analyses. For the initial part of the project described in this Chapter, a simple use case, based on the LHC Olympics 2020 dataset [160], is employed as a test bench. Several Anomaly Detection ML models, including Decision Trees, AutoEncoders, Transformers and Graph Neural Networks, have been developed and tested, yielding encouraging results.

5.1 Anomaly Detection

Anomaly Detection (AD) is the set of all the identification strategies to detect patterns or instances that deviate significantly from the norm (see Fig. 5.1). These deviations, known as *anomalies* or *outliers*, can indicate important, often rare, events that need closer investigation. AD can be applied in various contexts, including data security, bank fraud detection, data cleaning and equipment malfunctioning,

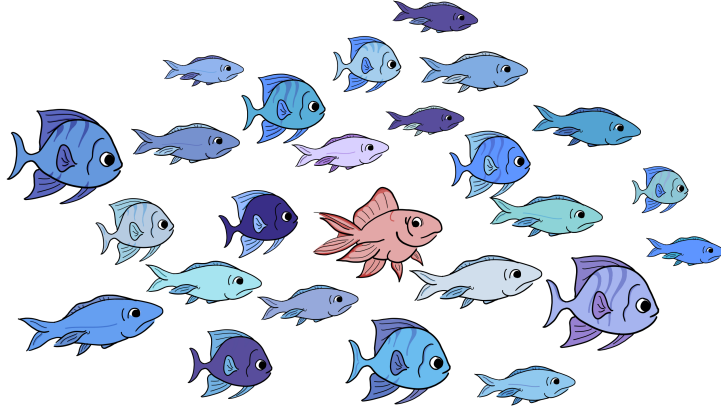


Figure 5.1. Pictorial representation of Anomaly Detection: AD searches for anomalies, the red fish, among a group of “standard” entities, the school of blue fish.

making it a powerful tool in cybersecurity, financial surveillance as well as health-care and risk management. The first AD procedures date back to the '70s [161], but it is only in recent years, with the exponential growth and use of ML and DL architectures, have AD algorithms achieved significant advancements [162]. A categorisation of AD algorithms can be made based on the ML technique employed. Supervised algorithms, such as Decision Trees or supervised NN, typically achieve higher detection rates than unsupervised techniques, such as AutoEncoders. The choice of a specific architecture strongly depends on the type of task, whether feature extraction or anomaly learning, on the dataset available, on the expected anomaly density, and should be discussed on a case-by-case basis.

5.1.1 Anomaly Detection for model-independent NP searches

Given the flexibility and results of these approaches, many analyses in HEP have started using AD to perform model-independent searches for NP. In this context, it is possible to categorise two types of algorithms: outlier detection and over-densities estimators [163]. The first focuses on identifying anomalous behaviours in specific regions of the kinematic phase space, such as the tails of distributions, where statistical fluctuations typically dominate and making it challenging to distinguish between genuine NP and mere statistical over-fluctuations. A general strategy consists of employing AutoEncoders (for further details see Sec. 5.4.3), as it has been done in [164, 165, 166, 167].

Over-densities estimators, by contrast, are the approaches designed to identify “bumps” in some kinematic distribution, such as the invariant mass shown in Fig. 5.2. In many HEP analyses, over-density estimation is achieved through weakly supervised trainings, where a signal is chosen a priori and injected into the unlabelled background events within the dataset; this method is known as Learning from Label Proportions [169]. Following this strategy, many models have been developed for NP searches: the Classification WithOut Labels (CWoLa) [170], the Tag N’Train [171] and Simulation Assisted Likelihood-free Anomaly Detection

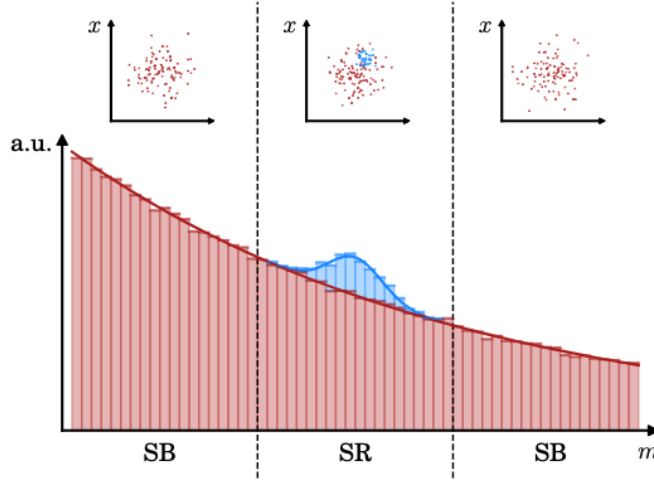


Figure 5.2. Sketch of a *bump hunt* search: the blue represents the contribution of a resonant signal to an exponentially falling background (in red) in the invariant mass distribution. In the sideband (SB) the signal is not present, while in the signal region (SR) both signal and background coexist. Through the use of some AD technique, the purity of the signal in the SR is increased and the significance of discovery is enhanced. Figure from Ref. [168].

(SALAD) [172] frameworks. Another studied approach is the New Physics Learning Machine (NPLM) [173] where a NN parametrises the alternative hypothesis by learning small deviations from a reference background model. The network output represents the likelihood ratio between data and reference, providing a direct test statistic for identifying discrepancies without relying on a specific signal model. Other strategies to enhance over density detection, such as CURTAINS [174], Classifying Anomalies THrough Outer Density Estimation (CATHODE) [168] and QUasi Anomalous Knowledge (QUAK) [175] frameworks, follow complementary paradigms based on density estimation and optimal transport techniques, to improve the sensitivity to localized anomalies in the phase space.

Although these methods have been tested on MC simulation, showing promising results that have led to multiple ATLAS and CMS searches, they also have certain limitations. Outlier detectors are usually difficult to interpret, as the learned anomaly scores are not directly linked to specific physical observables and often rely on complex, high-dimensional feature representations. Moreover, the intrinsically low number of anomalous events can make their training challenging. Weakly-supervised over-densities estimators, instead, are much more robust, but they need the presence of some signal in the training. In fact, even if the approach is easily generalisable to other signals, strictly speaking they are not model-independent. NPLM, on the other hand, are unsupervised methods, therefore completely model-agnostic, but it is challenging to identify bumps across multiple variable distributions simultaneously.

In this thesis, a different AD approach is proposed: the training is performed

only on background events and the model learns how to reconstruct the input from low-level variables. This is rather simple but effective, as the model will reconstruct differently signal and background events and, from the measurement of the distance, it is possible to distinguish background-likely from background-unlikely samples. The reconstruction performance is quantified through a variable called Anomaly Score (AS). In this approach, the loss of the ML model has been chosen as AS; in fact, the loss reflects how good the model has been trained to learn the underlying substructure of jets. The proposed approach is completely model-agnostic since the signal events are excluded from either the training or the validation sets, and used only in the test phase.

Before using this method for a model-independent analysis with ATLAS Run 3 data, it is needed to test it on benchmark datasets. The first test bench that has been chosen is the LHC Olympics 2020 dataset.

5.2 The LHC Olympics 2020 dataset

Given the importance of model-independent searches for NP and to encourage the HEP community to invest in developing powerful AD tools, the LHC community has created an open dataset in 2020, called the LHC Olympics 2020 dataset [176, 160], as a typical ML challenge. The 2020 edition was, in fact, the fifth, following those held in 2005, in February and August 2006 and in 2007. Although the underlying concept has remained the same over the years, the purpose and conditions of the challenge have changed significantly. The first three editions spanned only few days, during which experimentalists and theorists collaborated to solve some “black boxes” of data and find NP models unknown to the participants. These events took place few years *before* the start of LHC collisions, to prepare for the advent of LHC data taking and to facilitate communication within the HEP community. In the 2020 edition, three black boxes were provided as usual, but were released in different months, allowing for a more structured approach to the problem. The LHC Olympics 2020 dataset serves as a playground for physicists to develop powerful ML models and test their ability to detect NP in a model-agnostic manner.

The challenge includes one R&D and three Black Boxes (BB) datasets. The R&D dataset consists of 1 M of QCD multi-jet events (background) and 100 k coming from the BSM decay process of $Z' \rightarrow XY$ with $X, Y \rightarrow q\bar{q}$ (signal) (see Fig. 5.3). The masses of the Z' , X and Y resonances are 3.5 TeV, 500 GeV and 100 GeV, respectively, as shown by the peaks of Fig. 5.4. The events have been generated with PYTHIA 8.219 and DELPHES 3.4.1 with default settings, no simulation of PU or multiparton interaction. No specific detector experiment is assumed, therefore similar detector performance to both ATLAS and CMS geometry and reconstructions are set. Events are selected if at least one large-radius ($R = 1$) anti- k_t jet has been reconstructed having $p_T > 1.2$ TeV. All the particles reconstructed with DELPHES are considered massless and for each of them a vector containing the detector information (p_T, η, ϕ) is provided. The events are zero padded, in order to achieve a fixed shape containing up to 700 particles. The final array format consists of $(N_{\text{events}} = 1.1\text{M}, 2101)$, where 2101 comes from the 700 zero padded particles, each having 3 spatial coordinates, plus the last element referring to the event label. The

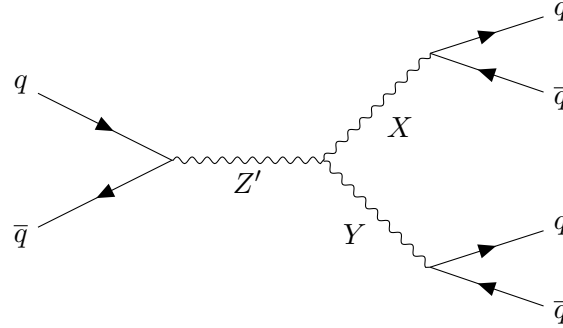


Figure 5.3. Feynman diagram of the signal of LHC Olympics 2020 R&D dataset.

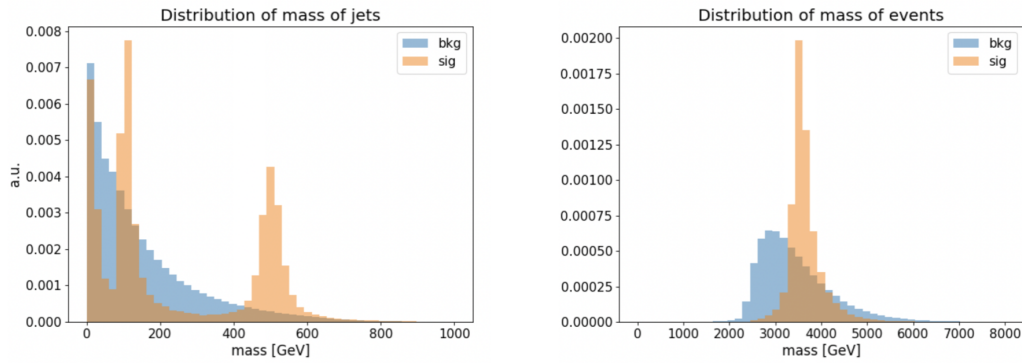


Figure 5.4. Distributions of the jet masses (on the left) and dijet invariant masses (on the right) for the LHC Olympics 2020 R&D dataset. The QCD background is represented in blue, while the $Z' \rightarrow XY$ 2-prong signal is in orange. From the jet masses, it is possible to notice the exponential falling background distribution; the signal, instead, shows the two peak of the X and Y resonances at 100 and 500 GeV, respectively, on top of the exponential falling distribution. In the invariant mass plot, the signal peaks at 3.5 TeV. In the plots, the background and signal distributions are normalised so that their total area is equal to 1.

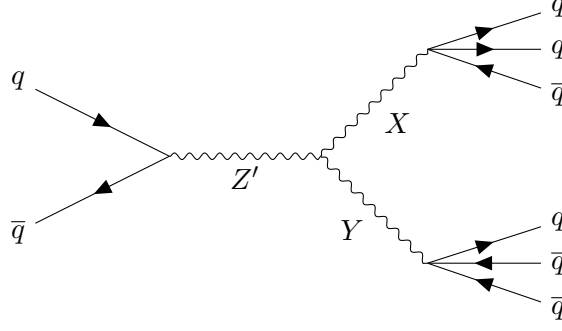


Figure 5.5. Feynman diagram of the additional 3-prong signal of LHC Olympics 2020 R&D dataset.

label can take the value of either 0, if the event is background, or 1, otherwise.

BB1 contains 1 M events, of which 834 are signal events with the same topology as the R&D dataset but with $m_{Z'} = 3.823$ TeV, $m_X = 732$ GeV and $m_Y = 378$ GeV, while the remaining events are background. The number of signal events has been chosen to make the local significance negligible, to test the ability of the developed ML models to detect the small amount of signals. Moreover, the DELPHES settings have been slightly modified to reflect more realistic inner detector measurements and calorimeter energy resolutions.

BB2 contains 1 M of background events only, but generated with HERWIG++ instead of PYTHIA.

The last BB3 is composed of 1 M of background and signal events. While the background is the usual QCD sample, the signal, instead, consists of a heavy resonance, the Kaluza-Klein (KK) graviton [177] with mass $m_G = 4.2$ TeV and two different decay modes. In the first case, the KK graviton decays into two gluons; in the other, it decays in one gluon and one new lighter neutral resonance R with mass $m_R = 2.217$ TeV, which subsequently decays into two gluons. The signal events are 3200 in total, 1200 of which come in the dijet topology and the others in trijets.

After the challenge has finished, given the success received among the community and the renovated interested, an additional R&D dataset was delivered. The new dataset contains 100 k signal events and no background, with similar signature to the R&D dataset apart that the jets resulting from the decay of the X and Y resonances are 3-prong (see Fig. 5.5).

5.3 Dataset preprocessing

The dataset [178, 179] was designed to accommodate various ML approach and was therefore as general as possible. In order to be used for the AD search, several preprocessing steps are required. Firstly, the proposed algorithms operate on single jets, therefore the 700 zero padded particles of each event should be reclustered into jets, using the FASTJET package [180]. Jets are constructed using the anti- k_t algorithm and $R = 1$; jets with $|\eta| > 2$ and $p_T < 30$ MeV are discarded. Only the first two jets, ordered by decreasing p_T , are considered for each event, in order to

focus on the dijet signature of the signal. At first glance, it may seem that imposing a dijet topology for NP reduces the generality of the search, making it no longer completely model-agnostic. This is undeniably true, but there are some caveats. First, the proposed approach works *jet-level*, so the number of jets considered does not alter the jet-level performance and can be changed at any time, if needed. Moreover, even in the most general model-independent cases, some assumptions about the expected appearance of NP must be made. In this case, a fully hadronic final state with two jets has been chosen, which maintains a broad search scope as no additional theoretical constraints have been imposed.

It is important to note that the jet constituents in the LHC Olympics 2020 datasets are not topoclusters or any other type of constituents used for jet reconstruction, as described in Sec. 3.4.3, but rather the particles produced directly in the simulation during the interaction with detector material. Since no detector is specified, the digitisation of the trackers or calorimeters is absent in the LHC Olympics 2020 dataset.

Other preprocessing steps include applying a transformation to the jets for removing potential correlations between the model’s performance and obvious kinematic discriminants, such as the jet mass. In fact, the ultimate goal of the search is to define an AS, select anomalous events and look for a bump in the dijet invariant mass distribution of the anomalies. If the AS is not correlated with the mass, it ensure the model-independence of the search. Then the low-level variables used by the models are arranged in the most appropriate format to be passed to the ML architectures and, as a last step, they are normalised.

5.3.1 The jet transformation

A desirable characteristic of the AD model is robustness, meaning that its performance should remain unaffected under certain circumstances. In the case of a training on QCD samples, there are some features the model should be robust against, such as the “size” of the jet but also its coordinates in space. The size of the jet can be parametrised using the dimensionless variable

$$\rho \equiv \frac{m_J}{p_T R} \quad (5.1)$$

with m_J and p_T being the mass and transverse momentum of the jet, respectively, and R reflects the spatial width of the jet. High ρ values correspond to jets with large internal structures, such as those originating from heavy particles; low ρ jets are typically more QCD-like and arise from light quarks or gluons [181]. Apart from the mass and the transverse momentum of the jet, it is important that the model does not have to learn any rotation or translation symmetry. Therefore, the model should be rotation and translation invariant. This means that two elements, where one is the roto-translation of the other in the input space, should be embedded at the same point in the feature space, as represented in Fig. 5.6. In this search, the NP the model should be sensitive to can appear anywhere inside the detector with no

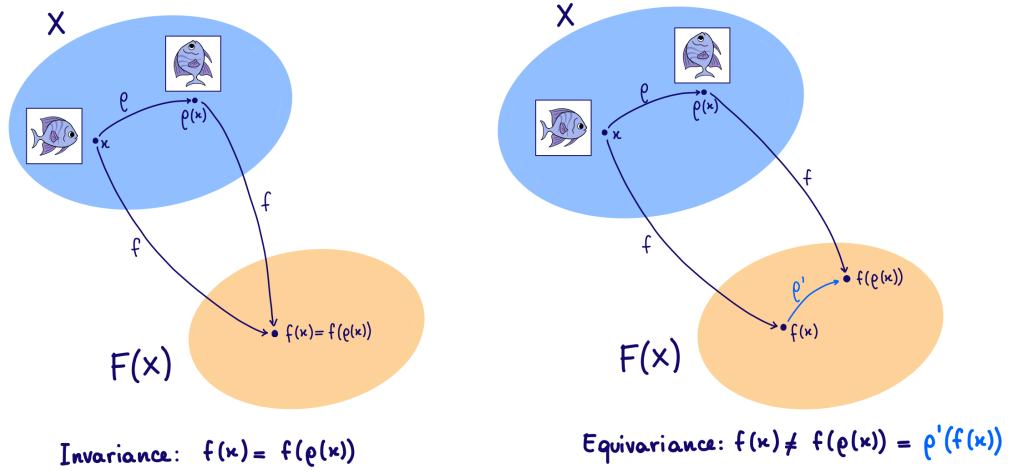


Figure 5.6. Graphical representation of the difference between invariance and equivariance. A model is invariant under a certain transformation ρ if given an element x in the input space X it maps both the points in the same representation $f(x)$ in the feature space. Instead, if the two representations in the feature space are different but it exists another transformation ρ' acting in the feature space so that $f(\rho(x)) = \rho'(f(x))$, then the model is equivariant.

preference for specific kinematic regions¹. There are two ways to make the network invariant: either the model is robust by itself, learning all these symmetries and dependencies and being taught not to depend on them, or it is helped by removing these symmetries from the dataset. The first approach requires significant efforts and statistical data for the model to capture the correlations while simultaneously not being influenced by them, or it can be achieved by directly modifying the model and selecting operations that are invariant under the given transformation. The latter is not only easier to implement, but also comes with other advantages. Therefore, a transformation has been applied to the jets of the dataset, as proposed in [182]. The transformation consists of the following four steps:

- **Mass rescaling:** the jet and constituents four-momenta are rescaled so that the final jet mass is fixed at a preassigned value, m_0 .
- **Lorentz boost:** the jet and constituents four-momenta undergo a Lorentz boost so that in the final reference frame the jet energy is fixed at a preassigned E_0 value; the Lorentz factor γ of such a boost can be found analytically.
- **Rotation around jet axis:** the jet, as well as its constituents, are rotated in ϕ and boosted in η so that the transformed jet axis lies in the origin of the η - ϕ plane; this step is required to make the dataset invariant under jet rotations and translations.

¹In many BSM resonance searches, the theoretical models give indications about where to expect the jet to lie in the detector, for example in the forward regions, based on their production mechanism.

- **Constituents rotation:** in order to be IR safe, a further rotation is applied *only* to all the constituents so that the hardest constituents has ($\eta_{const1} = 0$, $\phi_{const1} > 0$).

The fixed mass and energy values chosen for the dataset are $m_0 = 0.25$ GeV and $E_0 = 1$ GeV. For further details about the transformation, refer to App. B.

The result of having applied the transformation to the constituents of the LHC Olympics 2020 dataset is shown in Fig. 5.7. Fig. 5.7a and 5.7b represent not the

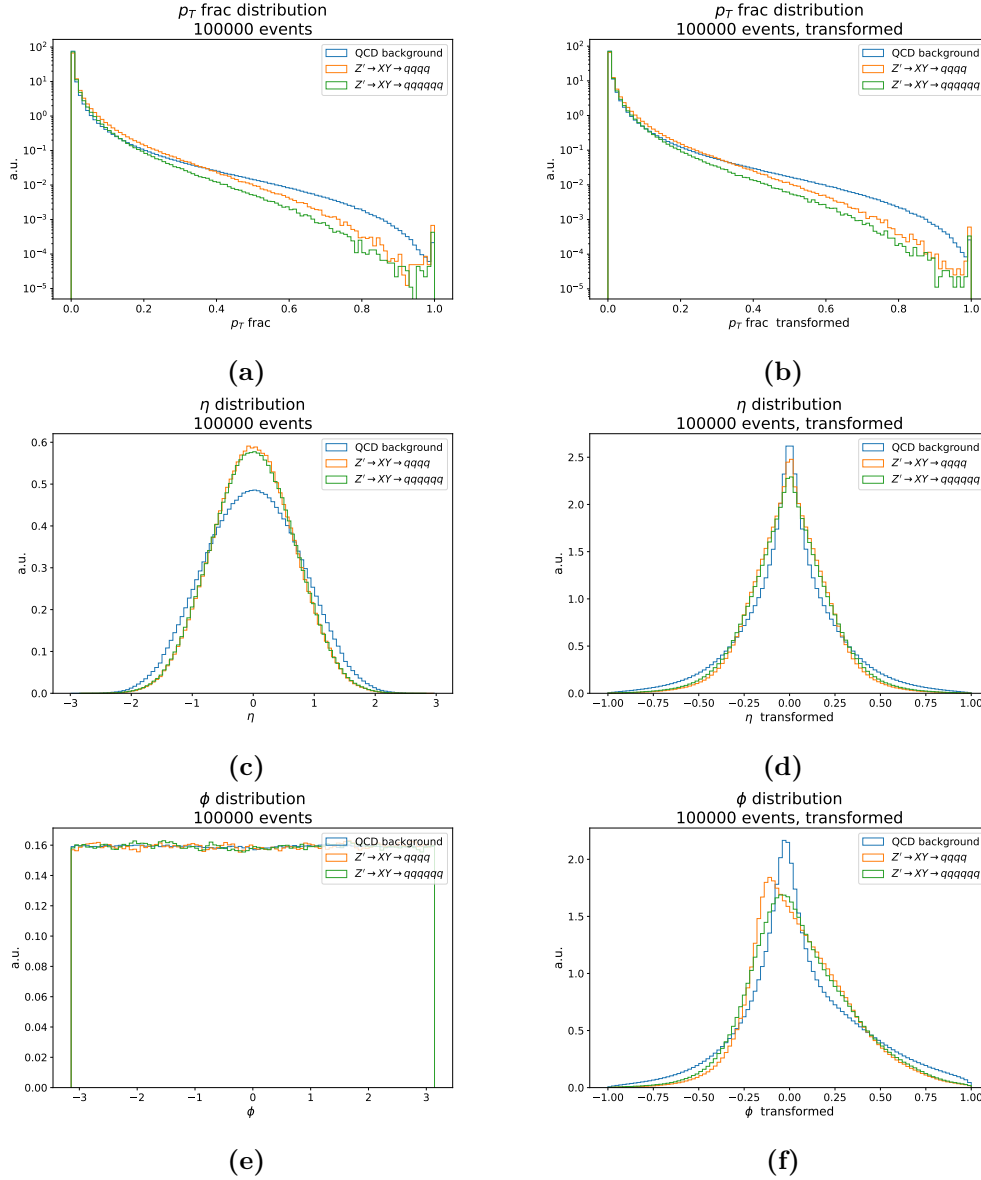


Figure 5.7. Kinematic constituent distributions for the QCD background (in blue), $Z' \rightarrow XY$ 2-prong (in orange) and 3-prong (in green) signatures. Fig. 5.7a, 5.7c and 5.7e show respectively the p_T frac, η and ϕ distributions *before* applying the transformation. Fig. 5.7b, 5.7d and 5.7f show the same distributions *after*. The plots have been obtained using only 100 k of the events per each sample.

p_T distribution but p_T frac, obtained by dividing the p_T of each constituent by that of the corresponding jet; p_T frac therefore spans the $[0, 1]$ range. The plots show similar p_T frac distributions before and after the transformation; the signal events have a slightly higher number of constituents with small p_T frac and fewer constituents at high p_T frac values compared to the background sample. The minimal changes observed in the p_T frac distributions can be attributed to the transformation acting simultaneously on the p_T components of the jet and its constituents during the mass rescaling and the Lorentz boost, leaving their ratio almost unchanged. The η and ϕ distributions, by contrast, change significantly before and after the transformation. Before, the pseudorapidity η (Fig. 5.7c) has a bell shape, with the signal distributions more peaked around $\eta = 0$ than the background; the ϕ angle (Fig. 5.7e), instead, remains flat in the $[-\pi, \pi]$ range for all the samples. After the transformation, the η distributions (Fig. 5.7d) appear shrunk in a $[-1, 1]$ range, but always peaking at $\eta = 0$. The ϕ distributions (Fig. 5.7f), instead, have radically changed their shape: spanning the $[-1, 1]$ range, they all have an asymmetric trend, with the background and the 3-prong signal samples more peaked around 0 and the 2-prong signal preferring negative values. These effects on η and ϕ result from the last two steps of the transformation: first, the jet axis is rotated to be centred in $(\eta = 0, \phi = 0)$; then, the constituents are further rotated so that the hardest acquires positive ϕ values and zero η . The pseudorapidity distribution, thus, remains peaked at $\eta = 0$, while ϕ takes an asymmetric shape, with preference of positive values. The direction of the hardest constituent approximately corresponds to that of the primary parton that has initiated the QCD jet; therefore, all other constituents will be located nearby. In the case of the 2-prong signal, the two quarks that have originated the jet correspond to the hardest and second-hardest constituents, respectively, and they exhibit a back-to-back topology. Therefore, during the final step of the transformation, if the first constituent acquires positive values, the second should have negative ϕ . In the 3-prong signal, the three original quarks have many constraints: they should follow the close dynamics of a three-body decay, the hardest constituent have to take a positive value, but the second- and third-hardest are linked to the first by the dynamics. The final effect is a combination of the three quark contributions, resulting in a distribution that includes element of the 1-prong QCD background, such as the peak around $\phi = 0$, and characteristics of the 2-prong signal, like the pronounced preference for positive values. These effects are also visible also in the η - ϕ plots in Fig. 5.8.

5.3.2 Graph and vectorial datasets creation

The dataset from the LHC Olympics 2020 challenge contains only low-level variables, which are the p_T , η and ϕ of the constituents of the jets in the event. This information can be used to construct high-level variables as input for the ML model or to directly provide the spatial and kinematic variables, allowing the model to determine how best to exploit them. We chose the second approach because, if NP lies within the jet substructure, we aim to avoid any human bias to the model, allowing the model to autonomously arrange the information to obtain the optimal discriminants. To best exploit the sparsity of data in the detector but also to stress

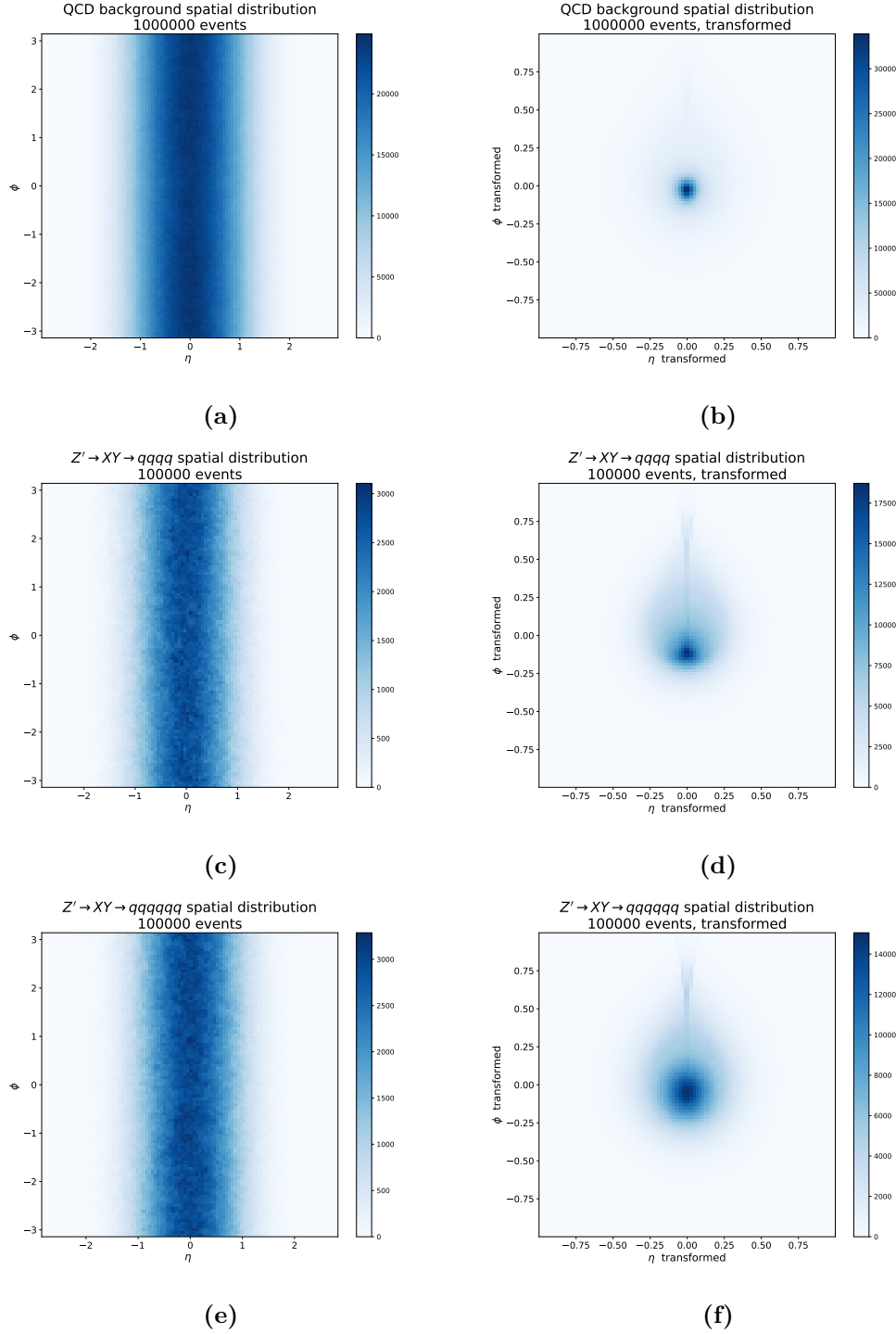


Figure 5.8. η - ϕ plots of the constituents before (on the left) and after the transformation (on the right) for the different samples. Fig. 5.8a and 5.8b are the plots for the QCD background; Fig. 5.8c and 5.8d for the 2-prong signal and Fig. 5.8e and 5.8f for the 3-prong signals. Before the transformation, the jets and their constituents are free to take any value in the η - ϕ plane. After the transformation, the background constituents peak around $(\eta = 0, \phi = 0)$, while the signals have a halo around the peak and a tail at $(\eta = 0, \phi > 0)$, as reported also in [182].

the importance of interconnections between constituents, a graph representation of the jet can be used.

Basics of Graph Theory

A graph is any abstract structure in which there are some elements that can be linked together. Mathematically, a graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ consists of a set of nodes $\mathcal{N}$

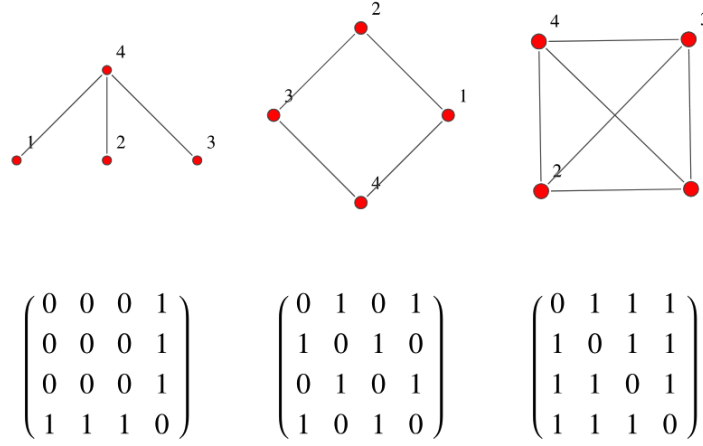


Figure 5.9. Example of graphs and their respective adjacency matrices. Figure from Ref. [183].

and a set of edges $\mathcal{E}$ between the nodes [184]. Given the nodes $u, v \in \mathcal{N}$, if the two are connected it means that there exists an edge $(u, v) \in \mathcal{E}$. Edges can be directional; in such cases, they are referred to as *directed* edges, otherwise, they are *undirected*. A graph is called *simple* if there is at most one edge between each pair of nodes and all edges are undirected. In order to represent conveniently the edges of simple graphs, it is possible to define the *adjacency matrix* $A \in \mathbb{R}^{N \times N}$ (see Fig. 5.9) where N is the number of nodes in the graph. The adjacency matrix is symmetric if the graph is undirected and it contains non-zero diagonal elements only if the self-loops are present. Typically, the elements of the adjacency matrix can take only binary values: 0 if there is no edge between the nodes, or 1 if an edge exists. In certain cases, the strength of the connection is relevant, and this results in *weighted edges*.

Key graph features include the *node degree*, the *average clustering coefficient* (ACC) and the *average shortest path length* (ASPL). The node degree is the number of edges connected to the i^{th} node and represents the tendency of the graph in having connections. The ACC quantifies the degree to which nodes tend to cluster together and can be defined as:

$$\overline{C} = \frac{1}{N} \sum_{i=1}^N C_i, \quad \text{where } C_i = \frac{\#e_{ij} : n_j \in \mathcal{N}_i}{k_i(k_i - 1)} \quad (5.2)$$

e_{ij} is the edge connecting the nodes i and j ; $\mathcal{N}_i$ is the neighbourhood of the node n_i , that is the set of all the nodes connected to n_i ; k_i is the number of neighbouring nodes the node n_i has. The ASPL, on the other hand, represents the average number of steps along the shortest paths for all the possible pairs of nodes, defined as:

$$\bar{L} = \frac{1}{N} \sum_{i=1}^N L_i, \quad \text{where } L_i = \frac{1}{N(N-1)} \sum_{i \neq j} \text{dist}(n_i, n_j) \quad (5.3)$$

These three quantities help characterise a graph dataset more effectively and provide insights for optimising the hyperparameters of ML algorithms working with graphs, Graph Neural Networks.

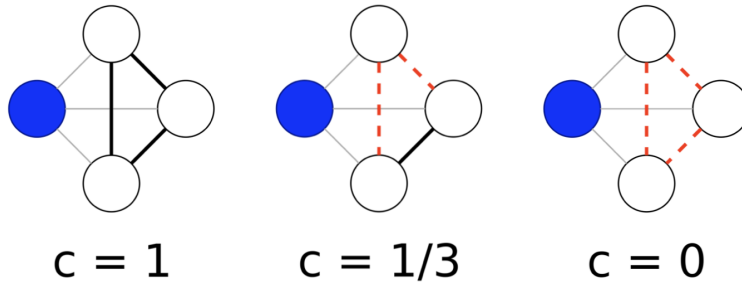


Figure 5.10. Example of average clustering coefficient of the node coloured in blue. For the graph on the left, the ACC is 1 because it is connected to 3 other nodes, all of them are connected to each other (solid black line). For the graph at the centre and on the right, instead, the other 3 nodes have respectively 1 and 0 connections (dashed red lines). Figure from Ref. [185].

The graph dataset

Jets are ideal objects to be represented as graphs, because they consist of a set of elements interconnected with one another. Moreover, graphs are particularly suited for situations where relatively little data, such as the energy deposit of the jet's constituents, are spread over a large environment, such as the whole calorimeter area. For these reasons, we have chosen to use of graph representation of the jet defined as follows:

- The nodes are the jet's constituents. In the LHC Olympics 2020 dataset, the constituents are exactly the particles generated in the simulation process; the node features are represented by the vector $(p_T \text{ frac}, \eta, \phi)_i$, where i is the node index.
- Two nodes are connected if a specific criterion based on the distance ΔR between constituents in the η - ϕ plane is met. The criterion is defined by the ΔR_c parameter and should be optimised based on the characteristics of the dataset. If two nodes i and j have a relative distance $\Delta R_{ij} \leq \Delta R_c$, then the two nodes are connected. Various ΔR_c have been tested, as shown in Fig. 5.12,

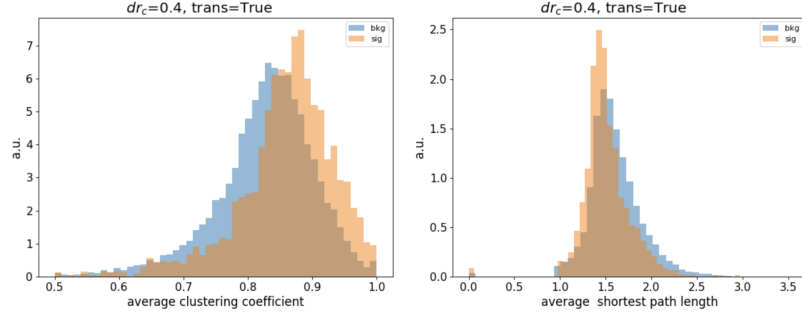


Figure 5.11. Distributions of the average clustering coefficient and aspl for the LHC Olympics 2020 R&D dataset, while using $\Delta R_c = 0.4$ and applying the transformation. The different colours represent the QCD background (in blue) and the $Z' \rightarrow XY$ signal (in orange).

but ultimately $\Delta R_c = 0.2$ and $\Delta R_c = 0.4$ were selected, depending on the architecture. In fact, the $\Delta R_c = 0.4$ choice is motivated by the observation that the value represents a good compromise between having a considerable fraction of nodes that have low node degree or are completely isolated, like in the $\Delta R_c = 0.2, 0.3$ cases, and fully connected graphs, like in the $\Delta R_c = 0.6, 0.8$ cases. If the graphs are fully connected, the purpose of the ΔR_c criterion becomes irrelevant. On the other hand, if there are many isolated nodes in the graphs, their kinematic content will not blend with those of the other constituents, not fully exploiting the graph connectivity characteristics. At the same time, the choice of $\Delta R_c = 0.4$ can be considered a physical bias: we observed better loss behaviours with $\Delta R_c = 0.2$ and decided to use this criterion from a certain point onwards.

The ACC and ASPL quantities of the graphs identified by $\Delta R_c = 0.4$ are shown in Fig. 5.11.

- The edges are weighted to ensure they are not equal. Many edge features have been tested, such as:

$$\begin{aligned} & \frac{1}{\Delta R_{ij} + \epsilon}, \quad \Delta R_{ij}, \quad p_{Ti} \cdot p_{Tj} \cdot \Delta R_{ij}, \\ & \frac{p_{Ti} \cdot p_{Tj}}{\Delta R_{ij} + \epsilon}, \quad |p_{Ti} - p_{Tj}| \cdot \Delta R_{ij}, \quad \frac{|p_{Ti} - p_{Tj}|}{\Delta R_{ij} + \epsilon} \end{aligned} \quad (5.4)$$

where ϵ is a set to 10^{-8} . No significant differences have been observed, but slightly better performance was achieved with the $\frac{1}{\Delta R_{ij} + \epsilon}$, that has become the default for dataset creation.

By default, no self-loops are introduced; however, some GNN models require them to propagate information among nodes. In such cases, the self-loop are added using the `dgl.add_self_loop` function of Deep Graph Library (DGL) [186].

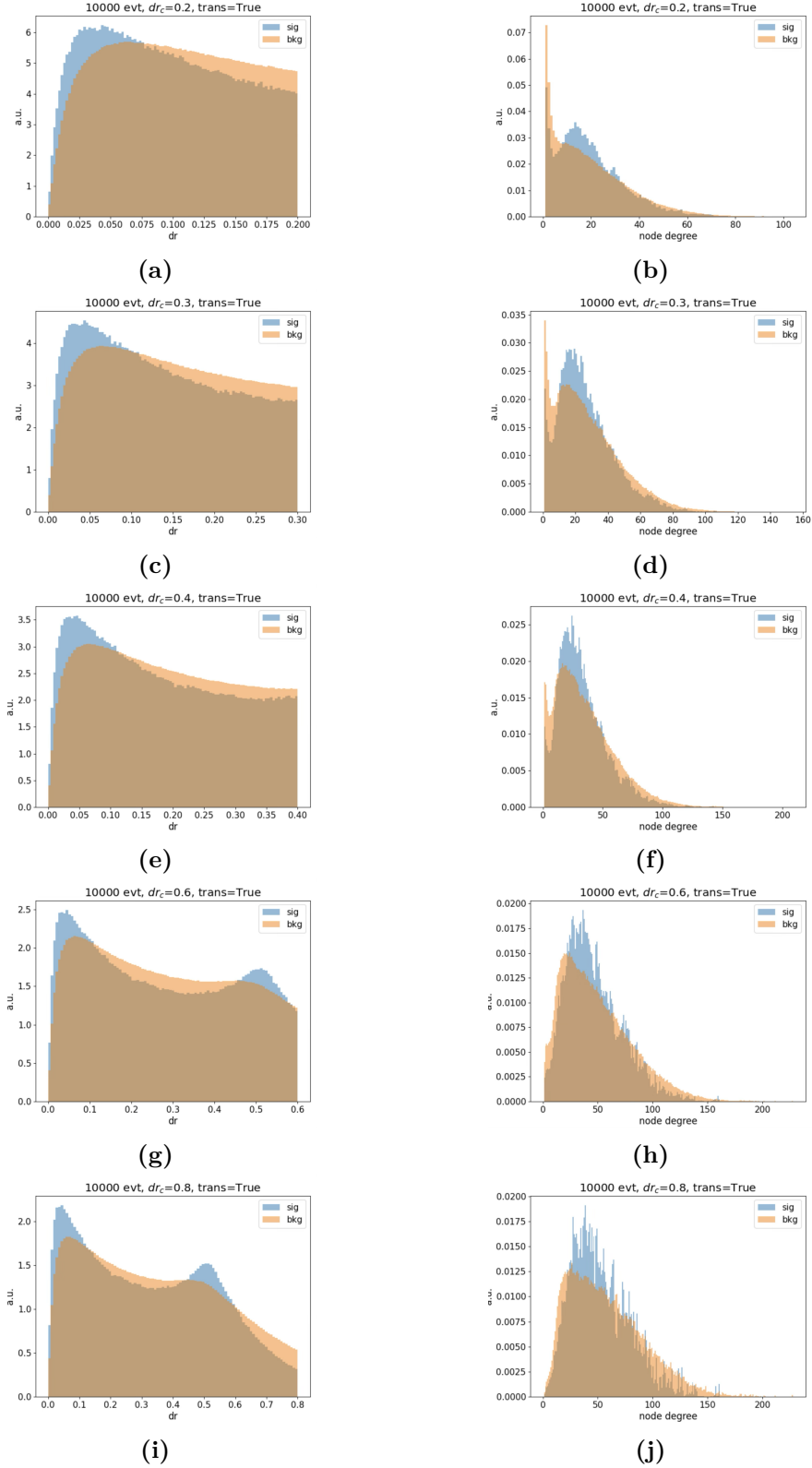


Figure 5.12. On the left, distributions of ΔR between all the constituents when different cuts are applied: $\Delta R_c = 0.2, 0.3, 0.4, 0.6, 0.8$. On the right, the corresponding node degree. The plots are obtained for the LHC Olympics 2020 R&D dataset, in blue the distributions for the QCD background, in orange for the $Z' \rightarrow XY$ signal. For the graph dataset, the $\Delta R_c = 0.2$ and $\Delta R_c = 0.4$ have been chosen as thresholds for connecting two constituents. The above distributions correspond to 10 k events and are normalised so that the total area is 1.

The vectorial dataset

In addition to the graph dataset, a standard vectorial representation has been created from the LHC Olympics 2020 dataset. Indeed, among the ML models developed to provide a baseline, some require a vector with a fixed shape. Moreover, the vectorial input is needed for Transformers that, thanks to the attention mechanism which will be discussed in details in Sec. 5.4.2, use a representation equivalent to a fully connected graph with the same number of nodes. In the LHC Olympics 2020 dataset, there are up to 700 jet constituents in the full event; however, after reclustering into jets, the maximum number of constituents per jet decreases to approximately 200. In Fig. 5.13, in fact, the distribution of the number of nodes of the graph dataset is shown: this corresponds also to the number of constituents in the jet. Therefore, the vectorial format of the dataset has a shape (250, 3) with

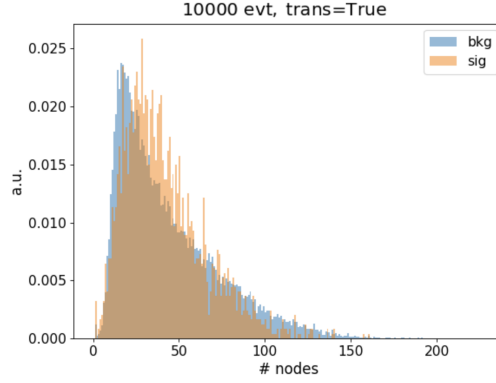


Figure 5.13. Distribution of the number of nodes in the LHC Olympics 2020 R&D dataset: in blue for the QCD and in orange for the signal events.

zero-padding: 250 is a conservative estimate to contain all jet particles, while 3 corresponds to the usual $(p_T \text{ frac}, \eta, \phi)$ low-level variable vector for each constituent.

Normalisation of the variables

Both the graph and vectorial datasets consist of the constituents' low-level variables $(p_T \text{ frac}, \eta, \phi)$. These variables have different ranges and must be renormalised, to ensure a smooth training process. Min-max normalisation was applied with poor results; therefore, a standardisation was implemented, obtaining better performance. The mean and standard deviation of the normalisation are calculated over the training dataset.

5.4 Model architectures

The task of searching for BSM resonances using a model-agnostic approach with the LHC Olympics 2020 dataset is challenging. However, transitioning to real ATLAS data, with detector reconstruction and the real QCD background distribution, will

further degrade performance. Although the detector response can be modelled and learnt by the ML algorithm, real QCD data may differ significantly from the MC dijet events, which can be simulated only at leading order in perturbative QCD. For these reasons, it is advisable to develop multiple ML architectures, to test and compare their performance, assessing the advantages and disadvantages of each approach. In particular, the models developed exploit Random Forests, Transformers, AutoEncoders and Graph Neural Networks. Some of the ML networks are trained in a supervised manner, while others undergo unsupervised training, so that it is possible to compare the performance of the two.

5.4.1 Decision Trees, Random Forests and Boosted Decision Trees

In 1963 Morgan and Sonquist proposed an approach to survey data focused on reducing predictive errors and operating sequentially, the *Decision Tree* [187]. Decision Trees are non-parametric models that can be applied to both classification and regression tasks. They operate by recursively splitting the dataset into subsets based on the value of input features, ultimately forming a tree-like structure where each internal node represents a decision based on an attribute, each branch corresponds to the outcome of that decision and each leaf node signifies a final class label or continuous value for regression tasks.

Combining the predictions of multiple Decision Trees creates Random Forests, where each tree is trained on a different bootstrap sample of the original dataset. The idea is that while individual trees might overfit to the noise in their specific samples, the ensemble of trees will collectively produce a more accurate and stable prediction. Random forests are controlled by several key parameters that influence their performance, with the most commonly used being the number of trees and the maximum depth. A larger number of trees generally leads to better performance, as the model benefits from more diverse predictions. However, beyond a certain point, adding more trees yields diminishing returns and increases computational cost. The maximum depth of each tree controls how deep the trees can grow. Deeper trees can model more complex patterns but may also be prone to overfitting.

Boosted Decision Trees (BDTs), unlike Random Forests that build multiple trees independently, create a sequence of trees where each subsequent tree attempts to correct the errors made by the previous ones [188]. The key idea behind boosting is to focus on the training examples that previous models have struggled with, thereby iteratively improving the model's performance.

A Random Forest was trained with the LHC Olympics 2020 R&D dataset in a supervised manner, in order to assess the baseline performance of the best traditional analysis based only on cuts on the kinematic variables of the constituents. The implementation relies on the `RandomForestClassifier` of the scikit-learn Python library [189]. We have chosen the default hyperparameter values, with 100 trees, Gini impurity criterion [190] and no restrictions on the maximum depth of the forest.

5.4.2 Transformers

Transformers are DL models that process input data in parallel, rather than sequentially and are based on the *attention* mechanism. Typically, a Transformer consists of several MultiHead Attention (MHA) blocks, sandwiched between some MLP at the beginning and at the end of the model. The attention mechanism enables the model to focus on different parts of the input sequence when making predictions [191]. The core operation within an attention layer is the computation of the Scaled Dot-Product Attention (Fig. 5.14a), where weights are determined by the relevance of each input element to the current processing element. This relevance is computed using three vectors: *query* Q , *key* K , and *value* V . For each token in the input sequence, the attention score for a given query-key pair is computed using the scaled dot-product:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \cdot V \quad (5.5)$$

d_k is the dimensionality of the key vectors, while QK^T represents the dot product between the query and the key. Dividing by $\sqrt{d_k}$ serves as a scaling factor to mitigate the effect of large dot-product values, which can cause the softmax function to produce extremely small gradients, slowing down the learning process. The idea behind the attention mechanism is the following. The query vector represents the current token for which the model is seeking information from other tokens in the sequence and serves as the point of reference in determining which other tokens to attend to. The key vector represents the token in the sequence that is being evaluated for relevance to the query and is compared with the query to compute attention scores. The value vector, instead, is the information content of the token that may be relevant to the query.

Due to the attention mechanism, features from the vectorial input are extracted similarly to a fully connected graph, where the key and query correspond to interactions between vector elements. NNs, other than Transformers, process the input data in a way that considers the order of arrival. In contrast, the attention mechanism provides permutation invariance, a characteristics typical of GNNs (see Sec. 5.4.4). Therefore, although the input of the Transformer is a vector, it is equivalent to a fully connected graph with the same number of nodes, where elements are connected by optimised edges, weights learnt through the key and query. In this case, the Transformer itself must determine the optimal representation for the edges of the graph.

For the AD in the search for BSM resonances, two Transformer architectures have been developed: one for classification and the other for an unsupervised approach. These two architectures, whose schematics are shown in Fig. 5.14c, differ only in the final MLP layer. In the supervised case, the MLP exits with a binary output, that is needed for the binary Cross-Entropy loss, obtained by flattening the last two dimensions of the data vector into one. In contrast, the AD Transformer produces an output with the same shape as the input, so that an MSE loss between input and output of the network can be used.

Taking inspiration from [192], an additional architecture has been developed, the *Graph-Transformer* or simply GTransformer. It consists of the same architecture shown in Fig. 5.14c, where the adjacency matrix A of the graph dataset is passed through the mask of the Scaled Dot-Product Attention (Fig. 5.14a):

$$\text{GAttention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}} + A\right) \cdot V \quad (5.6)$$

In this way, the model is forced to learn only the interconnections between nodes that are actually connected, as in the graph dataset. The idea of using a Transformer architecture as a Graph Neural Network is motivated by the practical challenge that creating a graph dataset is more complex than creating a vectorial dataset. Moreover, GNNs take longer to train compared to Transformers; therefore, having another model that mimics the graph response, but is faster and easier to handle provides a valuable backup solution.

The implementation of the (G)Transformers is based on an initial block of linear layers alternated with GELU activation functions (Eq. 4.7) and a final **dropout** layer, followed by 8 layers of the PyTorch `torch.nn.MultiheadAttention` [193], each with 8 heads. Exiting the MHA block, there is the last MLP which is responsible for the different output shapes for classification and AD. In the case of the GTransformers, the adjacency matrix of the graph is passed through the `attn_mask` attribute of the forward function of the layer.

5.4.3 AutoEncoders

AutoEncoders (AEs) are a class of NN architectures designed for unsupervised learning. They function by learning to compress input data into a compact representation, known as the *encoder*, and then reconstruct the original data from this compressed form, referred to as the *decoder*. The encoder consists of one or more layers of neurons that progressively reduce the dimensionality of the input data, up to a lower-dimensional representation known as the *latent space* or *bottleneck*. The size of the latent space is a crucial hyperparameter in AE design, as it determines the degree of compression and the amount of information retained, thus it limits the performance. The decoder reconstructs the original input data from the latent space representation and typically mirrors the architecture of the encoder in reverse, to gradually expand back to the original data dimensions.

For the search for NP using a model-agnostic approach, several AE architectures have been tested. Given the advantageous features of Transformers, such as efficient training and powerful extraction of relevant information thanks to the attention mechanism, we combined an encoder-decoder structure with MHA blocks. The results are non-traditional AEs, while maintaining the key architectural feature, the bottleneck.

The first Transformer + AE architecture, referred to as TAE1, is shown in Fig. 5.15. It consists of the same initial block as the Transformer, followed by three analogous AEs with bottleneck dimension of 32. The outputs of the AEs represent the query,

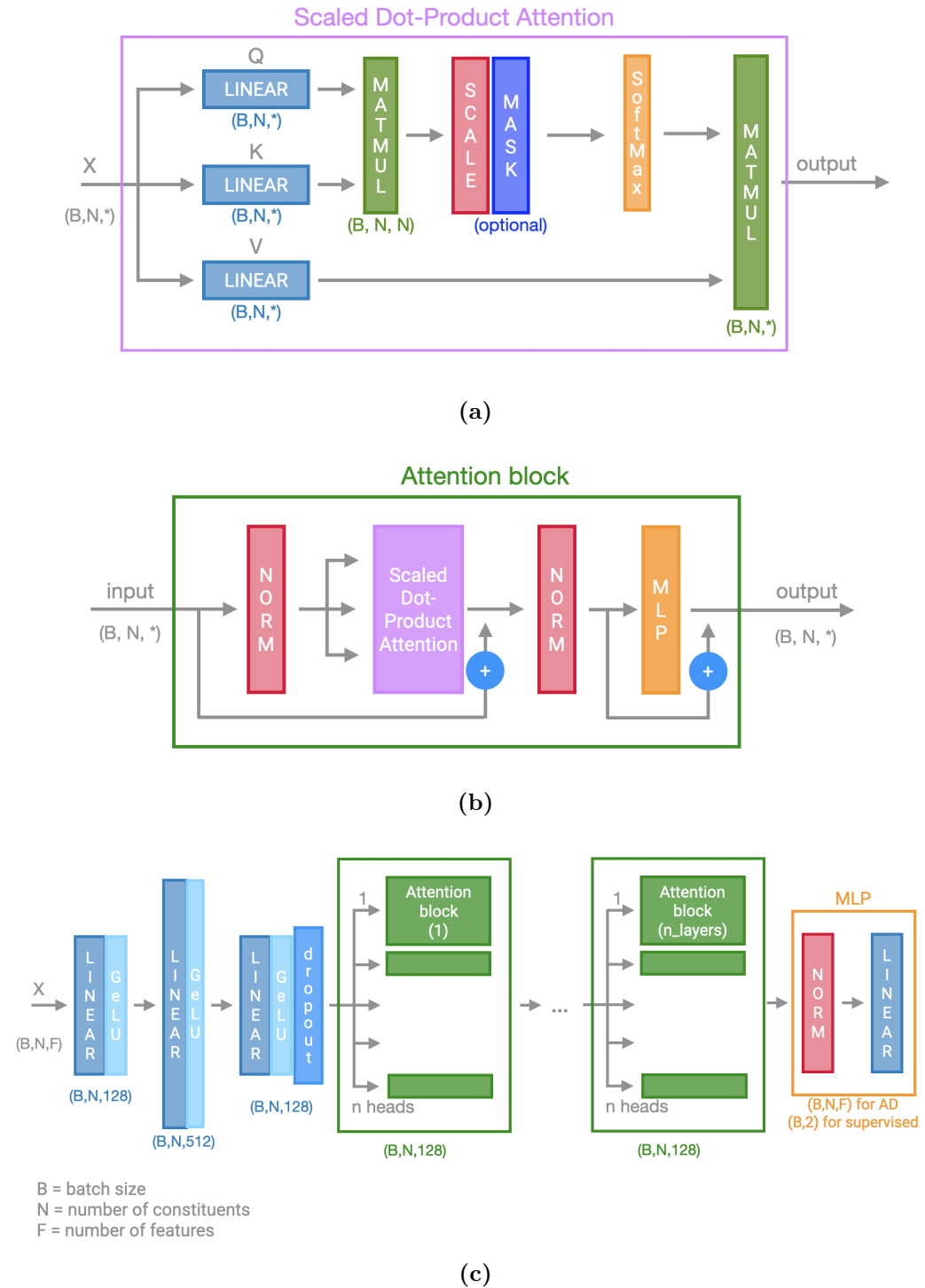


Figure 5.14. Schematics of the Transformer architecture. In Fig. 5.14a it is shown how the scaled dot-product attention is performed. The mask layer is optional and it is added in the architecture "GTransformer". At the bottom of each layer it is specified the dimension of the vector passing through, and it is evident that at the end of both the Scaled Dot-Product Attention and the Attention block in Fig. 5.14b, the dimension of the vector has remained the same. The self-attention is expressed by the "+" sign in Fig. 5.14b. The Transformer architecture used for this project is shown in Fig. 5.14c, where the last MLP returns a binary output in case of supervised trainings and an output with the same shape as the input in the unsupervised case.

key and value of the subsequent MHA block. Finally, the MHA output $\tilde{X}$ is compared with the AE input X' , as the two vectors share the same dimensionality. The second Transformer + AE architecture (TAE2), shown in Fig. 5.16, consists of an initial linear block followed by a sequence of MLP and MHA pairs with decreasing and subsequently increasing dimensions. The loss is again calculated between the AE input X' and the MHA output $\tilde{X}$.

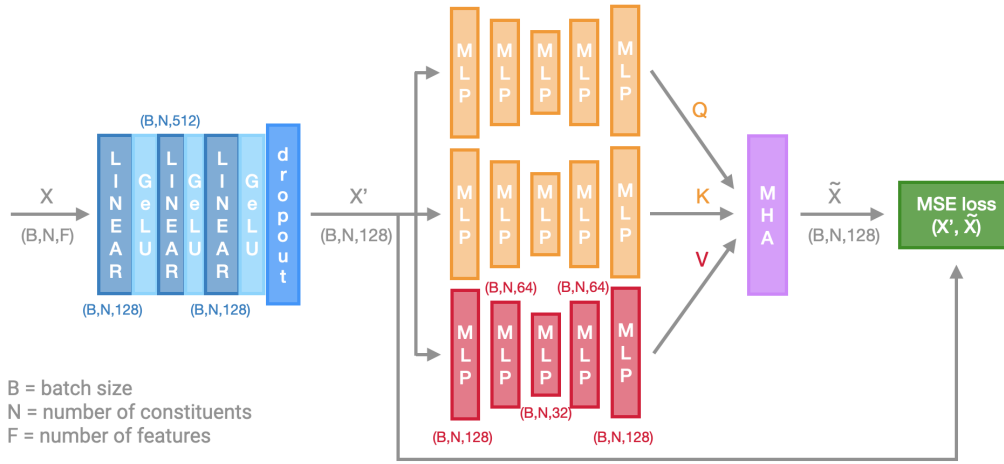


Figure 5.15. Schematics of the Transformer + AE model 1 (TAE1) architecture. The encoder-decoder part is repeated three times, to give a more insightful representation of the query, key and value. The loss is calculated between the input for the encoders, X' , and the output of the MHA, $\tilde{X}$.

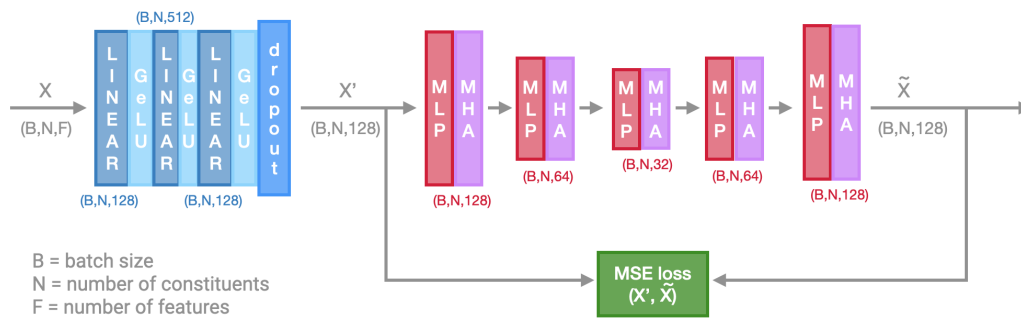


Figure 5.16. Schematics of the Transformer + AE model 2 (TAE2) architecture. The encoder-decoder structure is given by pairs of MLP and MHA which progressively decrease and then increase the size of the vector passing through. The loss is calculated between the input and output of this AE part, where the vectors X' and $\tilde{X}$ share the same shape.

5.4.4 Graph Neural Networks

Graph Neural Networks (GNN) include all ML models specifically designed to process and analyse data represented as graphs. Unlike traditional NN that operate on fixed-size inputs, such as images or sequences, GNN can handle irregular permutation-invariant data. In order to achieve this, GNNs propagate and aggregate information across graph nodes and their neighbouring nodes, through iterative *message passing* as shown in Fig. 5.17. Message passing involves three main stages: *message generation*, *aggregation* and *update*. For a given node u , message generation is computed at each layer k and embeds the information of all the neighbouring nodes $v \in \mathcal{N}(u)$ as follows:

$$m_{uv}^{(k)} = \psi^{(k)}(h_u^{(k-1)}, h_v^{(k-1)}, e_{uv}^{(k-1)}) \quad (5.7)$$

$h_u^{(k-1)}$ and $h_v^{(k-1)}$ are the embedding of the nodes u and v at layer $(k-1)$; $e_{uv}^{(k-1)}$ is the edge between the two nodes. The aggregation stage collects incoming messages for each node, often using permutation-invariant operations, such as summation, averaging or maximisation:

$$a_u^{(k)} = \oplus_{v \in \mathcal{N}(u)} m_{uv}^{(k)} \quad (5.8)$$

Finally, during the update stage, the aggregated messages are used to modify the node's feature vector via a transformation, such as an MLP layer:

$$h_u^{(k)} = \phi^{(k)}(h_u^{(k-1)}, a_u^{(k)}) \quad (5.9)$$

Message passing enables the network to propagate local information through the graph, facilitating the aggregation of both immediate and distant node features over multiple iterations. There are several variations of the learnable function h_u , which differentiate the GNN layers and will be discussed on a case-by-case basis. Edge

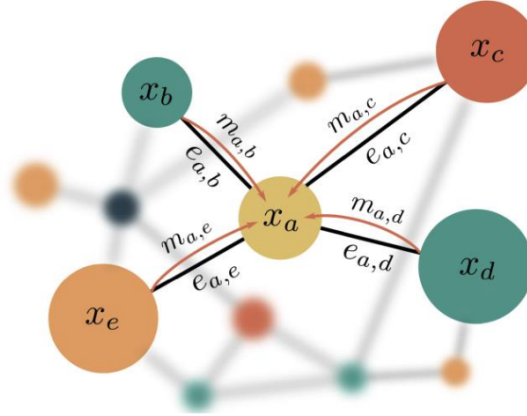


Figure 5.17. Visual representation of the message passing procedure. Figure from Ref. [194].

information can also be updated and embedded into a more useful representation.

This is achieved, usually, with a MLP layer applied to the edges, but variations can be developed. The last step in a GNN model is *graph pooling* [195]. Although not mandatory, it can be particularly important for tasks requiring graph-level predictions, where the aim is to map an entire graph into a single representation. Graph pooling can be categorised into two primary types: global pooling and hierarchical pooling. Global pooling methods, such as *mean*, *max* or *sum pooling*, aggregate all node features into a single vector, disregarding the graph structure in the process. In contrast, hierarchical pooling iteratively reduces the graph by selecting or clustering nodes based on certain criteria, such as their importance, connectivity or learned features.

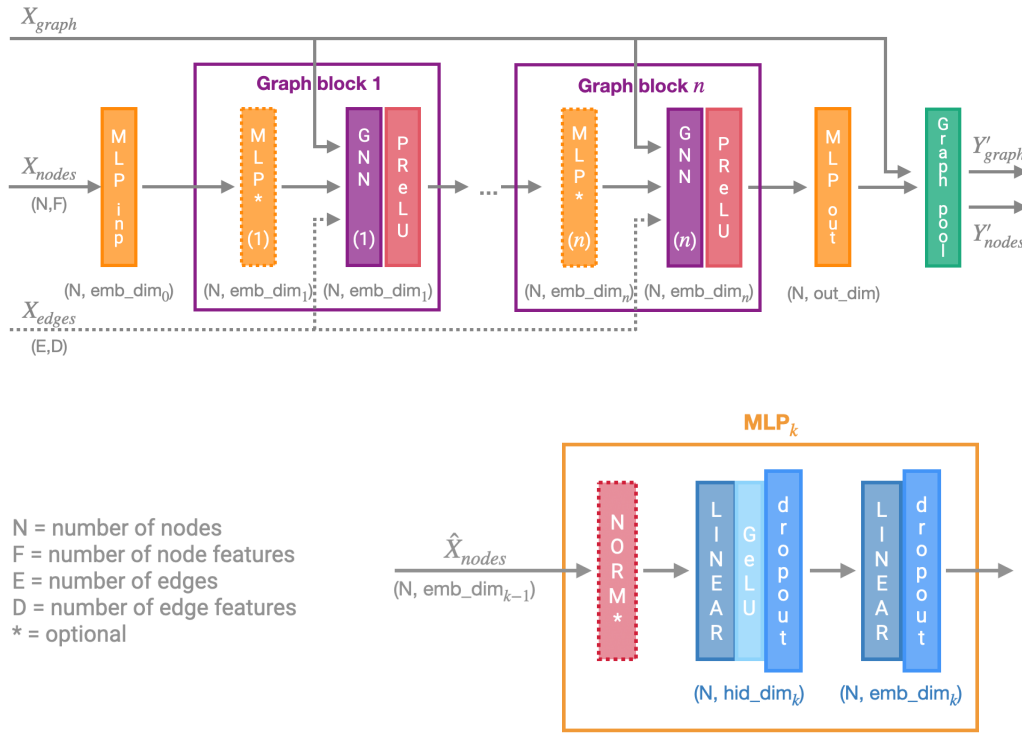


Figure 5.18. Schematics of the GNN architectures. First there is an initial MLP, followed by n graph blocks, final MLP and graph pooling layers. The dashed lines represent optional layers that can be enable or disabled. Not all the graph layers require the edge feature vector X_{edges} , but if needed it is passed through the corresponding dashed grey line.

The GNNs developed for the search for BSM resonances are based on the layers implemented in the DGL library [186]. In order to identify the GNN layers that performed best on the LHC Olympics 2020 dataset, a framework called *GNN4AD* was developed to test the available layers. GNN4AD is a modular environment where the model architecture is intentionally kept simple and training is conducted on a small number of events and for a limited number of epochs. Different graph layers can be tested easily by modifying a few lines of code. This approach is mo-

tivated by the fact that constructing the graph dataset and training GNN are time consuming, especially compared to vector-based datasets and models. GNN4AD also facilitates quick scanning of the model’s optimal hyperparameters, restricting the scan once the architecture is trained on more events. Naturally, training on smaller or larger datasets will yield different performance results and the best hyperparameters identified in one training may not necessarily be effective for training on the entire dataset. Nevertheless, training on smaller datasets can provide valuable insights into what to expect with the comprehensive training. Another caveat is that training on small datasets is more prone to overfitting, but this effect can be mitigated by limiting the number of epochs.

The architectures implemented in the GNN4AD environment share a similar structure, as illustrated in Fig. 5.18. The input of the model consists of the graph X_{graph} which contains the information about the graph structure, basically its adjacency matrix, along with the node feature vector X_{nodes} and the edge features vector X_{edges} . First, the node features pass through an initial MLP, then they undergo n graph blocks, a final MLP and finally through a graph pooling layer. All the MLPs used for these GNNs consist of an optional normalisation layer, a first linear block with GELU activation function and dropout layer, followed by a second linear block with a final dropout layer. Each graph layer includes an optional MLP, the proper graph layer and a PReLU activation function. Graph layers require not only the node features but also the graph structure X_{graph} to propagate the message passing; some graph layers additionally need the edge features X_{edges} . The graph layers used for GNN4AD are described hereafter.

GraphConv

GraphConv is a variant of CNN that operates directly on graphs. It applies convolution-like operations to symmetrically aggregate information from neighbouring nodes using the following formula:

$$h_v^{(k+1)} = \sigma \left(b^{(k)} \sum_{u \in \mathcal{N}(v) \cup \{v\}} \frac{e_{u,v}^{(k)} \cdot h_u^{(k)} \cdot W^{(k)}}{\sqrt{\deg(v) \cdot \deg(u)}} \right) \quad (5.10)$$

$b^{(k)}$ and $W^{(k)}$ are learnable weights and $\deg(u)$ represents the node degree; σ denotes the chosen activation function. The edge features $e_{u,v}^{(k)}$ can be explicitly passed to the function; if unavailable, they are substituted with 1 in the formula. This layer is implemented with the `dgl.nn.pytorch.conv.GraphConv` layer, is inspired by [196] and requires that all the nodes have the self-loop.

EdgeConv

EdgeConv is designed to focus on edge features, where the edges are dynamically constructed and updated by the layer itself. The constituents of jets can be easily associated with point clouds, the original dataset for which this layer was invented, and the edge representation described in Sec. 5.3.2 may not be the most expressive.

Allowing the model to learn and extract the edge features can be advantageous. The aggregation formula is given by:

$$h_v^{(k+1)} = \max_{u \in \mathcal{N}(v)} \left[\Theta \left(h_u^{(k)} - h_v^{(k)} \right) + \Phi \left(h_v^{(k)} \right) \right] \quad (5.11)$$

where Θ and Φ are linear layers. This layer is implemented with the `dgl.nn.pytorch.conv.EdgeConv` layer and inspired by [197]. Isolated nodes result in invalid layer output; therefore, it is common practice to add self-loops to each node in the graph.

Sample and aggreGatE, or SAGEConv

SAGEConv samples only a fixed number of neighbours per node. This is particularly effective when the graphs are very large and sparse, and the approach helps capture local graph structures and enables inductive learning. Each SAGE layer implements the following update rule:

$$h_v^{(k+1)} = \sigma \left(W \cdot \left[h_v^{(k)} \parallel \text{AGG} \{ h_u^{(k)}, \forall u \in \mathcal{N}(v) \} \right] \right) \quad (5.12)$$

σ and W are the activation function and the layer weights, respectively. The aggregation function, AGG, can be the mean, the max or another type of pooling of the neighbouring nodes' features, so that only a certain amount of information is retained. The output of the aggregation is then concatenated ($\parallel$) with the features of the node itself, creating a sort of self-attention mechanism. This layer is implemented with the `dgl.nn.pytorch.conv.SAGEConv` layer and takes inspiration from [198].

Graph Isomorphism Network (GIN)

GINConv is designed to maximise the expressive power of GNNs due to the approximation of the Weisfeiler-Lehman graph isomorphism test [199]. In fact, GIN aggregates the neighbour features through a sum operation and an additional MLP layer to introduce non-linearity. The update rule then becomes:

$$h_v^{(k+1)} = \text{MLP}_{\text{GIN}}^{(k)} \left((1 + \epsilon) \cdot h_v^{(k)} + \text{AGG} \{ e_{uv}^{(k)} \cdot h_u^{(k)}, u \in \mathcal{N}(v) \} \right) \quad (5.13)$$

If the edge features are not provided, the $e_{uv}^{(k)}$ of the formula are replaced by 1. The aggregation step can be implemented separately; therefore, even an MLP can be used as aggregator. This layer is implemented with the `dgl.nn.pytorch.conv.GINConv` layer and inspired by [200].

Graph Isomorphism Network with Edge features (GINE)

GINEConv extends GINConv by incorporating the edge features into the aggregation process, in the following way:

$$h_v^{(k+1)} = \text{MLP}_{\text{GIN}}^{(k)} \left((1 + \epsilon) \cdot h_v^{(k)} + \sum_{u \in \mathcal{N}(v)} \text{ReLU} \left(h_u^{(k)} + e_{uv}^{(k)} \right) \right) \quad (5.14)$$

This layer is implemented with the `dgl.nn.pytorch.conv.GINEConv` layer and is inspired by [201].

Graph ATtention network, or GATConv

GATConv includes the attention mechanism in the message passing phase, to assign varying levels of importance to the different neighbour nodes. The node features are then weighted by the coefficients found by the attention mechanism before being aggregated. Computationally, it corresponds to:

$$\begin{aligned} h_v^{(k+1)} &= \sum_{u \in \mathcal{N}(v)} \alpha_{vu} W^{(k)} h_u^{(k)}, \\ \alpha_{vu} &= \text{softmax} \left(\text{LeakyReLU} \left(\vec{a}^T [W h_v^{(k)} \| W h_j^{(k)}] \right) \right) \end{aligned} \quad (5.15)$$

α_{vu} is the attention coefficient and $\vec{a}$ is the learnable attention vector. This layer is implemented with the `dgl.nn.pytorch.conv.GATConv` layer and inspired by [202]. GAT requires all nodes to have at least a neighbour; therefore, a common practice to avoid invalid output from isolated nodes is to add the self-loop.

Edge Graph ATtention network, or EGATConv

EGATConv builds upon the GATConv layer by incorporating the edge features directly into the attention mechanism. The update rule, thus, is:

$$\begin{aligned} h_v^{(k+1)} &= \sigma \left(\sum_{u \in \mathcal{N}(v)} \alpha_{vu} \left(h_u^{(k)} \| e_{vu} \right) \right), \\ \alpha_{vu} &= \text{softmax} \left(\text{LeakyReLU} \left(\vec{a}^T [W \cdot h_v^{(k)} \| W \cdot h_u^{(k)} \| W_e \cdot e_{vu}^{(k)}] \right) \right) \end{aligned} \quad (5.16)$$

This layer is implemented with the `dgl.nn.pytorch.conv.EGATConv` layer and inspired by [203, 204].

The models used for GNN4AD have a default configuration that consists in 3 graph blocks. The MLP default includes a fixed embedding dimension of 128 and a fixed hidden dimension of 64. Different graph poolings, such as max and sum, have been tested; self-loops and edge weights have been enabled or disabled depending on the model compatibility. The output node feature dimension is 128. Certain models have additional hyperparameters to be optimised: for the SAGEConv layer, mean and gcN aggregators have been tested; for GAT and EGATConv, configurations with 2, 4, 8 and 16 heads have been tried.

After the GNN4AD trainings have pointed out the graph layers providing the best performance, bigger architectures have been developed. The most promising graph layers are EGAT and GIN, as better discussed in Sec. 5.6.1, and I will refer to the two architectures built upon these two layers as the EGAT and GIN models, by definition. The structure of the models are similar to that described for GNN4AD,

but with more graph layers or more MLPs sandwiched between the other layers. The number of the layers has been optimised taking into account the characteristics of the LHC Olympics 2020 dataset. In fact, the number of graph layers determines the extent of the neighbourhood or receptive field that each node can “see” and aggregate information from. Numerous graph layers are needed for large graphs, but tend to overfit in small graph datasets. In fact, as the number of layers increases, node representations tend to converge to similar values, making them indistinguishable and resulting in a over-smoothing performance. Since in the LHC Olympics 2020 datasets graphs have low average degree, fewer layers are sufficient. Therefore the EGAT model has 5 graph layers, while the GIN has only 3 graph layers, but with the addition of an MLP layer exiting each graph block, as aggregators. Each MLP used for these two architectures has embedding and hidden dimensions of 128. A max graph pooling have been applied, while the message passing aggregation is performed with the sum. The EGAT model requires also the number of attention heads, that has been set to 4. No dropout has been used for these models.

5.5 Training of the models

Two types of training, supervised and unsupervised, have been conducted for this project with specific configurations and strategies.

5.5.1 Classification trainings

For the classification, we decided to use balanced datasets; therefore, only 100 k QCD background events have been used in order to match the number of signal events available in the LHC Olympics 2020 R&D dataset. Signal and background events have been shuffled to create a uniform dataset. The first two dijets in each event have been considered, thereby doubling the dataset statistics. Out of the 400 k total jet-events, 320 k form the training dataset, 40 k are used for validation and the last 40 k for test, corresponding to the 80%, 10% and 10% of the available dataset, as shown in Fig. 5.19. The classifier models have been trained for 50-100 epochs; however, the loss often reached a plateau well before completing the training. The loss employed for the classification is a binary Cross-Entropy; therefore, all the classification models output two neurons: one expressing the probability of belonging to class 0, the background, and one to class 1, the signal.

5.5.2 Anomaly Detection trainings

For AD, the signal is used only during the test phase. The dataset is divided as follows: 800 k QCD events, thus 1.6 M jet-events, are used for training; 100 k QCD events, thus 200 k jet-events, for the validation; lastly, 100 k QCD and 100 k signal events, resulting in 400 k jet-events in total, for the test (see Fig. 5.19). The MSE loss described in Sec. 4.4.4 has been primarily used for the AD task with the Transformer and AE architectures, since the output dimension equals the in-

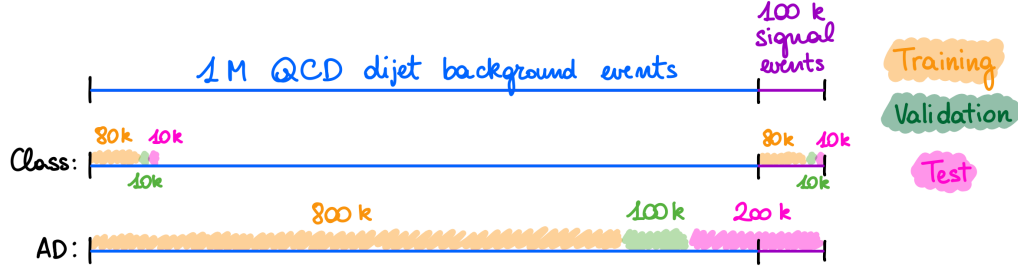


Figure 5.19. Sketch of the splitting of the LHC Olympics 2020 R&D dataset for the classification and AD tasks. The first line represents the total number of events in the dataset: 1 M QCD dijets, the background, and 100 k signal events. The other two lines show the subdivision into training (orange), validation (green) and test (magenta) sets for the classification and AD, respectively. Note that the numbers reported here are for the events; since for each event exactly two jets have been considered, the number of jet-events can be obtained by doubling all the value in the sketch.

put dimension². For models using graph datasets, the comparison between input and output was not trivial, nor is it necessarily the optimal solution. Therefore, alternative loss functions have been tested, including Deep Support Vector Data Description (Deep SVDD), Graph Local Knowledge Distillation (GLocalKD) and One Class Graph Transformation Learning (OCGTL).

Deep SVDD [205] operates directly in the output space of dimension n_{out} and computes the distance between the vector exiting the model and a fixed centre c of the hypersphere in the n_{out} -dimension space, as shown in Fig. 5.20a. Mathematically, it corresponds to:

$$\mathcal{L}_{\text{Deep SVDD}} = \frac{1}{B} \sum_{i=1}^B \|f_{\theta}(x_i) - c\|_{n_{out}}^2 + \frac{\lambda}{2} \sum_l \|W_l\|^2 \quad (5.17)$$

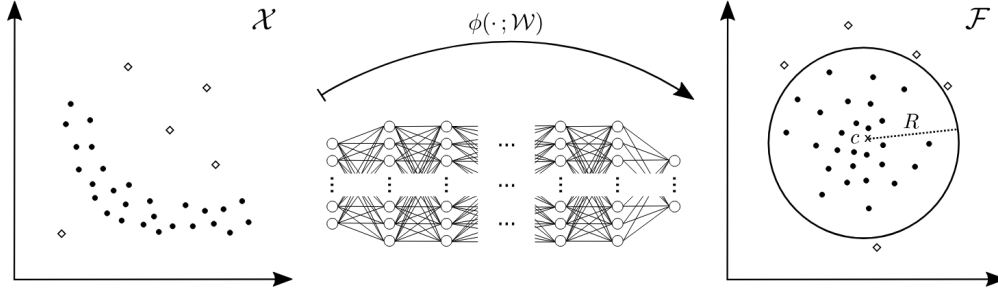
where B is the batch size, $f_{\theta}(x_i)$ is the output of the model for the input vector x_i and $\|\cdot\|_{n_{out}}^2$ is the squared Euclidean norm in the n_{out} -dimensional feature space. The second addendum to the loss is a regularisation term, where λ is a regularisation coefficient and W_l are the network weights. The regularisation term is needed to prevent the model from collapsing to a trivial solution where all the embeddings become zero. Therefore, as suggested in [205], it is important to initialise both the centre c and the second term of Eq. 5.17 before the training. Minimising the Deep SVDD loss has the effect of encouraging all embeddings to be as close as possible to c , reducing the radius of the hypersphere. When used for the AD task, this loss clusters all the standard data around c , while anomalous data are expected to lie outside the hypersphere, as shown in Fig. 5.20a. The AS for each data point passed to the model is then computed as:

$$\text{AS}_{\text{Deep SVDD}} = \|f_{\theta}(x_i) - c\|_{n_{out}}^2 \quad (5.18)$$

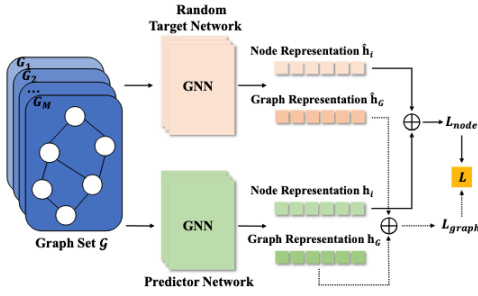
When using the Deep SVDD loss, the regularisation parameter has been set to $\lambda = 1 \cdot 10^{-6}$, while the centre c is obtained by the mean value of the network predic-

²In reality, in TAE1 and TAE2 the loss is calculated between the output and the input exiting the initial MLP block, which share the same vector dimension.

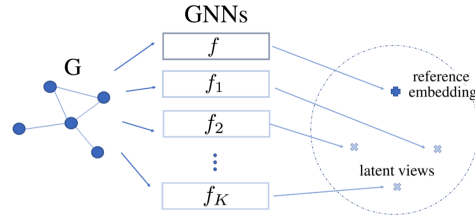
tion over the elements of the training dataset, once the model has been initialised randomly.



(a)



(b)



(c)

Figure 5.20. Visual representations of the Deep SVDD, GLocalKD and OCGTL losses.

In Fig. 5.20a it is shown how starting on the left from not so well separated anomalous (empty dots) and standard samples (filled dots) in the input space $\mathcal{X}$, then the input is passed to the network and the loss tends to cluster in the same hypersphere the standard entities in the feature space $\mathcal{F}$. For an immediate representation the input and feature spaces are drawn two-dimensional, but it can be generalised to every dimension. Figure from Ref. [205]. Fig. 5.20b shows the Knowledge Distillation for the graph (global) and node (local) representations in the Random Target Network, the teacher, and in the Predictor Network, the student. Figure from Ref. [206]. Fig. 5.20c represents the $K + 1$ feature extractors projecting their embedding of the graph in the latent space. Figure from Ref. [207].

GLocalKD is a loss function specifically designed for graph-level AD [206]. It combines a variation of Knowledge Distillation techniques at global and local scales. The loss needs two models to be trained: a teacher T that learns high quality graph representations and a student S that attempts to mimic the behaviour of the teacher. Global (graph-level) and local (node-level) structures are analysed and compared, since the method assumes that discrepancies between the two structures indicate anomalous events. A visual representation of it is depicted in Fig. 5.20b.

Therefore, the loss can be written as:

$$\begin{aligned}
\mathcal{L}_{\text{GLocalKD}} &= \mathcal{L}_{\text{glob}} + \lambda \mathcal{L}_{\text{loc}} \\
\mathcal{L}_{\text{glob}} &= \sum_{i=1}^B \|g_i^T - g_i^S\|^2 \\
\mathcal{L}_{\text{loc}} &= \sum_{i=1}^B \sum_{v \in \mathcal{N}_i} \|h_v^T - h_v^S\|^2
\end{aligned} \tag{5.19}$$

where λ is the parameter that balances the global and local losses, g_i and h_v are the graph and node representations for the teacher and student models and B is the number of graphs in the batch. As reported in [206], the common practice to use this loss involves randomly initialising a network, called the Random Target Network, that will act as the teacher, and train another model with exactly the same architecture, the Predictor Network, playing the role of the student. The effect of this strategy is to force the global and local representations in the Predictor Network to closely resemble the output of a randomly initialised model. In this way, patterns extracted from frequently encountered data, the standard samples, will have a smaller prediction errors than rare anomalous events. The GLocalKD loss was tested but quickly discarded, as it requires not one but two models, which in our case are considerably large, and the results were suboptimal. In the case of jets, local information that represents the jet's substructure, is likely more important than the global overall embedding. Even setting the λ parameter to high values does not help, since the problem lies in the norm between the teacher and student representation of the nodes: the randomly initialised network fails in detecting both anomalous and standard substructures, and so does the student.

The OCGTL loss function [207] is a variation of one-class losses, such as Deep SVDD, intended specifically for graph-level AD. It consists in the sum of two terms, the one-class classification loss $\mathcal{L}_{\text{OCC}}$, which is analogous to Deep SVDD for graphs, and the graph transformation learning loss $\mathcal{L}_{\text{GTL}}$ which promotes the diversity among the embeddings produced by the different feature extractors of the graph (see Fig. 5.20c), that is:

$$\begin{aligned}
\mathcal{L}_{\text{OCGTL}} &= \mathcal{L}_{\text{OCC}} + \mathcal{L}_{\text{GTL}} = \mathcal{L}_{\text{Deep SVDD}} + \mathcal{L}_{\text{GTL}} \\
\mathcal{L}_{\text{GTL}} &= \sum_{k=1}^K \log \frac{c_k}{C_k} \\
c_k &= \exp\left(\frac{1}{\tau} \text{sim}(f_k(g), f(g))\right) \\
C_k &= c_k + \sum_{l \neq k}^K \exp\left(\frac{1}{\tau} \text{sim}(f_k(g), f_l(g))\right)
\end{aligned} \tag{5.20}$$

where τ is a temperature parameter that controls the sharpness of the distribution, $f_1, \dots, f_K, f$ are the $K+1$ graph feature extractors and sim is the similarity measure of two graphs. The similarity measure is often computed as the cosine similarity $\text{sim}(z, z') = z^T z' / \|z\| \|z'\|$. In order to use this loss, the $K+1$ graph feature extractors were 5 networks, with exactly the same architecture, simultaneously trained.

The OCGTL loss has been tested on the LHC Olympics 2020 dataset, resulting in similar performance as the Deep SVDD. Since OCGTL requires a training time five times higher (on the whole dataset OCGTL takes 15 minutes per epoch with respect to 3 minutes per epoch when using Deep SVDD), we decided not to use this loss and preferred the Deep SVDD loss.

5.5.3 Configuration of common hyperparameters

In addition to the specific characteristics of each individual model and training, some hyperparameters, such as dropout and batch size, have been set to default values which proved robust and optimal for most models. The dropout value, in fact, has been set to 0.3, for all the networks that have a dropout layer in their architecture, but the EGAT and GIN architectures for which the dropout has been set to zero. Although different values were tested, the optimal batch size value was found to be 1024 for all models. This represents a trade-off: smaller values slow down the epoch duration and make the loss very unstable, while larger values may induce overfitting. A similar argument can be done for the learning rate: many values have been probed, but very often the best results were obtained with learning rate in the range $lr = [10^{-6}, 10^{-3}]$. Moreover, all trainings employed an adaptive learning rate scheduling, that starts from a predefined value and decreases at specific milestone epochs. The milestone epochs are set to be at the 50%, 75% and 90% of the total epochs, while the learning rate is decreased by a factor $\gamma = 0.2$ at each milestone epoch. In the learning curves, the milestone epochs are often clearly visible with a drop in the loss values. No early stopping was applied during trainings, to fully exploit the adaptive learning rate.

During the training, the weights of each model are saved in the epoch that minimises the loss. This approach may result in overfitting, therefore learning curves for training and validation losses will be always monitored and overtraining will be discussed case-by-case.

5.6 Results

In this Section, I will detail all the tests conducted over the LHC Olympics 2020 R&D dataset, in order to fully understand the behaviour of the models, to assess advantages and disadvantages of each approach, to comprehend the effects of the transformation described in Sec. 5.3.1, to validate how the AD models are able to generalise to unseen datasets and, finally, to select a set of models and an AD strategy for the ATLAS Run 3 data search. The results will be discussed on the basis of the learning curves of the models, to monitor overfitting, as well as in terms of the AUC values and, for supervised trainings, accuracies and Confusion Matrices. Another important plot is the AS distribution for the background and signal datasets. While the AUC summarises the average behaviour of the AS distribution in a single value, the AS can exhibit complex shapes that a simple AUC cannot fully capture. In these situations, the AS distributions will be shown and discussed.

5.6.1 GNN4AD

The GNN4AD framework is intended to be a small platform where designing and evaluating the best graph layer and its hyperparameters. GNNs are complex models to train and optimise because the input graph data lacks of a fixed structure and a single graph layer significantly can slow down the training phase depending on the topology of the specific graph. For these reasons, it is more practical to train small architectures, as described in Sec. 5.4.4, on small datasets to explore the hyperparameter space, optimise the performance, and eventually scale up to larger datasets and/or more complex architectures. The graph layers that have been tested with GNN4AD are: GraphConv, EdgeConv, SAGEConv, GINConv, GINEConv, GATConv and EGATConv. During the first part, the networks are trained with the Deep SVDD loss for 20 epochs with only 10 k jet-events. Once found the best hyperparameters for each graph layer, the same architectures are trained on a bigger dataset consisting of 200 k jet-events, to avoid overtraining and to confirm the obtained results. The best hyperparameter values are obtained by performing a random grid search, using the Deep SVDD loss function and calculating the AUC from reconstruction performance. All the results and the corresponding hyperparameter values are shown in Tab. 5.1; the learning curves and the AS distributions for each graph layer are shown in Fig. 5.21 and 5.22.

Graph layer	lr	Graph Pooling	other hyperparameters	AUC on 10 k events	AUC on 200 k events
GraphConv	10^{-4}	max		57.5%	58.3%
EdgeConv	10^{-5}	max		65.2%	59.6%
SAGEConv	10^{-7}	sum	aggregator: mean	62.9%	60.5%
GINConv	10^{-4}	max		64.0%	62.5%
GINEConv	$5 \cdot 10^{-6}$	max		60.5%	62.1%
GATConv	10^{-7}	max	# heads: 2	66.7%	66.7%
EGATConv	$5 \cdot 10^{-7}$	max	# heads: 8	69.7%	68.2%

Table 5.1. Table reporting the hyperparameters that resulted the best for each graph layer. The last two columns show the AUC obtained in the training with 10 k and 200 k events. In general, almost all the models perform the same or slightly worse on the larger dataset, since it is easier to overfit with 10 k events. The graph layers in bold represent the models that will be used for the other tests.

The general behaviour of most graph layers tested is to perform similarly, or slightly worse, when transitioning from the smaller dataset to the larger, indicating a mild overfitting on the small dataset. The EGATConv layer has obtained the best results, both on the 10 k and 200 k events datasets, with only a small degradation of the AUC value when scaling up to more events. The EGATConv layer is immediately followed by the GATConv layer, with comparable results, showing that the atten-

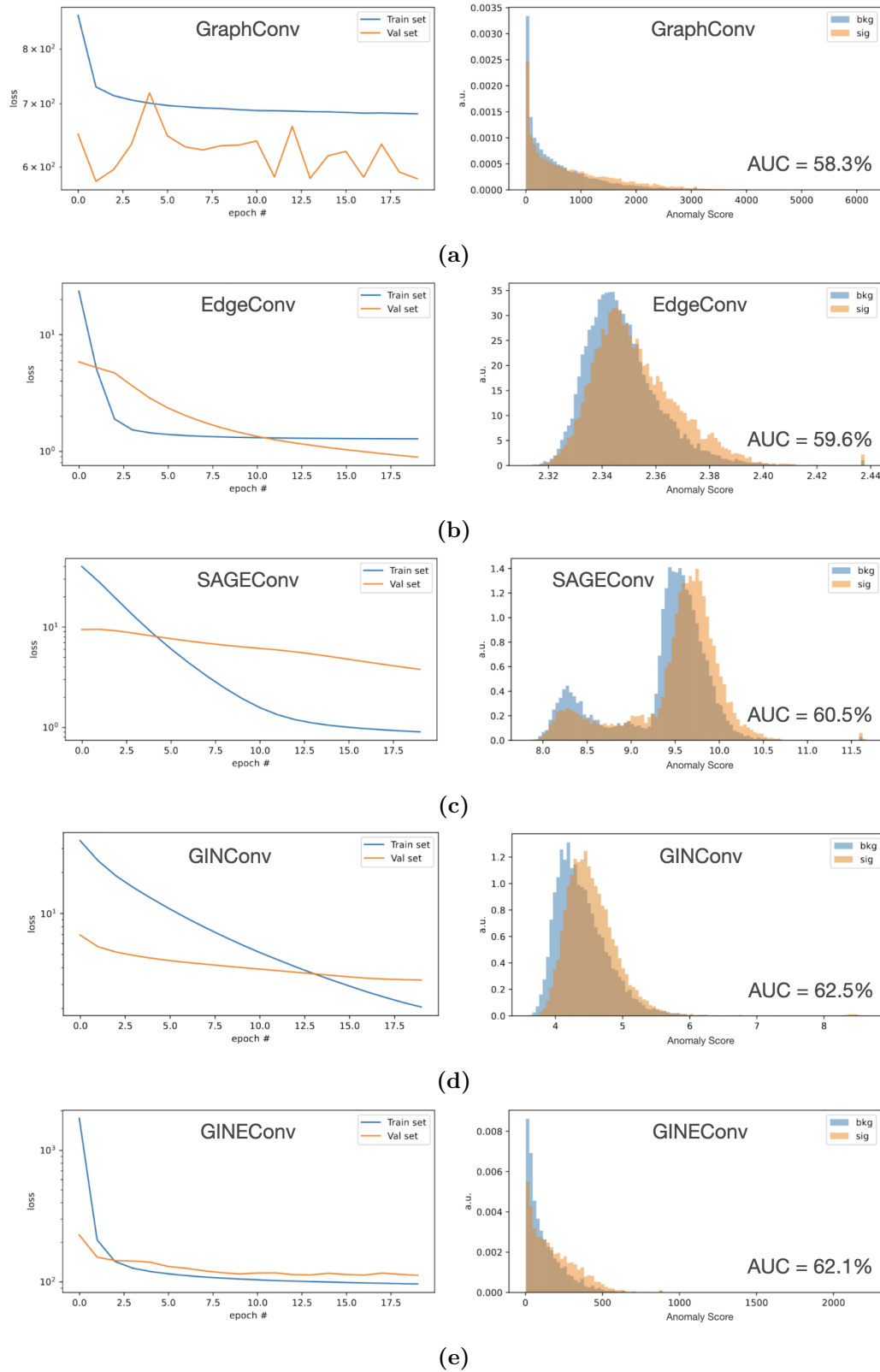


Figure 5.21. The learning curves (on the left) and the AS distributions (on the right) for different graph layers in the GNN4AD framework for the training over 200 k events. Starting from above, the results for GraphConv, EdgeConv, SAGEConv, GINConv and GINEConv. Among these five models, GINConv (Fig. 5.21d) seems to be the most promising, since both the training and validation loss decrease with the epochs, the validation loss plateaus at small values. This results in a small peak value in the AS distribution, that is also discriminant with respect to the signal. The learning curves are in semilog scale.

tion mechanism is highly effective on the jet substructure identification. In fact, by examining the corresponding learning curves and AS distribution in Fig. 5.22a and 5.22b, it is clear that the losses in both cases are learning the representation of the input data and, when tested on signal jet-events, the AS distribution differs from the background distribution, as desired. It is worth noting also that the AS

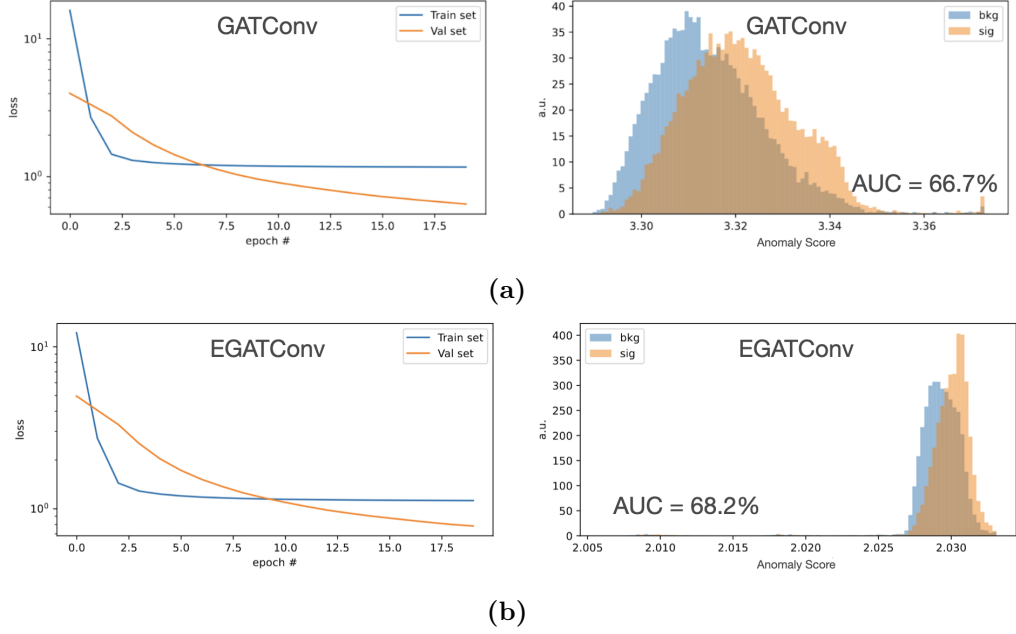


Figure 5.22. The learning curves (on the left) and the AS distributions (on the right) for different graph layers in the GNN4AD framework for the training over 200 k events. Here are shown the results for the GATConv and EGATConv graph layers. The two models show similar results, with EGATConv (Fig. 5.22b) performing slightly better than GATConv. The learning curves are presented on a semilog scale.

values in both cases are small in absolute, indicating a good pairing between the model architecture and the Deep SVDD loss. In the case of the other graph layers, many effects occur simultaneously. The GraphConv layer performs the worst on both datasets, even if slightly better on the bigger dataset. By looking at the corresponding learning curve in Fig. 5.21a, it is immediately apparent that the validation loss is overfluctuating and the loss values in absolute are significantly large, indicating a poor training. Similar AUC results are obtained by the EdgeConv layer, with the difference that in this case the training is performed correctly, the AS values are small (see the learning curve and AS distribution in Fig. 5.21b), but the model assigns similar AS values to both the signal and background events making it difficult to separate them. It is clear, then, that convolutions from GraphConv do not work well on the jet substructure, as well as representing the constituents of the jet as point clouds like in EdgeConv layers. The SAGEConv layer, instead, performs slightly better, the training is consistent with the others, but the AS distribution shape in Fig. 5.21c with its double peak risks mitigating the separation between signal and background. In fact, the signal AS distribution seems to be shifted with respect to the background, suggesting a certain degree of discrimination power, but

the double peak diminishes this effect. It is possible that the node aggregations typical of SAGEConv layers is averaging the peculiarities of jet substructure, and it is in these peculiarities that the discrimination lies. The GINConv and GINEConv layers at first glance seem to have similar AUC score, but even if both the learning curves are consistent they have quite different AS distributions. In fact, the AS values for GINEConv span the range $AS_{\text{GINE}} \in [0, 600]$, that is much larger than the $AS_{\text{GIN}} \in [3, 9]$ range of GINConv layer. This suggests that the model struggles to identify the underlying patterns in the data, which is undesirable. In general, a broad AS range is not problematic, as the key factor is whether the AS value is discriminative. But if under the same conditions, using the same Deep SVDD loss, the AS values differ significantly only by changing the graph layer, it is a sign that the GINEConv layer is missing completely relevant features of the dataset. It is likely that the MLP used as an aggregator for the GINConv layer (Eq. 5.13) outperforms the ReLU employed in the update rule for the GINEConv in Eq. 5.14. To conclude the GNN4AD study, the most promising graph layers are EGATConv, GATConv and GINConv which show good overall performance and a well behaved training. Since the GATConv and EGATConv layers share the same attention mechanism and differ only in the concatenation of the edge information in EGATConv, which slightly outperforms GATConv, we have decided to use the GINConv and EGATConv layers for subsequent studies.

5.6.2 Classification performance

When starting a new difficult task, it is always important to have benchmarks for comparison. In the case of this project, it was the same: the challenging task is to develop an unsupervised AD architecture to search for NP in a model-agnostic approach. The difficulties arise from many different levels, from the determination of the best data representation, to the choice of the ML network, from the strategies to adopt to make the model robust, to best hyperparameter search. Moreover, even after everything has been discussed and decided and the first results have emerged, such as for the GNN4AD framework, we still need a yardstick against which to compare the results and assess their adequacy or need for improvement. In this direction, then, the classification approaches come to the rescue and can provide a range of performance that can be expected from AD models. A sort of lower limit is offered by a Random Forest model with a supervised training, since traditional, model dependent analyses are based on a sequence of cuts applied on the event's high level or low level variables. A Random Forest represents the best selection obtainable with simple cuts; therefore, its performance is comparable to that of a standard analysis in HEP using low-level jet constituents to separate QCD dijet events from specific signal events. Of course, the lower limit set by the Random Forest should not be intended literally, since Random Forests are supervised models, but it can be a useful benchmark for AD. Since supervised approaches can focus on the specific signal and maximise the separation between signal and background, classification models will reflect the upper performance limit of an analogous AD model. For these reasons, different classification algorithms, already described in Sec. 5.5, have been developed: a Random Forest, the GIN model, the Transformer

and the GTransformer architectures. The AUC scores and accuracy values are shown in Tab. 5.2.

Classification model	Accuracy	AUC
Random Forest	70.5%	78.8%
GIN model	74.3%	83.2%
Transformer	84.5%	91.3%
GTransformer	73.8%	82.4%

Table 5.2. Results of the classification models in terms of accuracy and AUC.

The Random Forest model yields the lowest performance among the four models, but still demonstrates effective discrimination between background and signal jet-events. The Transformer model results in the best performance, reaching an AUC value above 90% and exhibiting excellent discrimination, as shown in Fig. 5.23. This

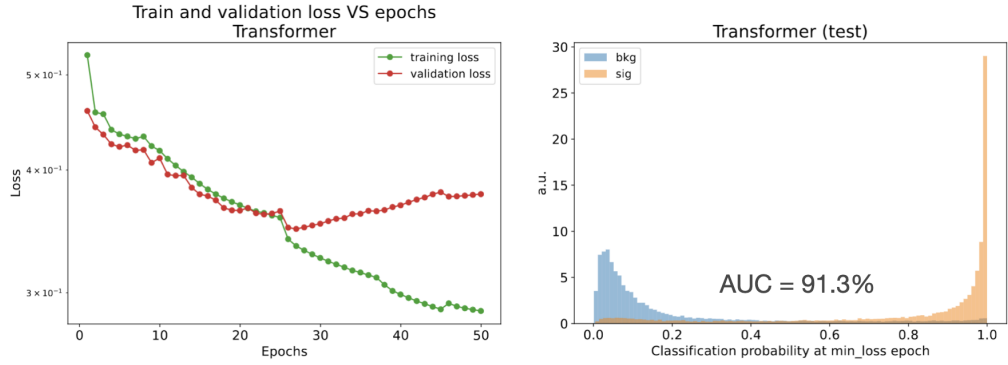


Figure 5.23. Performance of the Transformer model on the classification task. On the left, the learning curve for training (in green) and validation loss (in red) calculated on the balanced dataset. The model starts overfitting at around epoch 25, which corresponds to half the training. The steps in both the training and validation losses at epoch 25, 37 and 45 are due to the learning rate scheduler. On the right, it is shown the AS distribution for signal and background events from the test dataset. The distribution is evaluated at the epoch for which the validation loss reaches its minimum. The AS in a classification task is given by the value of the softmax operating on the output binary vector.

is also the value of the best AUC obtainable by using only low level variables with LHC Olympics 2020 R&D dataset. The GIN model, on the other hand, exhibits lower yet adequate performance. From Fig. 5.24, it is possible to notice that as the training loss is smoothly decreasing, the validation accuracy is increasing. The GTransformer model shows performance comparable to the GIN model; indeed, both utilise the same edge $1/\Delta R$ information. Fig. 5.25 shows a healthy learning curve, with overfitting arising around epoch 25. For the classification task, the inclusion of the graph structure and edge information has the effect of improving the performance, but it seems also that letting the model assign freely the edge value, through the Transformer architecture, is more effective than using fixed edges.

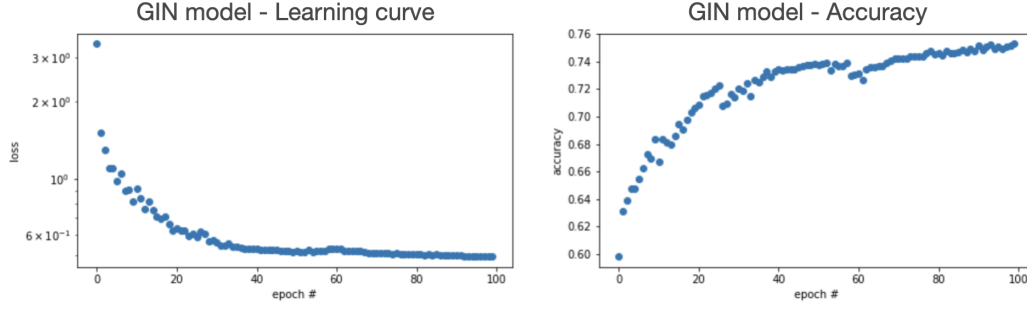


Figure 5.24. Learning curve (on the left) and accuracy evolution (on the right) during the training phase for the GIN model. The loss is calculated on the balanced dataset. The training occurs smoothly, the accuracy on the validation set keep increasing with the epochs plateauing at around 75.0%.

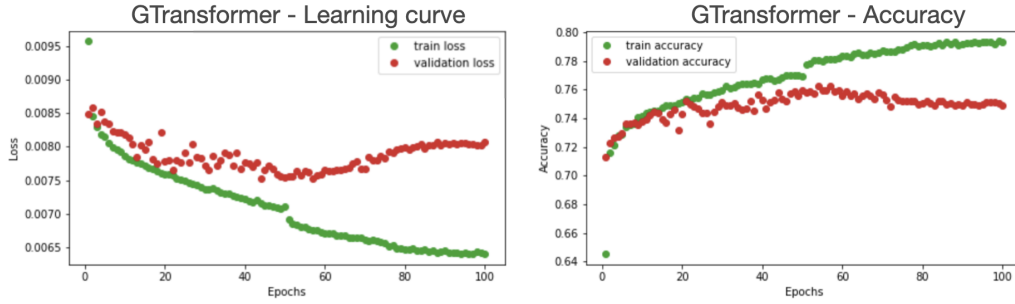


Figure 5.25. On the left, the learning curve for the training (in green) and validation losses (in red) during the GTransformer training. On the left, the accuracy values obtained with the training and validation dataset evolving with the epochs. In both the plots, it is evident when the model starts overfitting at around epoch 25: the validation loss starts increasing, while the accuracy tends to slightly decrease.

5.6.3 Anomaly Detection performance

For the AD task, we have tested the following models: the GIN AD³, the EGAT AD, the Transformer AD, the GTransformer AD, the TAE1 and TAE2 models. The results for all models are shown in Tab. 5.3 with the corresponding learning curves and AS distributions presented in Fig. 5.26 and 5.27. The majority of the models have similar AUC scores, around 73-75%. The graph models, GIN AD and EGAT AD, both show similar learning curves (Fig. 5.26a and 5.26b), with the validation loss that starts below the training loss but at epoch 4 and 3, respectively, it becomes higher than the training analogous. This effect, commonly observed in AD trainings, can be mistaken for overfitting, but overfitting cannot occur with this type of AD trainings. In fact, the validation set does not contain signal samples. What is happening is that in the first part of the training the training loss, which is calculated batch per batch and then averaged per each epoch, is higher

³By adding the term “AD” after a network name, I mean that the model is analogous to the one used for classification, but the last layer is exiting with a suitable shape for calculating the MSE or Deep SVDD losses, as already discussed in Sec. 5.4.2.

than the loss calculated at the same epoch over the validation set, with the weights calculated from the last batch. At a certain epoch, the loss value shows minimal variation across different batches, then the training loss, which is averaged over a larger dataset, will appear lower than the validation loss for the same epoch. A phenomenon similar to overfitting, where the model appears to perform exceptionally well while merely learning trivial patterns such as the identity transformation with the MSE loss, can occur even with AD trainings, but it is much more difficult to spot. The Transformer AD and GTransformer AD models also have similar learn-

AD model	AUC
GIN AD	73.7%
EGAT AD	75.5%
Transformer AD	75.5%
GTransformer AD	75.7%
TAE1	74.0%
TAE2	69.1%

Table 5.3. Results of the AD models in terms of AUC scores.

ing curves, but although the AUC values seem comparable, they differ from the previous models for the AS distribution, as shown in Fig. 5.27a and 5.27b. It is not surprising that the Transformer and GTransformer AD exhibit similar AS distributions, given the analogous model structure. They employ the same MSE loss with the sole difference that one represents the jet as a fully connected graph, while the other does not. In contrast, while the AS distribution for the GIN and EGAT AD models displays distinct peaks for the background and signal events, with the background peaking at a smaller value than the signal’s peak, the AS distribution for the Transformer and GTransformer AD models exhibits a completely different shape. It has two peaks, and both the background and signal populations show this feature. Signal and background, however, have different relative event fractions in the two peaks: signal events tend to populate the peak closer to zero, while background events do the opposite. The different AS shapes are the result of the different losses used: the graph models use the Deep SVDD loss which tends to minimise the radius of an hypersphere in the output space; the (G)Transformer models, instead, use an MSE loss to minimise the reconstruction error between input and output. In this sense, nothing prevents the signals to be reconstructed more easily, therefore with smaller loss values, than the background. This can also occur with the Deep SVDD loss, though it is rarer, since the evaluation does not focus on the reconstruction error of specific inputs. The TAE1 model (Fig. 5.27c) produces results consistent with the other networks, demonstrating that the representation generated by the three AEs to extract information for the query, key and value for the subsequent MHA module is comparable to that of a complete Transformer architecture. In contrast, the TAE2 model (Fig. 5.27d) performs slightly worse, suggesting that the compression achieved by pairing MLPs with varying dimensions and MHAs is less effective than the TAE1 approach.

Note that when the AS is calculated with the MSE loss, the loss scores in the learn-

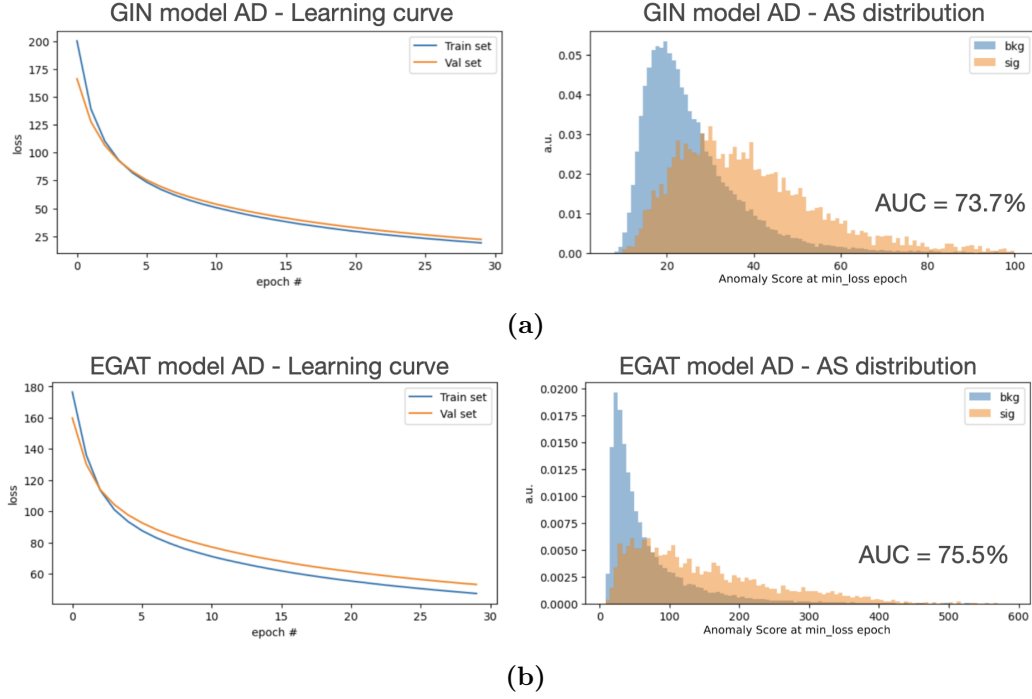


Figure 5.26. Learning curves and AS distributions for the GIN AD (Fig. 5.26a) and EGAT AD (Fig. 5.26b).

ing curves appear significantly different from the AS values. In fact, for the MSE calculation, the Pytorch `nn.MSELoss` function [208] has been used and then averaged over all the events. The AS, on the other hand, is computed event-by-event as the square root of the sum of the squares of the differences between input and output vectors. It has been verified that the square of the AS of all events in a given dataset equals the sum of all losses over the batches of the same dataset. Other apparent differences arise from the loss being computed over training and validation datasets, whereas the AS is evaluated on the test dataset.

In conclusion, with the exception of the TAE2 model, all the other architectures have shown similar results. Nonetheless, the Transformer AD is significantly faster than the others which is why it was often selected for tests that prioritised model performance over architectural structure.

5.6.4 Effects of the transformation

The transformation described in Sec. 5.3.1 and in App. B has the key responsibility to remove any dependence of the dataset from the mass and energy of the original jet and to make the dataset rotation and translation invariant. In order to verify whether the transformed dataset possesses these characteristics, we tested the same architecture, the Transformer AD, with different datasets with the transformation fully or partially removed. The aim of this Section is to validate the rotation and translation invariance of the dataset on the model performance.

Together with the transformed and untransformed, “pure” datasets, with kinematic

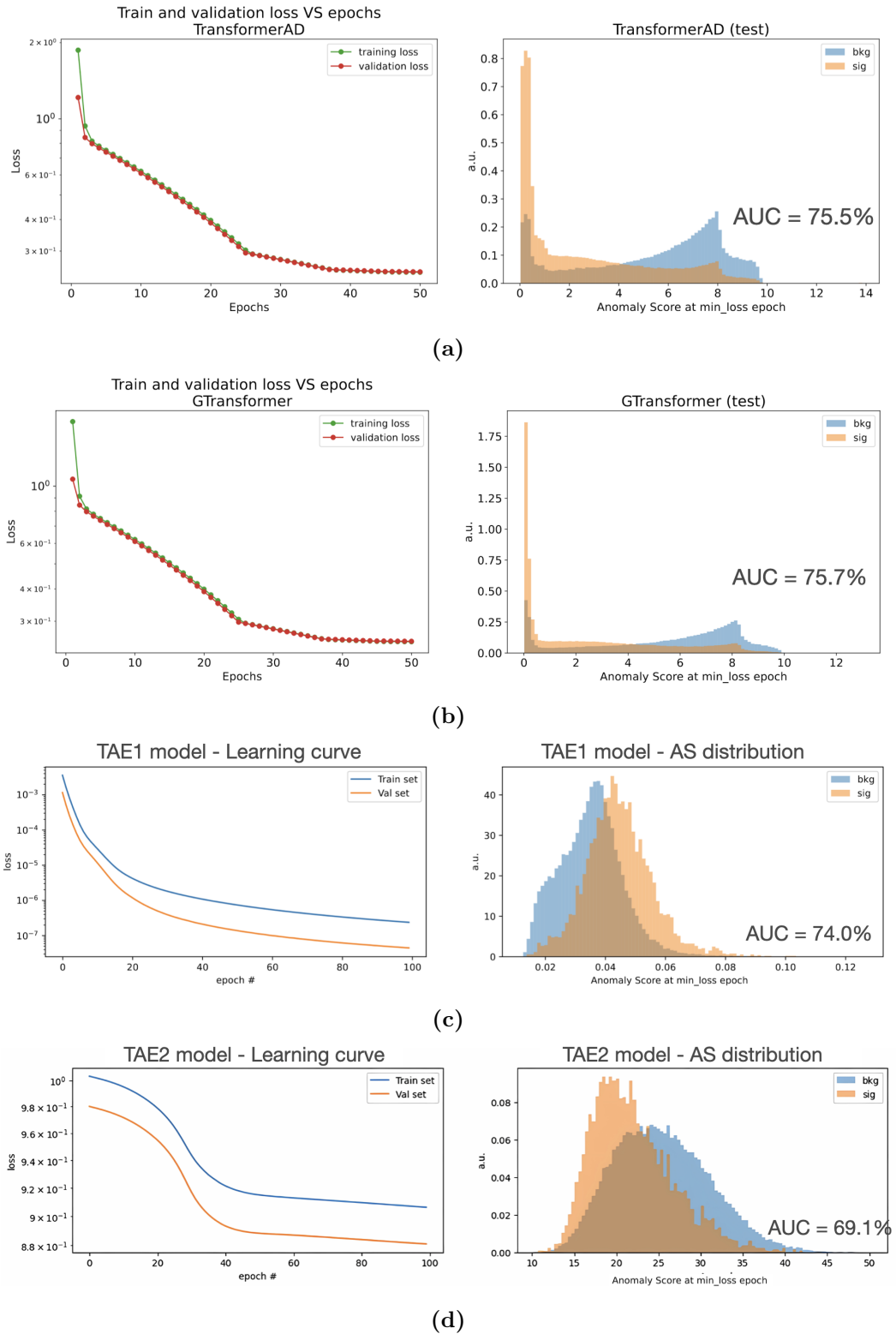


Figure 5.27. Learning curves and AS distributions for the Transformer AD, GTransformer AD, TAE1 and TAE2 models, in Fig. 5.27a, 5.27b, 5.27c and 5.27d respectively. In the Transformer AD and GTransformer AD learning curves, the different trends of the losses are due to the learning rate scheduler.

distributions shown in Fig. 5.7, a new dataset has been created. Only the rotation around the jet axis has been applied to this new dataset, corresponding to a rotation of angle $-\phi$ and a Lorentz boost along the z -direction, positioning the rotated jet axis at the origin of the η - ϕ plane. The rotated jet four momentum become the one reported in Eq. B.7. For now on, I will refer to this dataset as the rotated dataset. The kinematic distributions for the LHC Olympics 2020 R&D rotated dataset are shown in Fig. 5.28. The jet η and ϕ variables are now shrunk to the origin, while

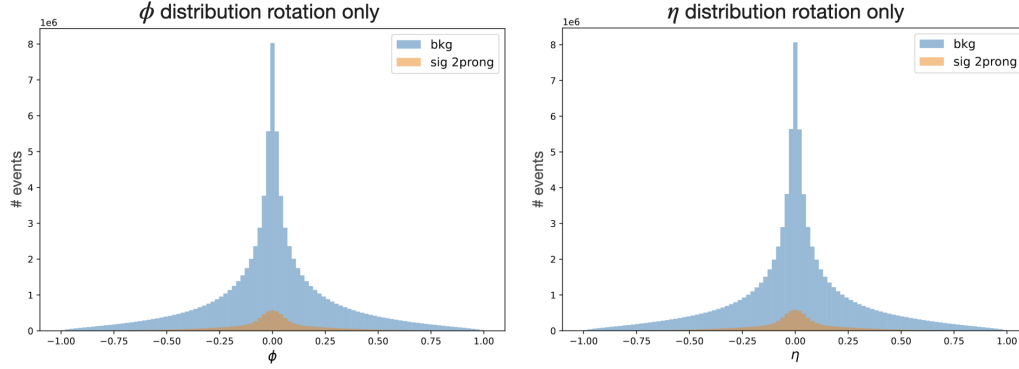


Figure 5.28. η and ϕ distributions for the constituents of the LHC Olympics 2020 R&D dataset while applying only the rotation of the jets. The distributions of the two variables appear almost identical. The distributions have not been normalised, therefore the difference in the signal and background samples are due to the different statistics.

the p_T distribution, which is not shown, has not been changed by the rotation. In practice, the described rotation does not completely remove the dependence on the jet's position in the detector, since the constituents have not been further rotated around the hardest constituent.

A Transformer AD architecture has been retrained and optimised for this test with the three datasets: the pure, the rotated and the transformed datasets. The AS distributions and the corresponding AUC values are shown in Fig. 5.29. When using the pure dataset, the model struggles to extract meaningful information from the constituents. In this case, the network has to learn by itself the symmetries of the dataset, not focusing at all on the jet substructure. As a result, the AS, which reflect the model's ability to reconstruct the input, attains large values and the distributions for the signal and the background events appear nearly identical. The Transformer AD trained with the pure dataset obtains an AUC of 51.6%. The model trained with the rotated dataset, instead, should not learn the symmetry from scratch, as it is partially removed from the dataset itself, and this translates into improved performance. The corresponding AS distribution exhibits lower values and has the two peak shape as previously discussed for the Transformer AD model. This training achieves an AUC of 66.4%. Finally, when using the dataset with the complete transformation applied, the model obtains the same results as before, with an AUC of 75.6%. Note that, as this is a statistically independent training, the performance is slightly different, though fluctuations in the AUC on the order of $o(10^{-3})$ are expected.

In conclusion, we have proven that the application of the transformation removes

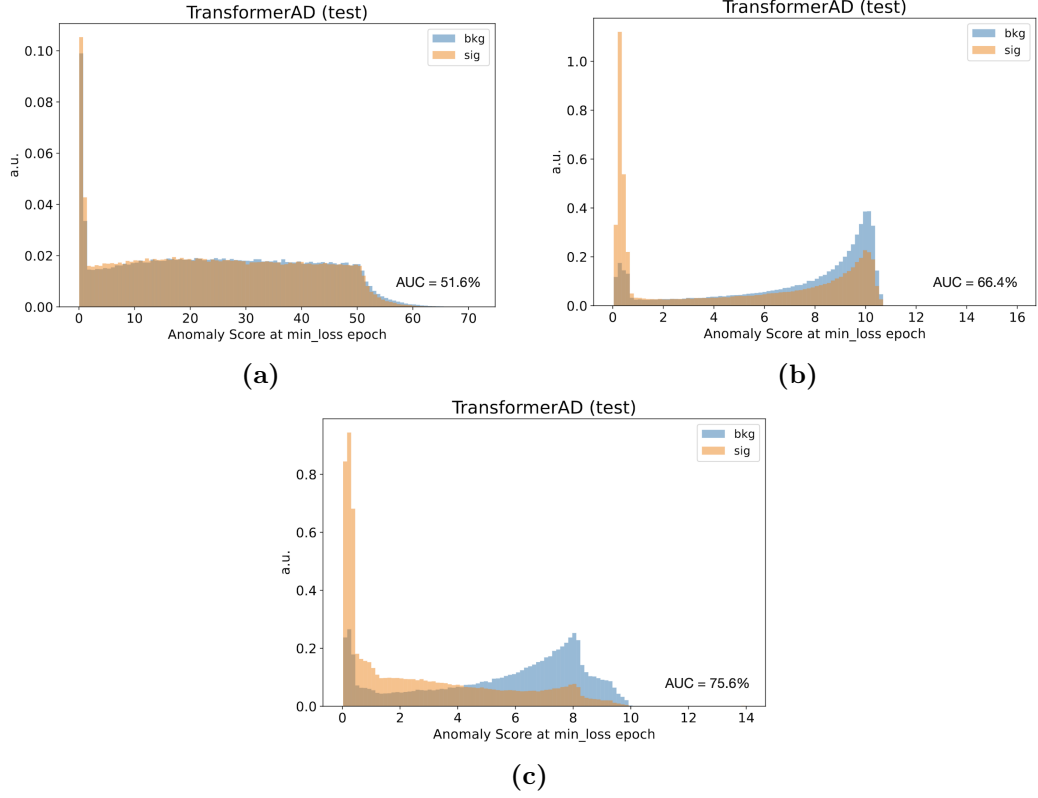


Figure 5.29. The AS distributions for the Transformer AD model with the pure (Fig. 5.29a), rotated (Fig. 5.29b) and transformed datasets (Fig. 5.29c). The AS distribution obtained with the pure dataset is very similar between signal and background, and almost no discrimination can be done. With the rotated dataset, instead, the AS distributions starts to differentiate and an AUC of 66.4% has been obtained. Finally, by using the fully transformed dataset, the AS distribution becomes even more discriminant, reaching a 75.6% AUC.

the spatial symmetries of the jet constituents and helps the model focus on the jet substructure rather than its positioning inside the detector.

5.6.5 Anomaly Detection performance on LHC Olympics 2020 Black Boxes

The LHC Olympics 2020 challenge included three BBs, as discussed in Sec. 5.2. Among the models that joined the contest [160], many were able to detect the 3.8 TeV decaying resonance when acting on BB1, such as Tag N'Train, CWoLa, QUAK and VRNN. In contrast, only the VRNN successfully identify BB2 as containing solely background events, while other models produced false positives. On BB3 no model was able to correctly identify the content of BB3, probably because of the low boost of the resonance R . It is therefore important to assess the performance of the proposed AD approach on the LHC Olympics 2020 BBs. For these studies, we have decided to use the same Transformer AD model, which was already trained and whose performance is reported in Tab. 5.3 and in Fig. 5.27a. Since we

had access to the complete event information, including whether a specific jet belonged to the background or signal, it is interesting to evaluate the performance of the model on the BBs alone or using a statistically independent R&D QCD dijet dataset as background. The results are shown in Tab. 5.4 and in Fig. 5.30.

Signal		Background		AUC
R&D	$Z' \rightarrow XY$	R&D QCD dijet	PYTHIA+DELPHES default settings	75.5%
BB1	$Z' \rightarrow XY$	R&D QCD dijet	PYTHIA+DELPHES default settings	75.6%
BB1	$Z' \rightarrow XY$	BB1 QCD dijet	PYTHIA+DELPHES realistic resolutions	73.2%
BB2	QCD dijets with HERWIG++	R&D QCD dijet	PYTHIA+DELPHES default settings	53.4%
BB3	$G \rightarrow gg$ (2prong) $G \rightarrow gR$ (3prong)	R&D QCD dijet	PYTHIA+DELPHES default settings	50.7%
BB3	$G \rightarrow gg$ (2prong) $G \rightarrow gR$ (3prong)	BB3 QCD dijet	PYTHIA+DELPHES realistic resolution	50.6%

Table 5.4. Performance of the Transformer AD model already trained only on the background events of the LHC Olympics 2020 R&D dataset when tested on different samples. The AUC scored by the model on the R&D testing signals is 75.5%. The BB1 signal dataset consists of 834 $Z' \rightarrow XY$ events with masses $m_{Z'} = 3.823$ TeV, $m_X = 732$ GeV and $m_Y = 378$ GeV. The BB3 consists of 1200 $G \rightarrow gg$ 2prong events and 2000 $G \rightarrow gR$ 3prong events. The masses of the KK graviton and of the neutral R resonance are respectively $m_G = 4.2$ TeV and $m_R = 2.217$ TeV.

The Transformer AD has an AUC score on the LHC Olympics 2020 R&D dataset is 75.5%. When tested on the BB1, the AUC becomes 73.2%; it gets back to 75.6% when tested on the BB1 signal and the usual R&D QCD background (Fig. 5.30a). This is encouraging, as the background modelling has minimal influence on performance and the results are independent of signal statistics, as desired. In contrast, BB2 contains only QCD dijet background events, generated with a different tool. When compared to the usual R&D dataset, the AS distributions are expected to exhibit similar shapes, with the AUC approaching 50%. In fact, this is what happens, as the AUC score with the BB2 is 53.4% and the AS distributions in Fig. 5.30b are very similar. The BB3 consists of a 2prong and a 3prong signals, both coming from the decay of a KK graviton. In the 2prong final state, the graviton decays into two massless gluons; in the 3prong final state, the graviton decays into a gluon and a new massive resonance R . The Transformer AD model tested on the BB3 fails to detect any excess, as indicated by the 50.7% and 50.6% AUC scores with the R&D and BB3 background events, respectively. Again, the small AUC differences when tested on the two background models demonstrate flexibility in handling unexpected detector resolutions. The overall performance on the BB3 dataset is consistent with what the VRNN model has found. In fact, the transformation applied to the datasets

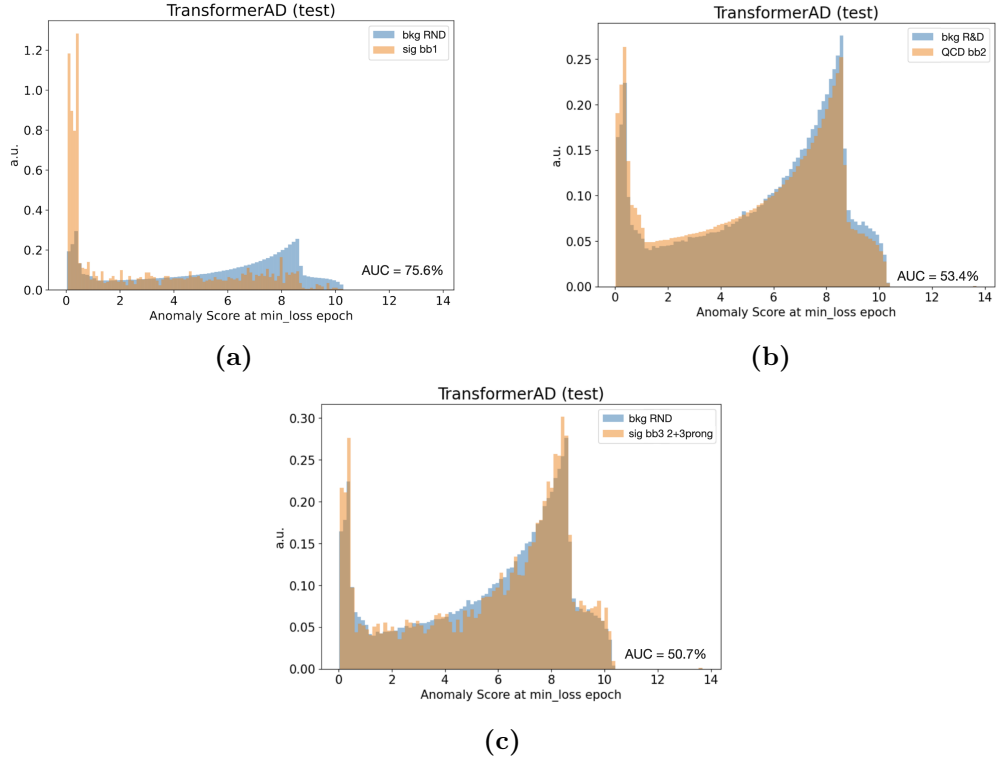


Figure 5.30. The AS distributions when the signal (in orange) corresponds to the different BB content, while the background (in blue) is the LHC Olympics 2020 R&D dataset consisting only of QCD dijet events. All the plots share the same background distribution, but the range on the y -axis is different.

removes dependence on the mass and energy of the initial jet, but retains dependence on the initial boost of the decaying resonances. It is likely, as also suggested by the VRNN developers in [160] who used the same dataset transformation, that models trained on transformed data are specifically tuned for boosted final states, thus struggling to detect resonances in the BB3.

Lastly, the Transformer AD model has been also tested on the 3prong dataset provided in the LHC Olympics 2020 repository. The AUC reaches the 69.0% value, slightly worse than the result on the 2prong data. Fig. 5.31 clearly shows that fewer signal events occupy the initial bins of the distribution, leading to a reduced peak in the AS distribution and a lower AUC. This result was expected, as the kinematic distributions of the constituents after the transformation in the 3prong case closely resemble the QCD dijet distributions compared to the 2prong case, as shown in Fig. 5.7.

5.6.6 Jet- and event-level Anomaly Score

The AD approach discussed so far provides a jet-level AS, meaning that each jet within an event is assigned a distinct AS value. The AS distribution enables the definition of one or more thresholds to identify jets as anomalous if their scores

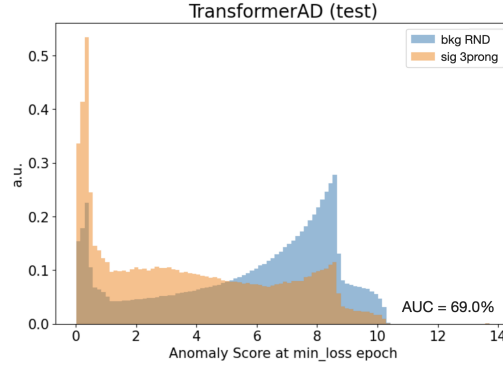


Figure 5.31. The AS distribution of the Transformer AD model tested on the LHC Olympics 2020 3prong dataset (in orange). The background distribution (in blue) is given by the usual LHC Olympics 2020 R&D dataset.

fall above (or below, depending on the distribution) a specific threshold. In order to progress towards the real ATLAS analysis, it is crucial to establish a strategy to aggregate jet-level scores into a single event-level AS. For this study, we have taken into consideration the EGAT AD model, whose jet-level AUC is 75.5% (see Tab. 5.3 and Fig. 5.26b). The jet-level AS from the first two leading jets of the event, ordered by descending p_T , can be combined in various ways. Here we have adopted the simplest approach: the event-level AS is defined as the maximum or the mean of the jet-level scores. Once the anomaly cut in the jet-level AS distribution is defined, there scenarios arise:

- the two jets are not anomalous, hence the event is not anomalous;
- the two jets are both anomalous, hence the event is anomalous;
- one of the jets is anomalous, but the other is not.

In the latter case, the event may be classified as anomalous by considering the maximum AS of the two jets, or as non-anomalous by using the mean of the AS. Results on the EGAT AD architecture indicate that the mean jet-level AS ($AUC_{ev} = 84.3\%$) offers greater discriminative power than the maximum ($AUC_{ev} = 82.0\%$). The distribution of the event-level AS resulting by the mean of the single jet-level scores is shown in Fig. 5.32.

It is now essential to verify whether the AS depends on the kinematical features of the dataset, as it reflects the generalisation ability claimed by the AD approach. In Fig. 5.33, the plots show the correlation between the event-level AS and the mass of the jets for the background and the signal samples and no correlation is found. This validation allows to define a cut on the event-level AS: events with $AS_{ev} \geq 55$ are classified as anomalous; otherwise, they are not anomalous. Note that the threshold has been defined arbitrarily for the specific 2prong event, which is neither model-agnostic nor signal independent. This section is intended to be qualitative rather than quantitative, serving as a starting point for what to expect in the ATLAS analysis. Regardless of how the threshold value has been determined, the AS_{ev} defines a Working Point (WP) characterised by specific background rejection

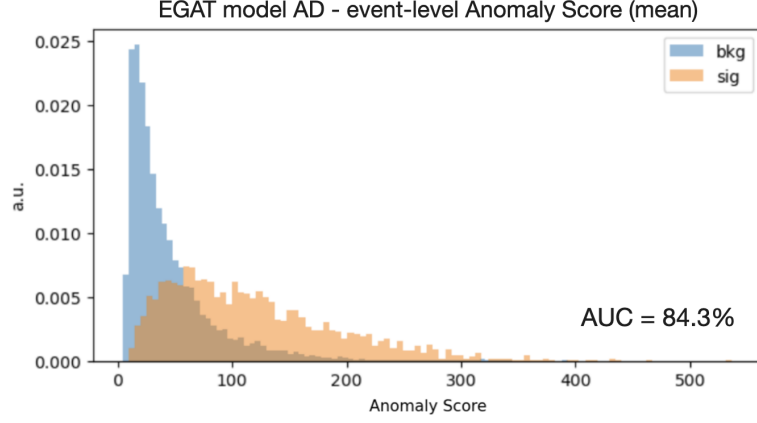


Figure 5.32. The event-level AS distribution for the LHC Olympics 2020 R&D 2prong dataset obtained with the EGAT AD model.

and a signal efficiency, both dependent on the signal under consideration. For the chosen $AS_{ev} \geq 55$ threshold, the background rejection factor, defined as the inverse of the background efficiency, is 3.67, with a corresponding 80.0% efficiency for the 2prong signal. Fig. 5.34 illustrates the effect of applying the cut on the single jet mass and dijet invariant mass distributions. No mass sculpting is observed; the peaks corresponding to the resonance masses appear “cleaned” of background and are clearly identifiable. This represents a preliminary yet promising result in preparation for the ATLAS Run 3 analysis, which will be discussed in Chap. 6.

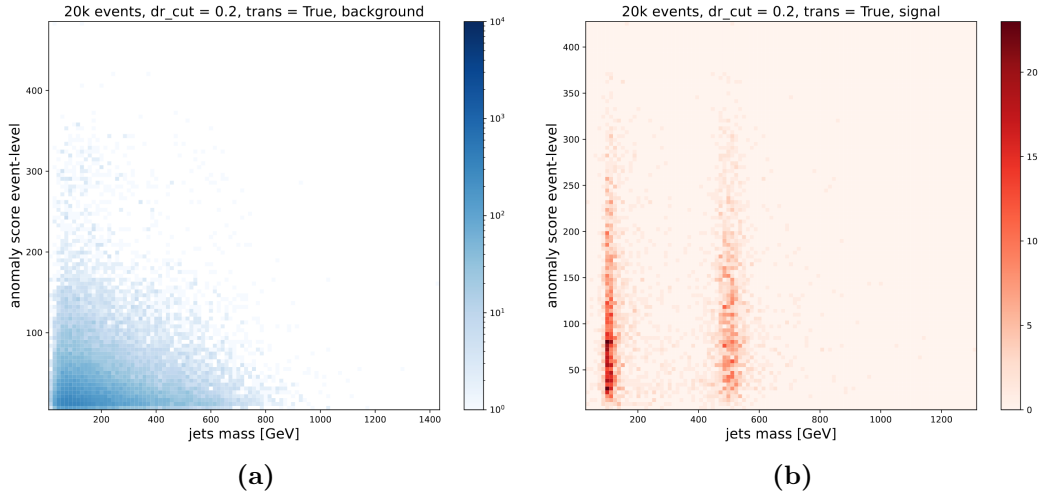


Figure 5.33. Correlation plots of the event-level AS and the single jet mass for the background events on the left, and for the signal on the right. No striking correspondence is shown between the two variables.

5.6.7 Summary of the R&D results

The R&D studies have shown promising results for the application of the AD tech-

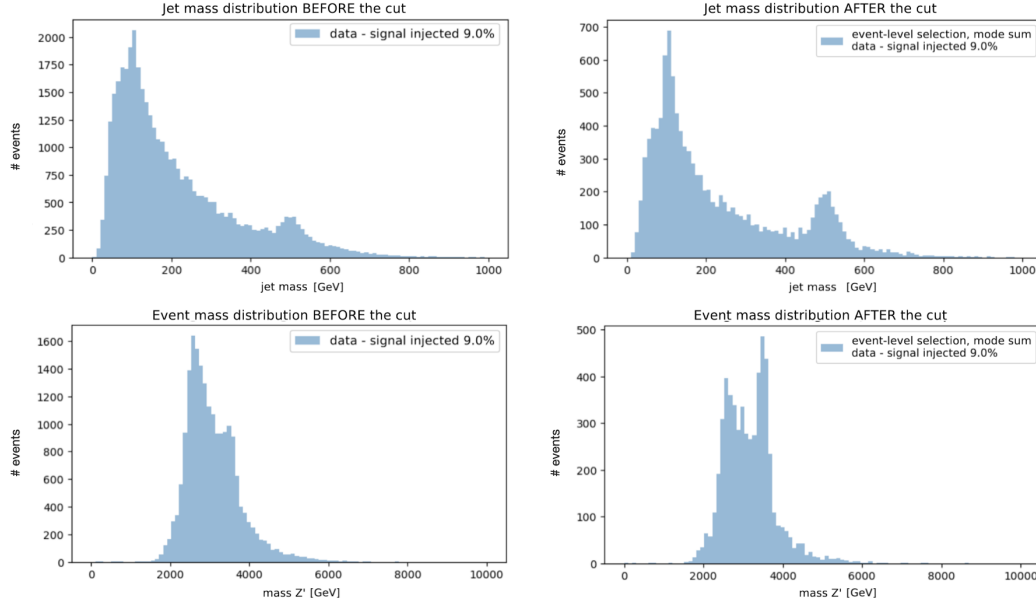


Figure 5.34. Mass distributions before (on the left) and after (on the right) having applied the $AS_{ev} \geq 55$ cut. On top, there are the single jet mass distributions; on the bottom, there are the invariant dijet mass distributions. The effect of applying the cut improves significantly the signal discrimination.

niques to the ATLAS analysis. The training were performed on QCD LHC Olympics 2020 R&D jets, corresponding to the background, providing the independence from the specific theoretical model. For the GNNs, Deep SVDD loss was used as AS, representing the distance of the jets from the centre of a hypersphere in the representation map of the network. On the contrary, for the vectorial models the MSE loss was chosen and therefore the AS coincides with the reconstruction performance. Thanks to the GNN4AD framework, it was possible to test small GNN architectures on a subset of the dataset and for a limited number of epochs. The graph layers that performed best were the GINConv and the EGATConv (Tab. 5.1). When trained on the full LHC Olympics 2020 R&D dataset, both layers confirmed the optimal performance, achieving approximately 75% AUC, with EGATConv slightly outperforming GINConv, comparable to the results obtained with the Transformer AD architecture (Tab. 5.3).

To better understand the model’s ability to detect anomalies, a study was conducted on the dataset transformation. The applied jet transformations, rotation, boost and rescale, proved effective during training, as they remove symmetries and enhance the statistical representativeness of the events. Moreover, these transformations helped decorrelate the AS from other kinematic variables, as shown by tests on the LHC Olympics 2020 BBs (Tab. 5.4). In fact, the Transformer AD model yielded similar results on BB samples containing the same process as the training events but with different masses, and correctly identified no anomalies in BB2, which consisted solely of background events. On BB3, however, the model failed to detect the anomalies. This demonstrates that the network is effectively learning the jet substructure, but is not sensitive to all types of signal, an important consideration

when transitioning to the ATLAS analysis. Finally, it was shown that combining single jet-level AS into an event-level AS leads to a more discriminative variable, thereby improving the model's final performance (Fig. 5.32).

Chapter 6

Anomaly Detection for new resonances search within the ATLAS detector

AD has emerged as a powerful tool for NP searches, as evidenced by the increasing number of publications from the ATLAS and CMS collaborations in the recent years. Heavy resonances decaying into two jets represent an interesting final states, as they can be described by a wide range of theoretical models, including HVT and DJ. However, these final states are heavily dominated by QCD multi-jet production, making it challenging to isolate potential signals. To enhance sensitivity, ML algorithms are exploited for background suppression and signal extraction. Previous Run 2 ATLAS [60, 167, 209] and CMS studies [210] have employed weakly supervised or fully unsupervised methods, using low-level variables to set upper limits on the production cross section of these processes.

In this work, we propose an unsupervised AD search for a BSM resonance decaying into a fully hadronic final state, using partial Run 3 ATLAS data and graph-structured low-level variables. A minimal event selection is applied, requiring only events with two jets and broad kinematic cuts. The trigger selection is optimised to lower the p_T threshold and maximise event recovery, particularly at low transverse momenta. Datasets with background only MC samples are created to validate the approach developed with the LHC Olympics 2020 R&D dataset on more realistic data. The performance of a transformer AD architecture is evaluated on preliminary MC dataset and upper limits on the signal production cross section are derived. Additionally, an EGAT AD model is trained and optimised using a more realistic training strategy. The ultimate goal is to train the models directly on a relatively small amount of data, ensuring that any potential signal contamination remains negligible. This is possible because ML algorithms extract information only from statistically significant samples. Moreover, previous analogous searches, which have already set limits on the NP production in diboson signatures, indicate that the signal contamination, if present, is expected to be minimal.

6.1 Data and MC samples

BSM resonances decaying in the ATLAS hadronic calorimeter leave a signature that differs from the QCD dijet topology. These differences arise from the unique kinematics of the decay, which influence the properties of the decay products, particularly the jet constituents. To exploit these differences, the AD approach we developed relies on ML models described in Chap. 5 that require low-level constituent information. However, the default derivation format for Run 3 ATLAS analyses, DAOD_PHYS, does not include any constituent-level variables. To address this limitation, we requested a new derivation format, the DAOD_LLJ1, which incorporates the low-level information required for our study. The DAOD_LLJ1 format retains the baseline variables present in DAOD_PHYS, with the addition of the constituent variables for large R jets ($R=1$). However, this addition comes with a significant drawback: the event size increases substantially, as each event contains hundreds of jet constituents. In order to keep the size comparable with the default format, an event preselection was applied to the data while preserving physics sensitivity.

6.1.1 Event preselection

The final stage of derivation production consists of the following operations: skimming (thinning) that is the removal of events (objects) whose features do not satisfy specific selection criteria; slimming which removes unnecessary variables associated with a given object; augmentation where new variables, referred to as *decorations*, or new objects are included. For DAOD_LLJ1, the format is augmented with jet constituents information and skimmed to reduce overall data size. Each jet constituent is characterised by its four momentum and an integer *taste* value, which indicates whether the constituent is charged (0), neutral (1) or combined (2). The taste of UFO constituents is determined based on TCC definitions for neutral, charged and combined object definitions, as shown in Fig. 3.6. When a UFO jet has low p_T , it is reconstructed using the PFlow algorithm and its constituents can be either charged or neutral. To further optimise the data size, skimming is performed on the trigger selections and on broad kinematic cuts applied to the jets. To be included in the dataset, an event must satisfy the following conditions:

- the event must be included in the good run list (GRL), meaning that it has been recorded during a set of runs where the ATLAS detector was operating under nominal conditions; this condition does not apply to MC simulations;
- the event must fire at least one of the baseline Run 3 single-jet triggers listed in Tab. 6.1;
- there must be at least one UFO jet reconstructed with anti- k_t algorithm and $R = 1$, applying the SD grooming algorithm and the Constituent Subtraction and SoftKiller (CSSK) PU mitigation algorithm;
- the jet must have pseudorapidity $|\eta| < 2.8$, transverse momentum $p_T > 150$ GeV and mass $m > 30$ GeV.

This event selection procedure reduces the dataset to approximately 15 events per 1000 in Run 3 data, achieving a size reduction factor of approximately 10 compared to the DAOD_PHYS format.

E_T at HLT	Additional clustering info	Other algorithms applied
360 GeV	/	PU subtraction
420 GeV	/	/
420 GeV	/	SD + CSSK + JES factors mass $m > 35$ GeV
420 GeV	trimmed with $R = 0.2$ subjets	JES factors mass $m > 35$ GeV
460 GeV	trimmed with $R = 0.2$ subjets	JES factors
460 GeV	/	SD + CSSK + JES factors
460 GeV	/	PU subtraction + JES factors
460 GeV	seeded with $R = 0.4$ jets	/

Table 6.1. Features of the single jet trigger items used during the event skimming. All items use jets triggered at L1 with $E_T > 100$ GeV and clustering algorithm anti- k_t with $R = 1$.

The DAOD_LLJ1 samples used in this search include Run 3 data collected in 2022 and in 2023 (*data22* and *data23* samples) as well as MC samples simulating detector conditions for the 2023 data taking year (*MC23d* samples). The background for this search consists primarily of SM multi-jet processes, predominantly QCD dijets, with additional contributions from Z/W +jets and $t\bar{t}$ events. QCD dijet samples are generated at leading order in perturbation theory using PYTHIA8 [25]. Given that this background is dominant in several HEP analyses and spans a broad p_T spectrum, events are divided into slices labeled JZX , where X is an integer ranging from 0 to 12. These JZ samples are selected based on cuts applied to the hard scattering transverse momentum $\hat{p}_T$, followed by filtering with anti- k_t jet algorithm with $R = 0.6$. It is important to note that the $\hat{p}_T$ cut does not correspond directly to a cut on the leading jet p_T , as low $\hat{p}_T$ events can still exhibit significant underlying activity. The upper $\hat{p}_T$ thresholds for the QCD slices are set at 20, 60, 160, 400, 800, 1300, 1800, 2500, 3200, 3900, 4600, 5300 and 7000 GeV. In this analysis, only QCD slices JZ0-9 are used, corresponding to $\hat{p}_T \in [0, 3900]$ GeV. Z/W +jets events are generated using the SHERPA2.2.14 generator [111] at next-to-leading order. A selection cut is applied, requiring the Z/W bosons to have $p_T > 200$ GeV. Separate samples are produced for W +jets, $Z + b\bar{b}$ jets and Z +jets excluded $b\bar{b}$. $t\bar{t}$ samples, instead, are generated with POWHEG+PYTHIA8 and first high- p_T emission parameter h_{damp} set to approximately three times the top mass (517.5 GeV).

To benchmark the AD approach, multiple HVT and DJ signals have been generated. Four HVT samples simulate a vector boson V with masses $m_V = 2, 3, 4, 5$ TeV respectively decaying into two SM W bosons, which subsequently decay into 2prong jets. Another four HVT samples simulate a vector boson V with masses $m_V = 2.3, 3.4, 5, 6$ TeV decaying into a SM Higgs bosons and a BSM vector boson X with masses $m_X = 100, 200, 2500, 400$ GeV, respectively. Subsequently, both

the X and H bosons decay into 2prong jets. These eight samples are generated at leading order using MADGRAPH+PYTHIA8. An additional HVT sample has been generated using PYHTIA8 at leading order, simulating a W' boson with mass $m_{W'} = 3$ TeV decaying into BSM W and Z bosons with masses $m_W = 200$ GeV and $m_Z = 400$ GeV. The W and Z boson then decay into 3prong signatures. Lastly, we also prepared a DJ Model A sample with mass $m_{Z_D} = 3.5$ TeV and decaying into two dark quarks of mass $m_{q_D} = 20$ GeV using PYTHIA8. A summary of all generated signals is provided in Tab. 6.2.

Name	Process	Masses [GeV]			Theoretical cross-section [pb]
Heavy Vector Triplets into SM W bosons					
VzWW	$V \rightarrow WW$	$m_V = 2000$	$m_W = 80$	$m_W = 80$	$3.8476 \cdot 10^{-3}$
VzWW	$V \rightarrow WW$	$m_V = 3000$	$m_W = 80$	$m_W = 80$	$3.5969 \cdot 10^{-4}$
VzWW	$V \rightarrow WW$	$m_V = 4000$	$m_W = 80$	$m_W = 80$	$4.5256 \cdot 10^{-5}$
VzWW	$V \rightarrow WW$	$m_V = 5000$	$m_W = 80$	$m_W = 80$	$6.2648 \cdot 10^{-6}$
Heavy Vector Triplets into H boson + X boson					
VcXH	$V \rightarrow XH$	$m_V = 2300$	$m_X = 100$	$m_H = 125$	$4.7435 \cdot 10^{-1}$
VcXH	$V \rightarrow XH$	$m_V = 3400$	$m_X = 200$	$m_H = 125$	$4.1550 \cdot 10^{-2}$
VcXH	$V \rightarrow XH$	$m_V = 5000$	$m_X = 2500$	$m_H = 125$	$1.6359 \cdot 10^{-3}$
VcXH	$V \rightarrow XH$	$m_V = 6000$	$m_X = 400$	$m_H = 125$	$2.1811 \cdot 10^{-4}$
Heavy Vector Triplets into BSM W boson + Z boson					
WprimeWZ	$W' \rightarrow WZ$	$m_{W'} = 3000$	$m_W = 200$	$m_Z = 400$	$6.9682 \cdot 10^{-16}$
Dark Jets Model A					
DarkJet	$Z' \rightarrow q_D \bar{q}_D$	$m_{Z'} = 3500$	$m_{q_D} = 20$	$m_{q_D} = 20$	$1.1142 \cdot 10^{-9}$

Table 6.2. Summary table of the generated signals.

The total luminosity delivered in the ATLAS detector during 2022 and 2023, and collected in the GRL, is 27.6 and 25.8 fb^{-1} , respectively. The MC samples were generated to match the luminosity conditions of 2023 data-taking period.

6.1.2 Trigger strategy and event selection

The trigger efficiency study for all preselected trigger items is crucial to ensure full trigger efficiency for the analysis. Existing trigger recommendations suggest selecting events that fired the trigger item, with p_T -leading jet satisfying $p_T > 500$ GeV and the dijet invariant mass satisfying $m_{jj} > 1.3$ TeV. However, this selection is relatively conservative and there is room for improvements, particularly at lower p_T values. Events stored in the LLJ1 format are already required to satisfy a trigger selection, therefore the trigger efficiency study must be conducted with care, not to introduce biases. In particular, we employ a *tag and probe* method which is widely used for trigger efficiency measurements, particularly when a well-known process decays into two objects. One object, the “tag”, is selected using

a statistically independent method; the other, the “probe”, is used to check the trigger response. In this case, the tag element is the p_T -leading jet if it matches the trigger item and while the probe is the p_T -subleading that has also fired the trigger. Although tagged and probed jets are not entirely statistically independent, they still provide a valid proxy for the trigger efficiency. The efficiency of a given trigger item is, then, evaluated as:

$$ef f_{\text{trig it}} = \frac{(\text{jet}_1 \text{ matched}) \&\& (\text{jet}_2 \text{ matched})}{(\text{jet}_1 \text{ matched})} \quad (6.1)$$

where $\text{jet}_{1,2}$ refer to the p_T -leading and subleading jets of the event, respectively. The resulting efficiency curves are fitted using a sigmoid function $\sigma(x)$:

$$\sigma(x) = \frac{L}{1 + e^{-\frac{x-a}{b}}} \quad (6.2)$$

Here, L represents the plateau efficiency, a is the centre of the sigmoid and b defines the rise interval. From the fit parameters, we determined the p_T threshold at which the trigger efficiency reaches the 97-98% of its plateau value. Applying a cut at this threshold ensures that the selection remains fully efficient. We performed this study across all the trigger items listed in Tab. 6.1. Unfortunately, several trigger with lower threshold, corresponding to jets selected at HLT, were prescaled during 2023 data-taking period and therefore unsuitable for this study. The efficiency curves obtained for the lowest unprescaled trigger item, which corresponds to a selection requiring $E_T > 460$ GeV with SD, CSSK and JES factor algorithms applied, are shown in Fig. 6.1. The results of the fit are summarised in Tab. 6.3.

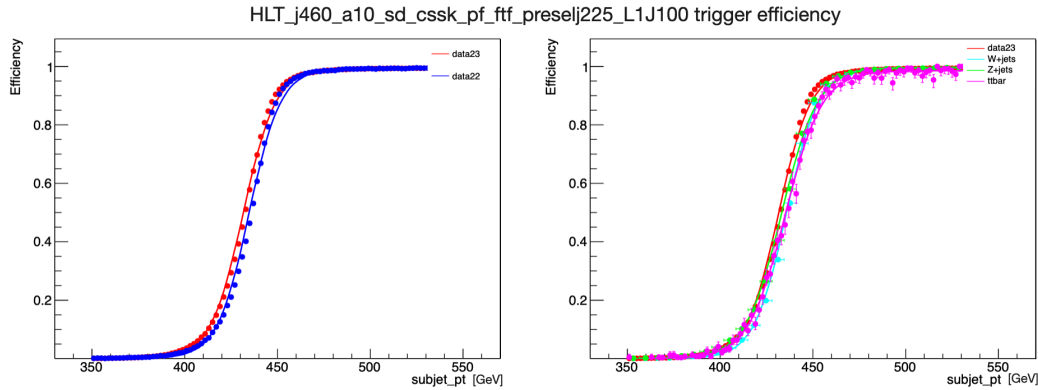


Figure 6.1. Efficiency curves for the lowest unprescaled HLT_j460_a10_sd_cssk_pf_ftf_preselj225_L1J100 single jet trigger item, which corresponds to selecting events with $E_T > 100$ GeV at L1, $E_T > 460$ GeV at HLT. The clustering algorithm used is anti- k_t with $R = 1$ and SD, CSSK and JES factor algorithms are applied. The efficiency curves are shown for the 2022 and 2023 data-taking periods on the left and for the background MC samples on the right. The fit is performed using the sigmoid template (Eq. 6.2) and the parameters of the fit are shown in Tab. 6.3.

Given the p_T values at the selected cut-off, it is safe to lower the selection of the

	L	a [GeV]	b [GeV]	cut-off	p_T at cut-off [GeV]
data22	0.9946 ± 0.0001	434.67 ± 0.03	85.5 ± 0.2	98.5%	474.0
data23	0.9941 ± 0.0001	431.49 ± 0.02	87.9 ± 0.1	98.4%	471.9
W+jets	0.9959 ± 0.0001	435.70 ± 0.03	86.4 ± 0.2	98.6%	475.4
Z+jets	0.9963 ± 0.0001	433.15 ± 0.03	97.1 ± 0.2	98.6%	477.7
ttbar	0.9853 ± 0.0017	435.09 ± 0.09	98.7 ± 1.8	97.5%	480.4

Table 6.3. Parameters of the sigmoid (Eq. 6.2) obtained from the fit of the efficiency curves in Fig. 6.1.

leading jet to $p_T > 470$ GeV. Additionally, we required not only that the event passes the trigger item, but also that the p_T -leading jet is the one that fired it, among all jets in the event. In contrast, no constraints are applied to the dijet invariant mass. The new trigger strategy allows for the recovery of a significant number of events, particularly at low p_T , which would have been discarded under the default selection. This effect is clearly visible in Fig. 6.2 and 6.3 and is quantitatively summarised in Tab. 6.4. Notably, most jets that fire the trigger item are also the p_T -leading, as indicated by the small number of events discarded when adding the matching requirement.

Trigger selection on data22	
no trigger selection	100%
one jet fires the trigger AND leading jet $p_T > 470$ GeV	47.46%
leading jet fires the trigger AND leading jet $p_T > 470$ GeV	47.42%
one jet fires the trigger AND leading jet $p_T > 500$ GeV	14.30%
AND invariant mass $m_{jj} > 1.3$ TeV	

Table 6.4. Fraction of data22 events selected by requiring the different trigger strategies.

The event selection is then performed as follows:

- leading jet firing the HLT_j460_a10_sd_cssk_pf_ftf_preselj225_L1J100 single jet trigger item;
- $\# \text{ jets} = 2$;
- leading jet $p_T > 470$ GeV;
- jet $p_T > 200$ GeV;
- jet mass $m > 50$ GeV;
- jet pseudorapidity $|\eta| \leq 2$.

The jet mass distributions for the signal samples are shown in Fig. 6.4. In the single

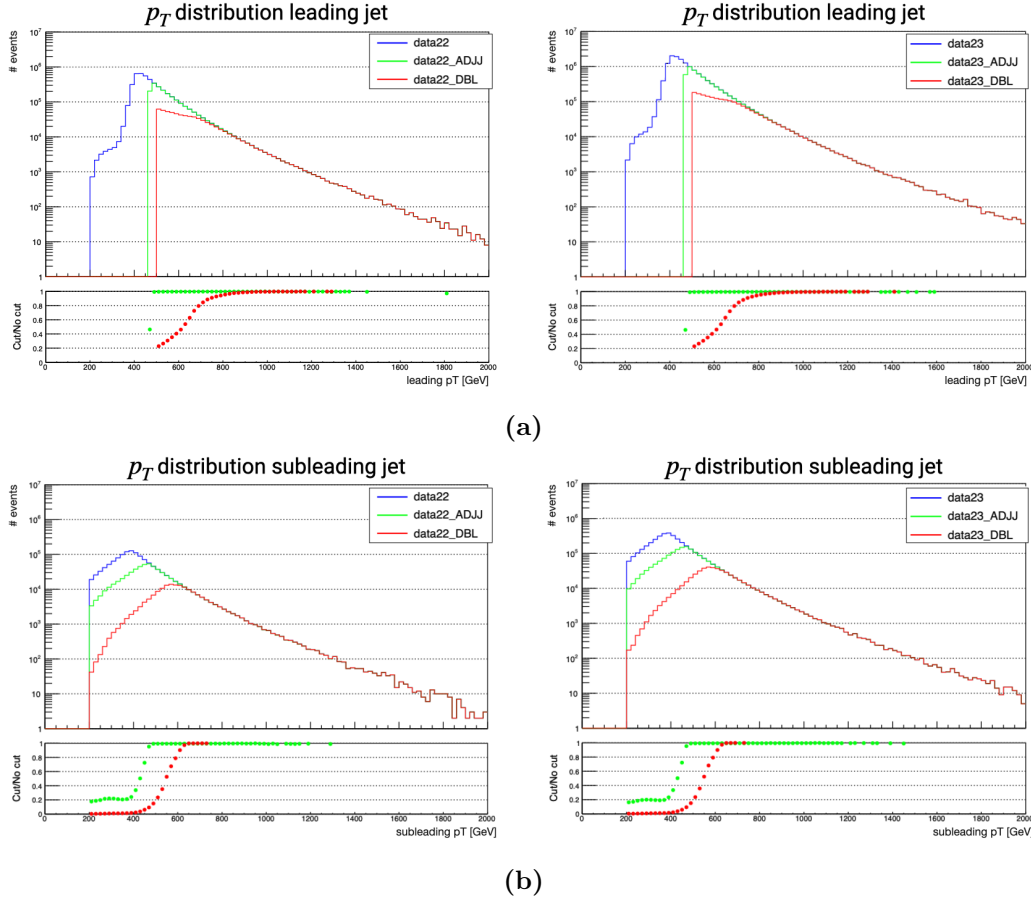


Figure 6.2. p_T distributions for the leading (6.2a) and subleading jet (6.2b) are shown for different trigger strategies. The blue curves correspond to data distributions without any trigger selection, using data from 2022 (left) and 2023 (right). The green curves correspond to the proposed trigger selection, where the leading jet must match the trigger item and satisfy $p_T > 470$ GeV. The red curves represent the old trigger strategy, requiring events to pass the trigger item, with leading jet $p_T > 500$ GeV and invariant mass $m_{jj} > 1.3$ TeV. In all plots, a lower p_T cut at 200 GeV and a jet mass cut $m > 50$ GeV are applied, as these will be included regardless of the trigger strategy adopted. In the subleading jet distributions, smearing due to the trigger not reaching the plateau is evident below 400-500 GeV. The ratio plots illustrate the fraction of discarded events per each p_T bin compared to the baseline where no cuts applied.

jet mass distributions, the peaks correspond to the mass of the particles originating the jet, that are SM W bosons for the VzWW signals (Fig. 6.4b) and the Higgs boson together with an X resonance for the VcXH models (Fig. 6.4a). The W boson peak is located close to the selection cut at $m > 50$ GeV, meaning that the event enhancement appears only in the first few low mass bins. In the VcXH models, the distribution features two distinct peaks; the first peak, always visible, corresponds to the Higgs mass. The second enhancement, associated with the X boson mass, is indistinguishable from the Higgs boson in the case where $m_X = 100$ GeV. When $m_X = 2500$ GeV, the X boson is so massive that it cannot be fully contained within a single large R jet, instead part of its mass is lost. This effects occurs whenever

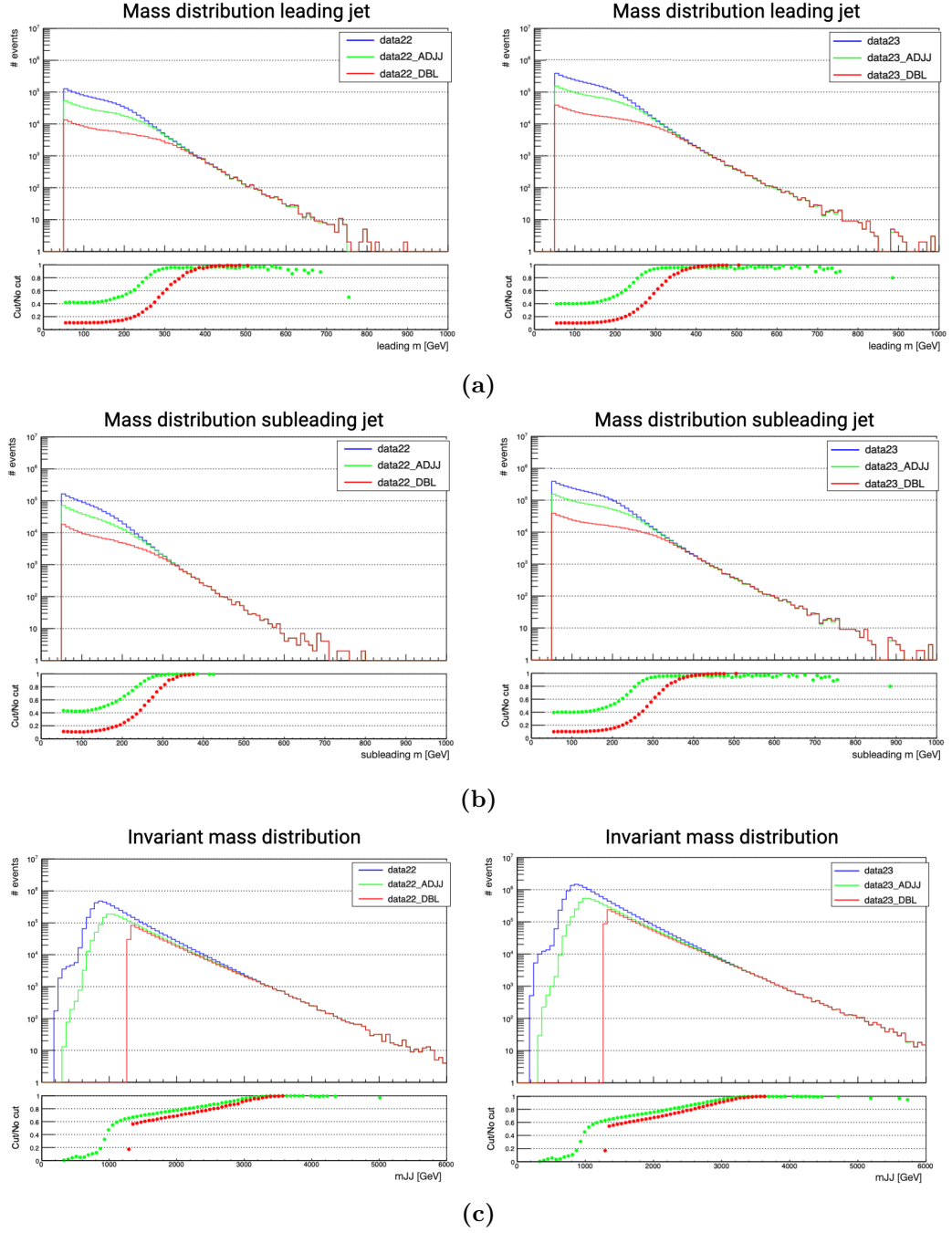


Figure 6.3. Mass distributions for the leading (6.3a), the subleading jet (6.3b) and the dijet invariant mass distribution (6.3c) are shown for different trigger strategies. The blue curves correspond to data distributions without any trigger selection, using data from 2022 (left) and 2023 (right). The green curves correspond to the proposed trigger selection, where the leading jet must match the trigger item and satisfy $p_T > 470$ GeV. The red curves represent the old trigger strategy, requiring events to pass the trigger item, with leading jet $p_T > 500$ GeV and invariant mass $m_{jj} > 1.3$ TeV. In all plots, a lower p_T cut at 200 GeV and jet mass cut $m > 50$ GeV are applied, as these will be included regardless of the trigger strategy adopted. The jet mass distributions exhibit smearing effects below 200-300 GeV due to the trigger not reaching its plateau.

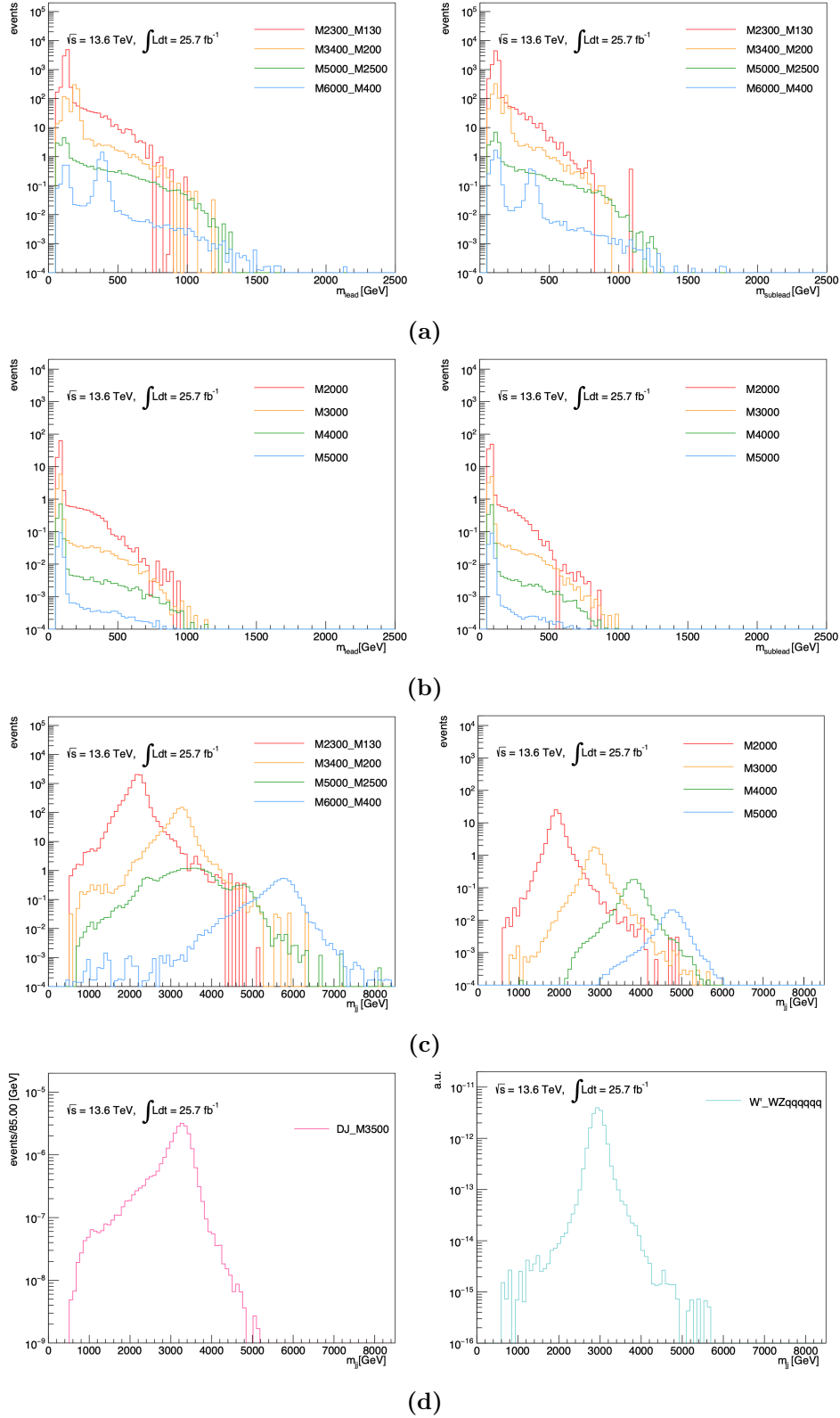


Figure 6.4. Mass distributions for the signal samples. Fig. 6.4a and 6.4b show the single jet mass distributions for the leading (on the left) and subleading (on the right) jets. The peaks around the masses of the X , H and W bosons respectively are clearly evident, but in the case $VcXH$ where $m_V/m_X \geq 0.5$ and the X boson is not fully contained in the large R jet. Fig. 6.4c and 6.4d show the invariant mass distributions. Plots are renormalised to the 2023 integrated luminosity.

the mass ratio satisfies $m_V/m_X \geq 0.5$. The dijet invariant mass distributions for all signals are displayed in Fig. 6.4c and 6.4d. In most cases, the distribution peaks at the parent resonance mass; however, in the HVT sample where $m_V/m_X \geq 0.5$, the X boson is not fully reconstructed within a single jet, leading to a missing mass contribution.

From the comparison between all MC background contributions and real data, the overall shape agreement is quite satisfactory; however, it is evident that the background prediction tends to overestimate the data, as shown in Fig. 6.5, 6.6 and 6.7.

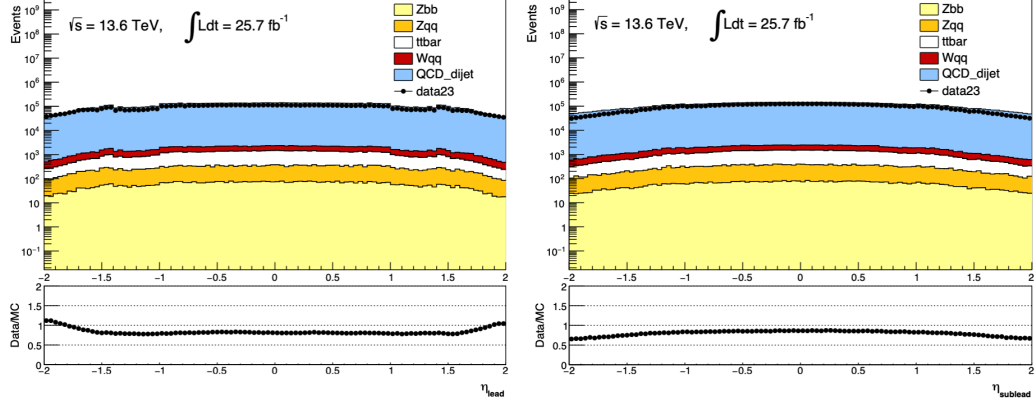


Figure 6.5. η leading and subleading distributions for all background contributions and comparison with data23. The overall shapes for data and MC are matching; however, the data-MC 0.8 ratio indicates PYTHIA8 mismodelling. Moreover, the opposite trend for leading and subleading jets at $|\eta| > 1.5$ is due to missing in situ calibrations in the MC samples.

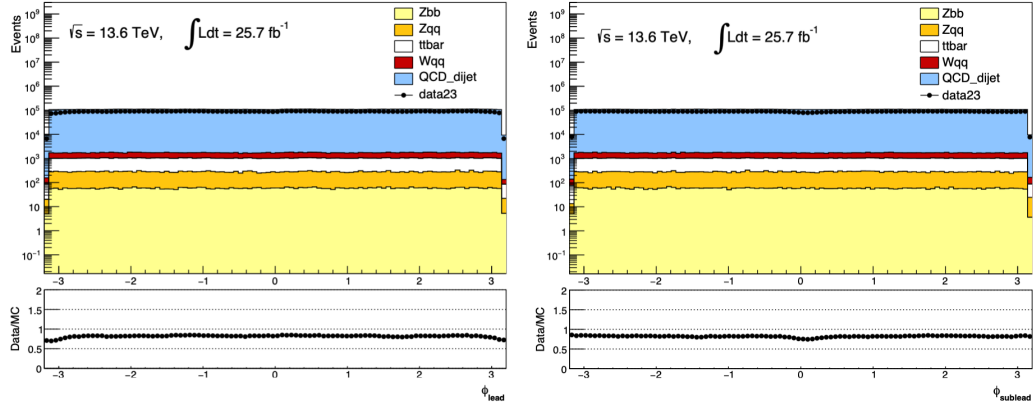


Figure 6.6. ϕ leading and subleading distributions for all background contributions and comparison with data23. The decrease of data occurrences at $|\phi| \approx \pi$ for leading jets and $\phi \approx 0$ for subleading is justified by dead modules in the tile calorimeter.

In fact, the data-to-MC ratio averaged across all distributions is approximately 0.8, primarily due to known mismodellings in PYTHIA8 which provides the dominant background contribution. The pseudorapidity distributions (Fig. 6.5) present an opposite trend between leading and subleading jets at the edges of the distributions.

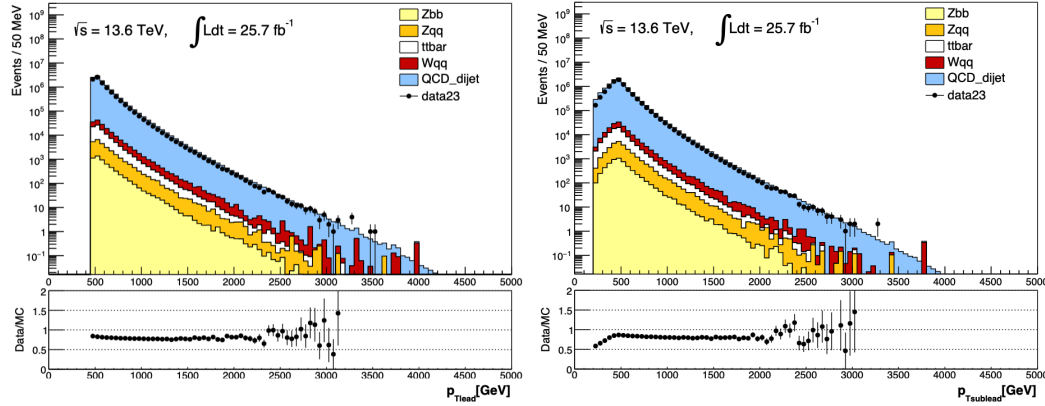


Figure 6.7. p_T leading and subleading distributions for all background contributions and comparison with data23. Data and MC shapes are in agreement; at low p_T , subleading jets in data are slightly less frequent than MC, but this is due to the trigger selection.

This discrepancy arises from missing in situ calibrations in the MC samples, particularly noticeable for $|\eta| \geq 1.5$. The effect manifests as an artificial enhancement of reconstructed leading jets, which coincides with the apparent depletion of subleading jets in the same region. The azimuthal angle distributions (Fig. 6.6) indicate a good data-MC agreement, apart from the systematic overestimation. The reduction in the event counts around $|\phi| \approx \pi$ for the leading jet distribution, mirrored at $|\phi| \approx 0$ for the subleading jet, is due to dead modules in the Tile Calorimeter. While MC simulations account for possible detector inefficiencies, the combination of the large R jet reconstruction and the single jet trigger selections makes this effect particularly sensitive. Given that two-body decays are back-to-back in ϕ , any inefficiency in the leading jet's azimuthal position $\hat{\phi}$ will be mirrored in the subleading jet at $\hat{\phi} - \pi$. To further investigate this effect, we produced analogous distributions for small R jets ($R = 0.4$) with the same trigger selection and for large R jets from DAOD_PHYS samples using a muonic trigger selection. In both cases, the reduction effect is present, but less pronounced. There exists a jet energy correction tool to compensate for detector failures; however, it often over-calibrates the jets, so we decided not to use it. While training the model on data, we do not intend to exclude events falling in these problematic regions from the training dataset, but rather we will monitor whether the AS exhibits any correlation with ϕ . Although AD techniques can be used for data and detector quality monitor, the models developed so far are designed for NP resonances search. Moreover, the applied transformation should, in principle, remove all the possible references to the spatial appearance of the jet. The transverse momentum distributions (Fig. 6.7) confirm that, aside from the overall 0.8 data-to-MC ratio, the shapes of the distributions are consistent. The decrease in subleading jets observed for $p_T < 470$ GeV is fully explained by the applied trigger selection.

6.2 ML trainings on MC dataset

The events passing the selection are used to create the datasets for training, validation and testing. To assess the discovery potential of the Transformer AD model on the given signal, the network is trained exclusively on background MC simulation.

6.2.1 Dataset strategies

The fully unsupervised AD strategy adopted with the LHC Olympics 2020 R&D dataset was to train and validate on background only events and then test on the different signals. Here, we proceeded analogously and decided to use only QCD dijet data as background, neglecting for the moment the other less dominant background contributions. QCD dijet samples are generated in separate slices and the number of events remaining after the event selection is shown in Tab. 6.5. Events in slices

QCD slice	JZ0	JZ1	JZ2	JZ3	JZ4
$\hat{p}_T$ range [GeV]	[0, 20]	[20, 60]	[60, 160]	[160, 400]	[400, 800]
# selected events	9	9	15	174 k	19.2 M
QCD slice	JZ5	JZ6	JZ7	JZ8	JZ9
$\hat{p}_T$ range [TeV]	[0.8, 1.3]	[1.3, 1.8]	[1.8, 2.5]	[2.5, 3.2]	[3.2, 3.9]
# selected events	20.3 M	15.6 M	12.0 M	13.8 M	15.8 M

Table 6.5. Number of events left after event selection per each slice of the QCD dijet background samples and its $\hat{p}_T$ range. Note that $\hat{p}_T$ is the hard scattering transverse momentum.

JZ0-2 do not have sufficient hard scattering transverse momentum $\hat{p}_T$ to meet the event selection criteria. However, due to reconstruction inefficiencies, a small number of these events still passes the selection. To avoid introducing biases, we decided to discard them. The remaining JZ3-9 slices are statistically well balanced, except for the JZ3 slice, but real data, which they represent, are naturally unbalanced, with an exponential falling transverse momentum distribution. Therefore, multiple dataset configurations can be constructed, enabling different training strategies. One approach is to train on the entire QCD MC dijet dataset while applying event weights to the loss function, thus to the AS. This method preserves the statistical power of the simulation and the expected shapes of the distributions. The AS obtained with this strategy should resemble the expected performance of the model when trained on real data. However, implementing a weighted loss function can be complex, depending on the specific loss used, and may require custom modifications. Another strategy is to perform training on a balanced dataset, ensuring that each QCD slice contributes equally and using an unweighted loss function. This method simplifies implementation and avoid introducing weights, but may lead to discrepancies when compared to training on real data. A third approach is to create a dataset by sampling events to match the data p_T distribution, eliminating the need for event weighting. This is the preferred strategy, as it better replicates

real data behaviour while minimising potential biases. We adopted the second approach, performing a full analysis on MC events to develop the necessary tools for the search. Although the results may not perfectly align with those obtained on real data, this represents an important first validation on the performance of the anomaly detector developed so far. In parallel, we are also pursuing the third strategy by extracting the transverse momentum distribution from data using the `scipy.stats.rv_histogram` function [211] from SciPy library [212]. This function generates a Probability Density Function (PDF) by assuming that the probability density is a step function corresponding to the histogram bin contents. The resulting function is continuous in the sense that each value within the histogram range has a well-defined probability; however, the PDF is not smooth, as it exhibits discontinuities at bin boundaries. Using this PDF, we randomly sampled MC events from the QCD slices in proportions that match the expected p_T distribution. The sampled QCD MC distribution, containing 1.1 M events corresponding to approximately 1 fb^{-1} , is shown in Fig. 6.8. The extrapolated MC distribution shows

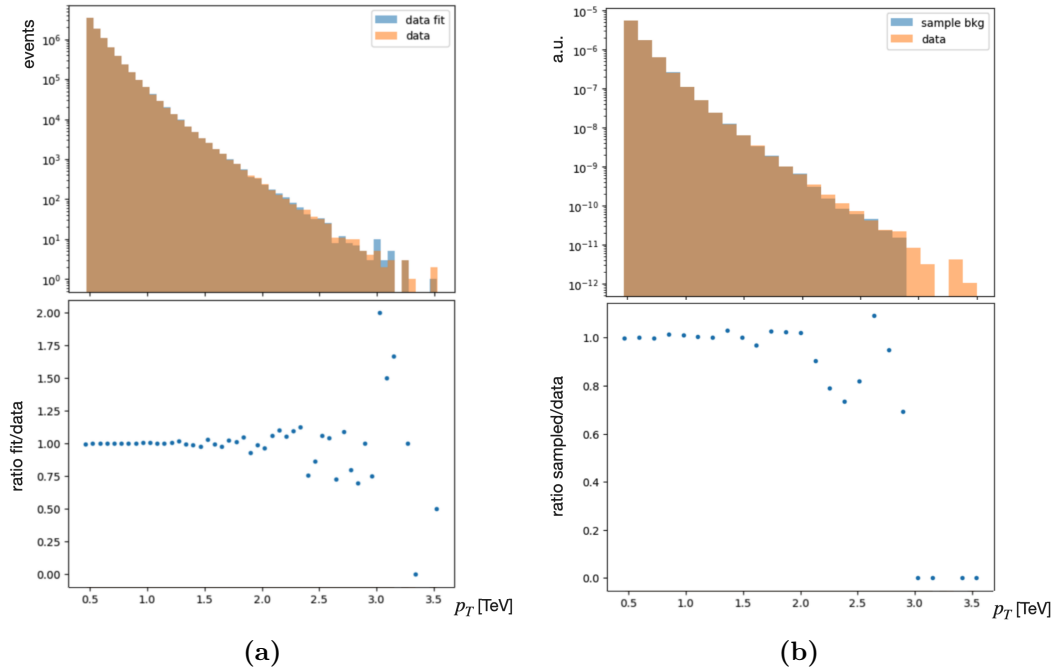


Figure 6.8. In Fig. 6.8a, the transverse momentum distribution of Run 3 data (orange) and the extracted fit (blue). The two distributions exhibit a good agreement up to 2 TeV, as indicated by the ratio plot below. In Fig. 6.8b, the p_T distribution for the data (orange) and the extracted QCD MC dataset (blue). The distributions are normalised to 1 in this plot.

good agreement with ATLAS data up to 2 TeV, beyond which statistics fluctuations become dominant due to the limited number of available events.

6.2.2 Dataset creation

The dataset used to train the Transformer AD model was constructed following the second strategy. All JZ3 events were included in the dataset, while up to 2 M events were selected from the other slices. The total number of events in the dataset is approximately 6.2 M, corresponding to 12.4 M jet-events. The dataset was then divided into training, validation and test sets, containing 80%, 10% and 10% of total events, respectively.

Since the dataset is not weighted, the jet transverse momentum distribution (Fig. 6.9) exhibits distinct peaks, one for each QCD dijet slice. Due to reconstruction ineffi-

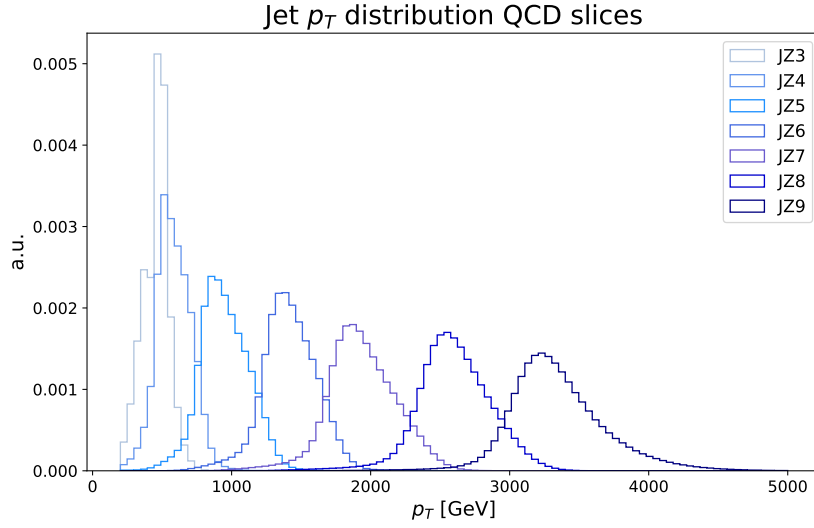


Figure 6.9. Jets’ transverse momentum distribution showing the contribution of each QCD MC dijet slice. Except for the JZ3 which contains only 174 k events, the other slices have the same number of events, being it 2 M. The distributions have been normalised to have area equal to 1.

ciencies, the distributions for each slice do not have sharp edges, but instead decrease smoothly within their corresponding hard scattering transverse momentum range. As the Transformer requires a vectorial input, each jet-event consists of up to 50 constituents, ordered by decreasing p_T constituents and zero padded where necessary. In reality, the distribution of the number of constituents, which depends on the QCD dijet slice as shown in Fig. 6.10, has mean value of approximately 40, but exhibits long tails extending up to 250 constituents. However, the constituents in the tail do not contribute significantly to the total transverse momentum of the jet, as illustrated in Fig. 6.11. The decision of limit the number of constituents to 50 was primarily motivated by considerations of resource occupancy reasons and, most importantly, the need to optimise training efficiency and loss calculation. When transitioning to the real data, however, such a strict truncation of the number of constituents is not viable. Instead, a reasonable trad-off will be implemented, likely setting the limit between 80 and 100 constituents per jet. Moreover, the number of constituents can serve as a discriminating variable, particularly for specific signals

such as DJ which are characterised by a high number of constituents.

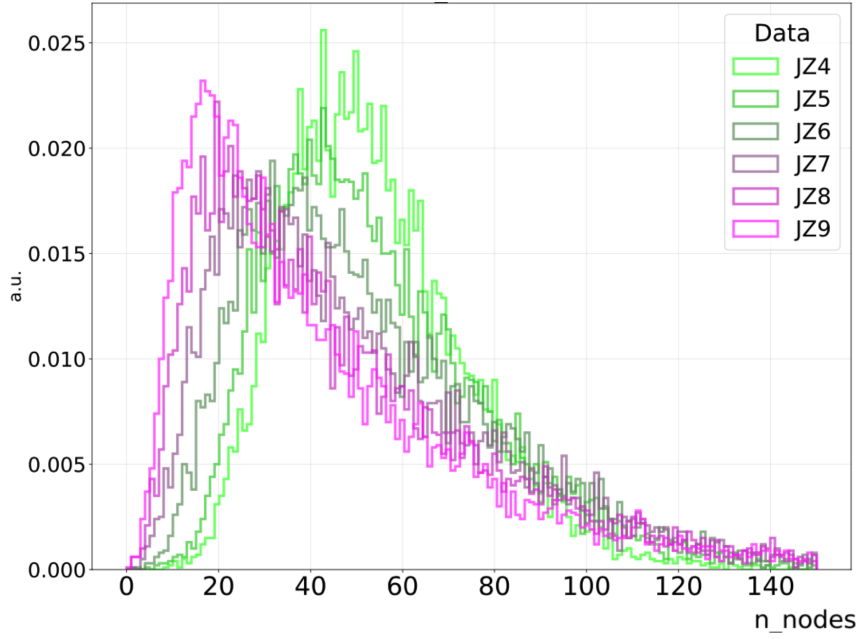


Figure 6.10. Distribution of the number of nodes in the graph dataset per each QCD dijet slice contribution. The number of nodes corresponds to the number of constituents in the jet-event.

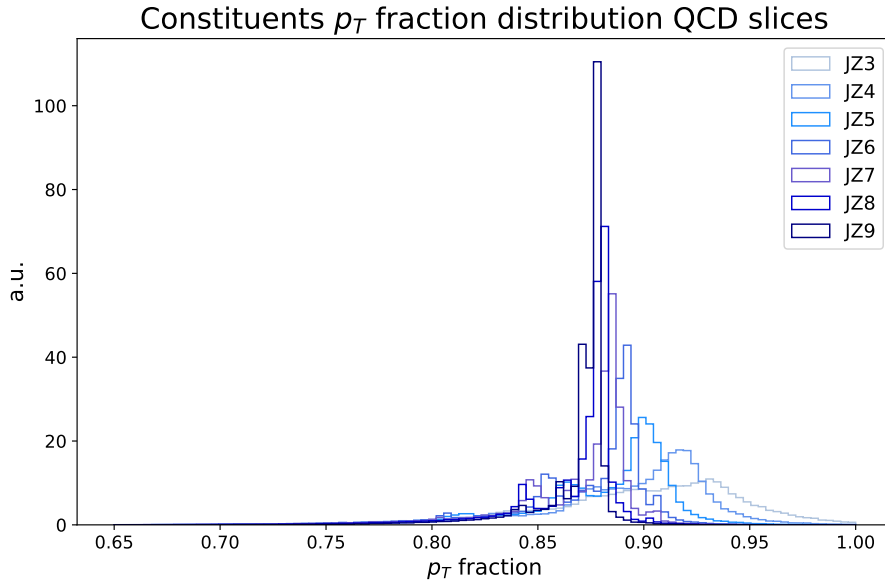


Figure 6.11. Fraction of transverse momentum carried by the 50 constituents over the jet p_T , showed per each QCD dijet slice separately. The first 50 constituents ordered with descendent p_T criterion contain approximately 90% of the total transverse momentum of the jet.

Each constituent in the dataset is characterised by its transverse momentum, pseudorapidity, azimuthal angle and associated mass. The taste variable was initially included in the dataset, but was found not to enhance the model's performance. This is because it assumes only three integer values, whereas the other variables are continuous, making it difficult for the model to exploit effectively. Consequently, the taste is not used as a feature for the constituents, but is left for future studies. The transformation described in App. B is applied to all constituents, to remove dependencies on mass and transverse momentum, ensuring that the dataset remains invariant under rotations and translations. The kinematic distributions of the constituents for both background and signal datasets are shown in Fig. 6.12, 6.13, 6.14 and 6.15.

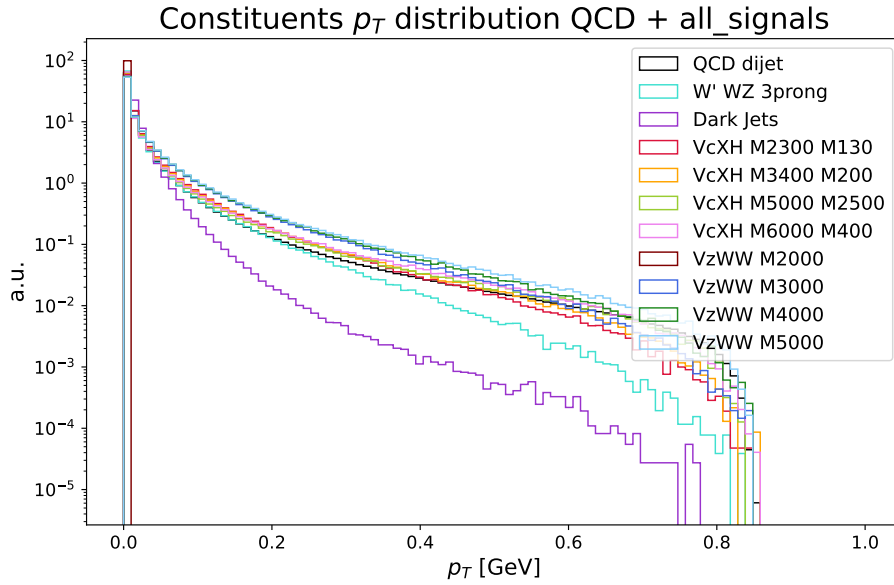


Figure 6.12. Transverse momentum distributions for the QCD dijet background in black and all the other signals. All constituents have undergone the transformation.

The transverse momentum distributions of the signals (Fig. 6.12) appear similar for most signal samples, except for the DJ and $W' \rightarrow WZ$ datasets, which is expected given the good performance of the transformation. The pseudorapidities distributions for both background and signal samples (Fig. 6.13) are also comparable, with only minor differences between datasets. This is a direct consequence of the transformation, which forces all constituents into a well-defined structure, positioning the hardest constituent of the jet at $\eta = 0$. Conversely, the azimuthal angle distributions (Fig. 6.14) exhibit significant differences among the samples. The background, which consisting solely of QCD dijets and is expected to be predominantly 1prong, displays two peaks, although the peak at positive ϕ values is significantly more pronounced than that at negative ϕ , aligning with a 1prong substructure. The VcXH and VzWW samples exhibit two distinct and well-defined peaks, indicative of their 2prong nature. However, in the VcXH sample

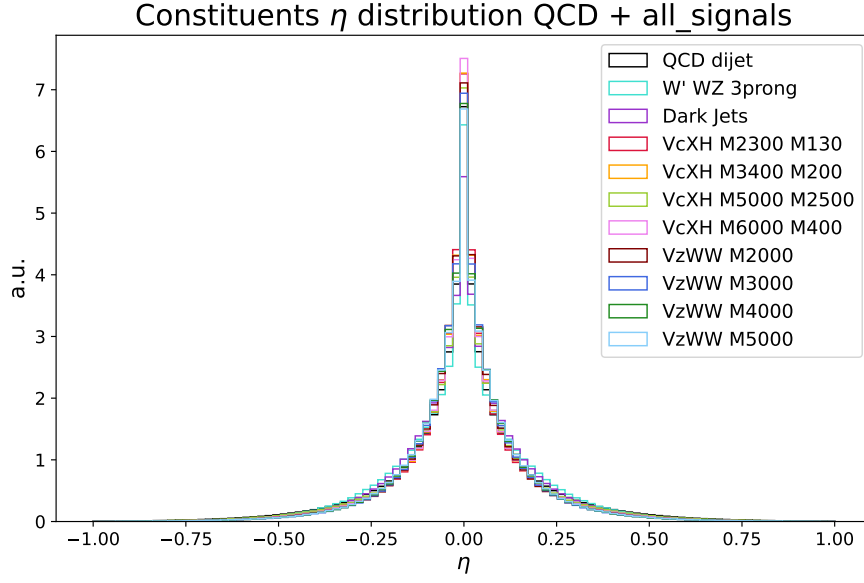


Figure 6.13. Pseudorapidity distributions for the QCD dijet background in black and all the other signals. All constituents have undergone the transformation.

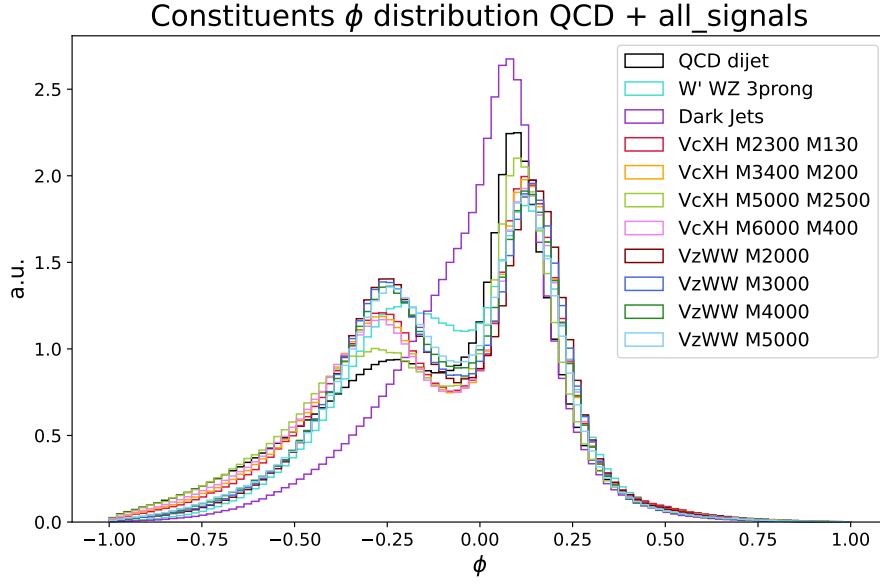


Figure 6.14. Azimuthal angle distributions for the QCD dijet background in black and all the other signals. All constituents have undergone the transformation.

with masses of $m_V = 5 \text{ TeV}$ and $m_X = 2.5 \text{ TeV}$ the two peaks are less prominent, resembling the 1prong QCD dijet distribution. The constituent ϕ distribution for the $W' \rightarrow WZ$ sample, which decays into three quarks or antiquarks, also exhibits two peaks though they are less pronounced, consistent with the behaviour observed

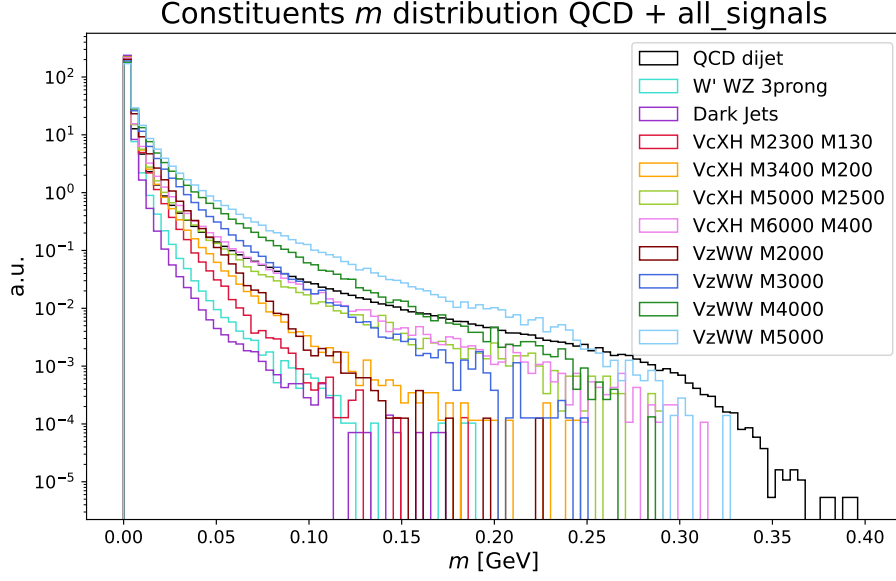


Figure 6.15. Mass distributions for the QCD dijet background in black and all the other signals. All constituents have undergone the transformation.

in the LHC Olympics 2020 R&D 3prong dataset (see Sec. 5.3.1). In contrast, constituents from the DJ sample display a distinctive shape with a single peak. The mass distributions (Fig. 6.15) vary across the different signals, as with the transformation they have all been rescaled by a constant factor $\hat{m} = \frac{m_0}{m}$ where m_0 is fixed to 0.25 GeV and m represents the jet mass. This variation arises from differences in the mass range of each signal sample. In principle, this should not pose an issue, as the model is trained only on background and there is no explicit indication that this variable is discriminative. However, it is crucial to verify whether the model inadvertently learns this dependence and exploits it to distinguish signal events. Before training the model, all signal and background datasets are normalised using the usual standardisation function. Nonetheless, it is evident that certain constituent variables, such as the constituents' mass and azimuthal angle, are more discriminating than other. Since normalisation is primarily required for training stability and convergence speed, issues that the Transformer AD architecture do not experience, it is possible to leverage the standardisation to enhance specific features. In particular, the azimuthal angle is the only viable candidate, as modifying the constituent mass would introduce a dependence that we have made significant efforts to avoid, while the other variables are not discriminative. To achieve this, we introduce a parameter α , referred to as the ϕ enhancement parameter, which is applied during the normalisation of the constituents' azimuthal angles:

$$\phi \longrightarrow \phi' = \frac{\phi - \langle \phi \rangle}{\alpha \cdot \sigma_\phi} \quad (6.3)$$

where $\langle \phi \rangle$ is the mean ϕ value and σ_ϕ is its standard deviation. The ϕ enhancement parameter can be considered as an additional model hyperparameter that must be

optimised alongside other architecture-related parameters. The smaller the α value, the wider the range spanned by the azimuthal angle of the constituents. The model, which is trained to reconstruct the input elements, will be more penalised if it fails to reconstruct ϕ accurately, compared to the other constituent variables. Several training runs have been conducted with ϕ enhancement values of 0.1, 0.05, 0.025 and 0.01. The best configuration was found to be $\alpha = 0.025$. The background constituent distributions after the standardisation are shown in Fig. 6.16. The ϕ variable is spread over a significantly larger range, yet its shape remains unchanged; the other variable are left unaffected. Various techniques exist to emphasise specific features during training, such as incorporating weights into the loss function to assign greater importance to a given variable. However, the approach presented here is both straightforward to implement and effective.

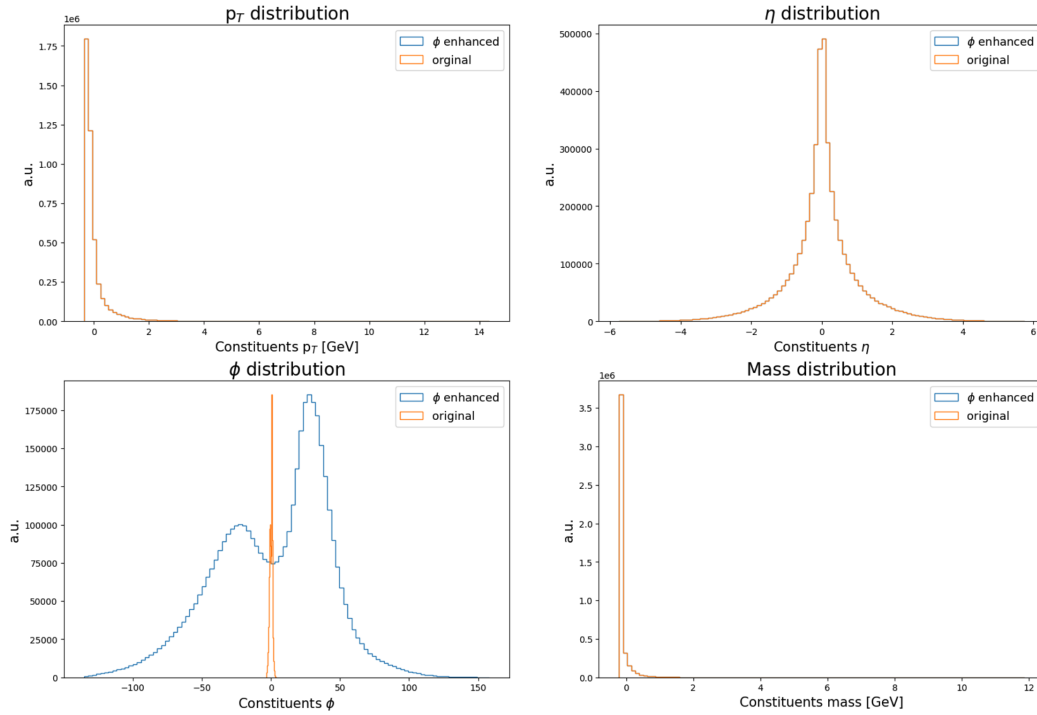


Figure 6.16. Background distributions of the constituents’ features after standardisation, with (in blue) and without (in orange) azimuthal angle enhancement. The α parameter, set to 0.025, is applied exclusively to the ϕ variable, while the other distributions remain unaltered. The azimuthal angle distribution retains its shape in both cases, but spans different value ranges. Zero padded constituents have been removed from the plots.

6.2.3 Transformer performance

The Transformer AD architecture is the same as described in Sec. 5.4.2. The learning rate is set to 10^{-6} , the dropout parameter is 0.3 and adaptive learning rate scheduling is applied at 50%, 75% and 90% of the training epochs with a decay factor of $\gamma = 0.2$. No early stopping has been implemented and models have been trained for both 10 and 50 epochs. The training is conducted using the MSE loss function,

enabling the model to learn by minimising the reconstruction error of background events. However, with this approach assessing whether the ML network is overfitting, meaning it relies excessively on the given input data, is challenging. The loss values continuously decrease, as the model extracts more information from the data and refines. Nevertheless, this does not necessarily indicate that the strategy results in the best separation between background and signal events. For this reason, the number of training epochs is a critical factor in determining the model performance. We observed that training for 50 epochs yielded worse results compared to training for only 10 epochs. This is likely due to the high statistical power of the dataset (12.4 M jet-events in total), which facilitates rapid loss convergence within the first few epochs. The training and validation losses for the 10-epoch model are shown in Fig. 6.17a. The jet-level AS distribution obtained from the training (Fig. 6.17b)

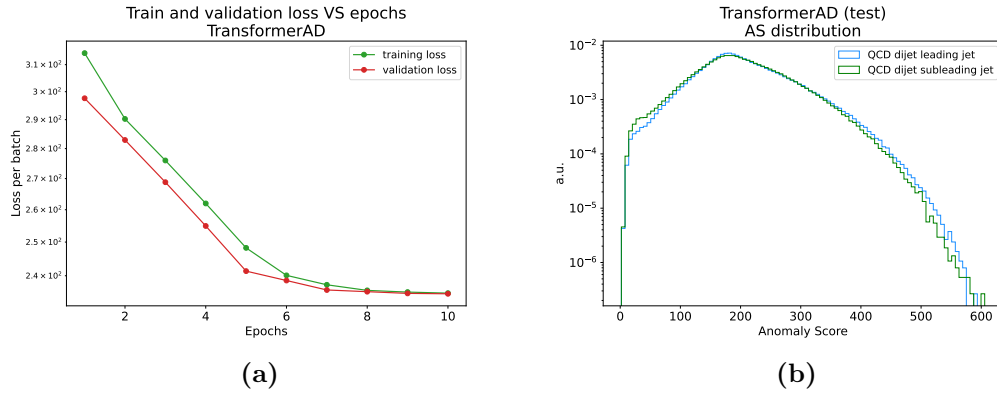


Figure 6.17. In Fig. 6.17a, training and validation values for the MSE loss function are shown as increasing the epochs. The training is conducted for 10 epochs. In Fig. 6.17b, the jet-level AS distributions separated for leading and subleading jets.

ranges between 0 and 600, with a peak at approximately 170. No significant differences are observed between the leading and subleading jet distributions, indicating that the model's performance does not depend on the jet transverse momentum. To further verify that the model has not learnt unintended dependencies on kinematic variables, a study on the correlation between the AS and these variables have been conducted. The model exhibits no dependence on the transverse momentum (Fig. 6.18a), with intensity peaks in the distribution corresponding to the p_T peaks of the different QCD dijet slices, as also observed in Fig. 6.9. This behaviour is more evident in the profile plot (Fig. 6.18b), where the average jet-level AS value shows slight bumps when transitioning from one slice to another. No dependence is observed on pseudorapidity or azimuthal angle (Fig. 6.18c and 6.18d). The jet-level AS exhibits slight correlation with the mass (Fig. 6.18e) at low mass values, although this could be an artificial effect due to the large number of events populating the first few bins. Moreover, the correlation plot with the D_2 jet variable, defined in Eq. 3.13 (Fig. 6.18f), demonstrates no significant correlation, indicating that the model is not learning one of the most discriminating variables typically used in traditional analyses. Overall, the model has so far demonstrated not dependence on the kinematic properties of the input jet, as desired.

The performance of the Transformer AD model is assessed using the AUC: the closer

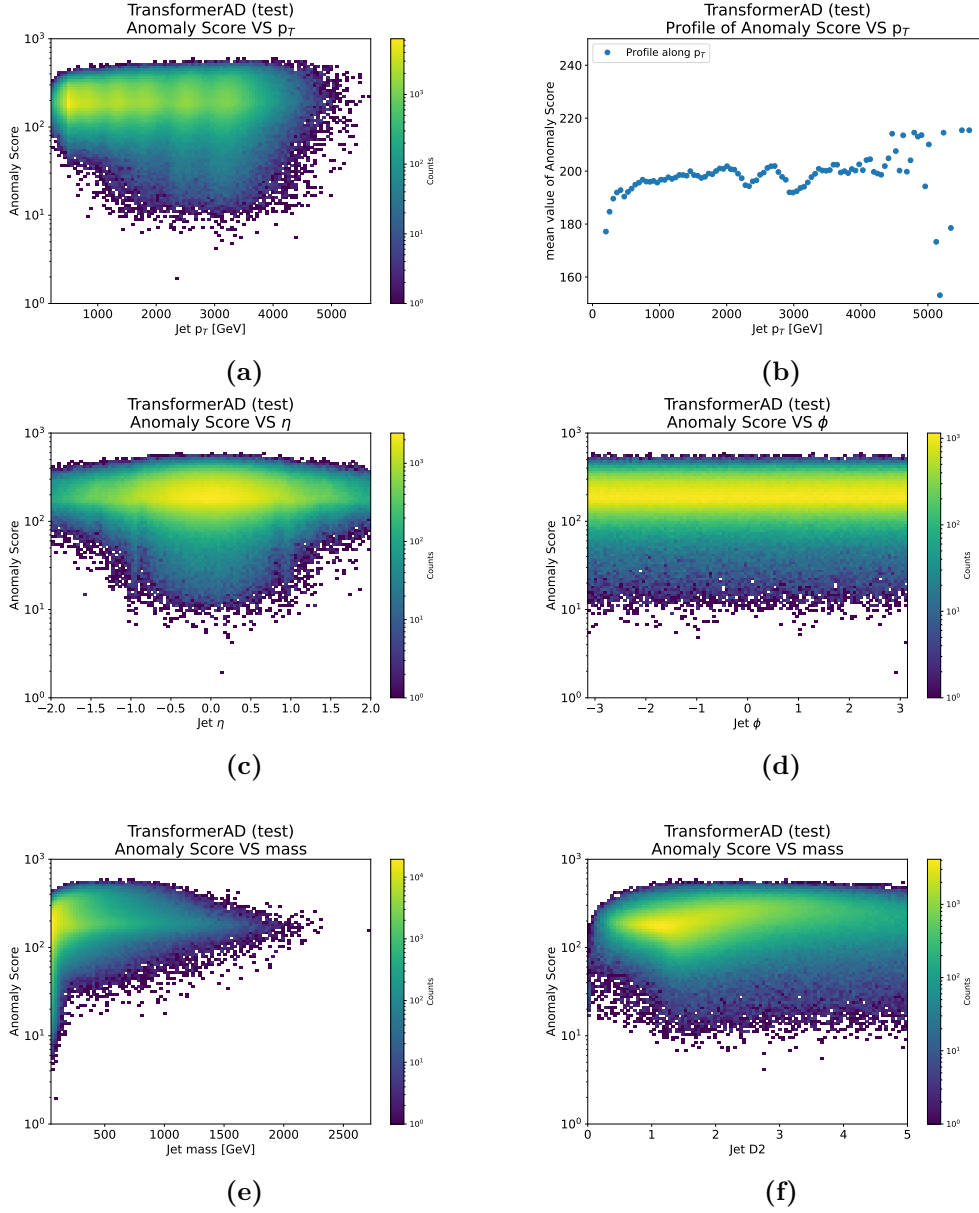


Figure 6.18. Correlation plots between the jet-level AS and kinematical variables. Fig. 6.18a illustrates the dependence of the AS on transverse momentum, while Fig. 6.18b presents the profiling of the same plot along p_T . The profiling is obtained by computing the average AS value for each p_T bin. The bumps in the profile plot and the intensity peaks in the AS distribution reflect the non-uniform p_T spectrum of the QCD dijet slices. Fig. 6.18c, 6.18d, 6.18e and 6.18f depict the correlations between AS and jet’s pseudorapidity, azimuthal angle, mass and D_2 , respectively. The results indicate that the jet-level AS is not significantly dependent on these kinematic quantities.

the value is to 50%, the less discriminative the model is. To evaluate this, the model was first tested on statistically independent samples containing events from individual QCD slices. It is essential to ensure that the model does not extract too much

information from a specific slice, as the slight AS dependence on p_T might suggest. The AUC values, summarised in the first row of Tab. 6.6, show comparable results for slices JZ4-9, with slightly higher values for JZ3. This is likely due to the lower statistical contribution of JZ3 to the training dataset (approximately one twentieth of the other slices), resulting in reduced model performance when reconstructing these jets effectively. Nonetheless, the obtained results are all close to 50%, indi-

QCD slice	JZ3	JZ4	JZ5	JZ6	JZ7	JZ8	JZ9
Jet-level AUC	56.3%	50.3%	50.6%	51.2%	51.2%	50.2%	51.8%

Table 6.6. AUC values of the Transformer AD model trained on all QCD dijet slices simultaneously and then tested on the single slices separately. The testing dataset are statistically independent from those used during training and validation.

cating that the model does not favour any particular slice over the others and has learnt an optimal representation of the input data. Moreover, models trained for a higher number of epochs exhibit greater fluctuations between slices, suggesting that they are overfitting specific patterns in the data.

When tested on datasets different from the QCD dijet samples on which it was trained, the model produces the results summarised in Tab. 6.7. Although in reality W +jet and $t\bar{t}$ events are considered background, in this case, the model has been trained exclusively on QCD dijet samples. Consequently, any sample that does not originate from QCD dijet can be treated as a signal. Jets originating from $t\bar{t}$ decays have a 3prong structure but not always the three subjects are reconstructed within the same large R jet. However, the model's performance on these events is similar to that obtained when using $W' \rightarrow WZ$ samples. The performance decreases when tested on the W +jet sample, which is also anticipated, as one jet originates from the decay of the W boson and has a 2prong structure, while the other follows a 1prong topology. Since no tagging is applied to distinguish the jet from the W boson decay, the jet-level AUC value is averaged between the two topologies, resulting in lower overall performance. Among the 2prong signals, DJ and $V \rightarrow WW$ samples exhibit higher AUC values, ranging between 70% and 80%. Conversely, the $V \rightarrow XH$ samples show lower performance, with AUC values between 50% and 60%. The jet-level AS distributions are similar in shape across the various tested signals, as illustrated in Fig. 6.19. The distributions exhibit one peak at different AS values which determines the discrimination strength, but always remains to the left of peak associated with the background. Additionally, a tail extends towards higher AS values, while a small fraction of signal jet-events assume AS values below 50. As for the AS distributions with LHC Olympics 2020 R&D dataset (see Sec. 5.6.6), the signal distribution has a peak with lower AS value compared to the background. Since the loss function employed is an MSE, this means that the model is reconstructing more accurately the signal events than the background ones. However, as far as no signal is used during the training phase, it can be attributed to a network's feature.

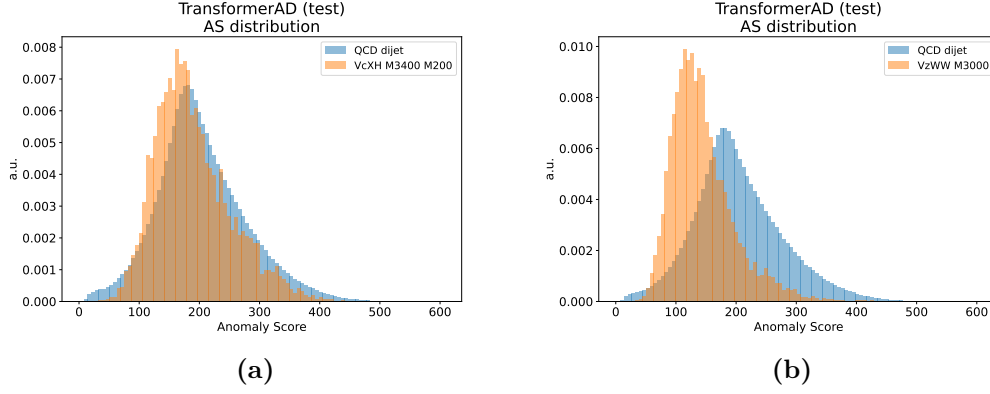


Figure 6.19. Jet-level AS distributions for one of the worse performing samples, the $V \rightarrow XH$ with $m_V = 3.4$ TeV and $m_X = 200$ GeV (on the left), and one of the best, the $V \rightarrow WW$ with $m_V = 3$ TeV (on the right).

6.2.4 Event-level Anomaly Score

An event-level AS can be derived from the jet-level AS values of the jets belonging to the same event, following a similar approach to that described in Sec. 5.6.6. However, in this case the event-level AS is defined as the product of the individual jet-level AS, rather than their average, as this results in better discrimination. The AUC values when tested on QCD slices are lowered by 1-4% using the product of jet-level AS rather than the average. The event-level AS distribution, shown in

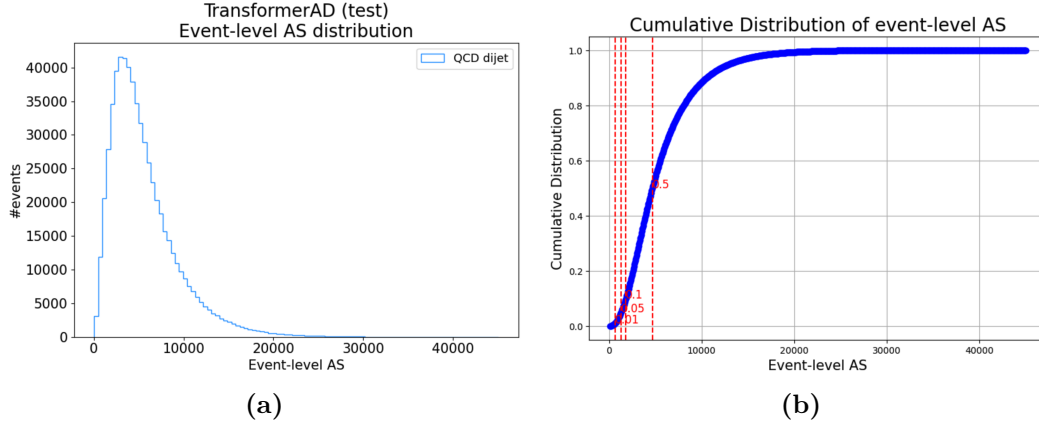


Figure 6.20. Event-level AS distribution on background QCD dijet events (on the left). Cumulative curve of the same distribution (on the right). The four WP are shown on the cumulative plot at 1%, 5%, 10% and 50% of background efficiency, corresponding to a rejection factor of 100, 20, 10 and 2, respectively.

Fig. 6.20a, exhibits more pronounced features compared to the jet-level distribution. It peaks at approximately 3000, has an asymmetric shape and extends into long tails reaching up to 40000. The AUC value obtained from the event-level AS distributions for the various signals are summarised in Tab. 6.7 and shown in Fig. 6.21. As expected, the performance improves when transitioning from jet-level to event-level

Signal	$W + \text{jet}$	$t\bar{t}$	Dark Jets	$W' \rightarrow WZ$
Jet-level AUC	54.8%	61.7%	70.0%	59.3%
Event-level AUC	56.1%	65.3%	76.8%	66.0%

Signal	$m_V = 2 \text{ TeV}$	$m_V = 3 \text{ TeV}$	$m_V = 4 \text{ TeV}$	$m_V = 5 \text{ TeV}$
$V \rightarrow WW$				
Jet-level AUC	73.1%	76.2%	77.6%	79.1%
Event-level AUC	78.2%	81.4%	82.7%	84.0%

Signal	$m_V = 2.3 \text{ TeV}$	$m_V = 3.4 \text{ TeV}$	$m_V = 5 \text{ TeV}$	$m_V = 6 \text{ TeV}$
$V \rightarrow XH$	$m_X = 130 \text{ GeV}$	$m_X = 200 \text{ GeV}$	$m_X = 2.5 \text{ TeV}$	$m_X = 400 \text{ GeV}$
Jet-level AUC	57.5%	56.6%	52.7%	56.5%
Event-level AUC	60.1%	59.3%	54.2%	59.8%

Table 6.7. AUC values of the Transformer AD model trained on all QCD dijet slices simultaneously and then tested on the different signal samples. Jet-level and event-level AUC values are shown.

AS, yielding an increase in discriminative power of 5-10%, with the only exception being the $V \rightarrow XH$ signal with $m_V = 3 \text{ TeV}$, where the gain is limited to 3%. As observed at the jet level, the peak of the event-level AS distribution for all signals remains positioned to the left of the background peak. This enables the definition of

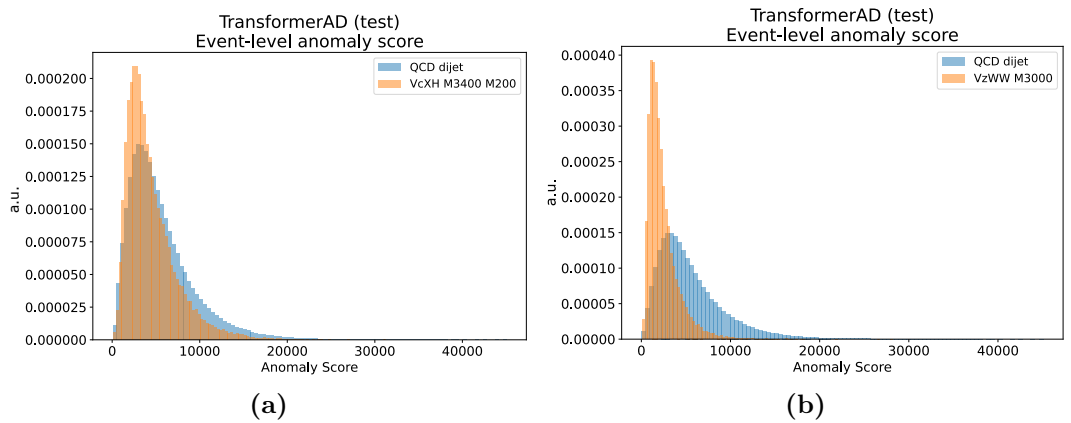


Figure 6.21. Event-level AS distributions for one of the worse samples, the $V \rightarrow XH$ with $m_V = 3.4 \text{ TeV}$ and $m_X = 200 \text{ GeV}$ (on the left), and one of the best, the $V \rightarrow WW$ with $m_V = 3 \text{ TeV}$ (on the right).

WP at specific background rejection factors. The values of the event-level AS corresponding to the chosen background efficiencies are determined using the cumulative curve of the event-level background AS distribution (Fig. 6.20b). The thresholds are set at the points where the desired background efficiency is achieved. Events are retained for the analysis if satisfying the following condition:

$$AS_{\text{ev}} \leq AS_{\text{WP}} \quad (6.4)$$

The selected WP values are summarised in Tab. 6.8, where the first rows refer to the background, the others to signals. A WP corresponding to a rejection factor 1000 was considered, but the signal efficiency of that point was too low to allow for a meaningful statistical analysis. Signal MC samples have a limited statistics, therefore if the number of selected events is too low, fluctuations become dominant. Almost all signals exhibit an efficiency higher than the corresponding background efficiency, as desired. The only exceptions are the $V \rightarrow XH$ signals at the WP corresponding to rejection factor 100. As visible also in Fig. 6.21a, the event-level AS distribution for these signals starts rising at AS values slightly higher than the background distribution, resulting in worse signal efficiencies.

Event-level AS	808	1525	1954	3814
WP rejection factor	100	20	10	2
Background efficiency	1%	5%	10%	50%
Signal efficiency				
$V \rightarrow WW, m_V = 2 \text{ TeV}$	5.0%	26.8%	41.6%	81.4%
$V \rightarrow WW, m_V = 3 \text{ TeV}$	8.8%	36.2%	51.1%	85.0%
$V \rightarrow WW, m_V = 4 \text{ TeV}$	13.9%	41.9%	55.6%	86.1%
$V \rightarrow WW, m_V = 5 \text{ TeV}$	18.4%	46.8%	59.4%	87.0%
$V \rightarrow XH, m_V = 2.3 \text{ TeV}$ $m_X = 130 \text{ GeV}$	0.9%	7.5%	15.0%	52.6%
$V \rightarrow XH, m_V = 3.4 \text{ TeV}$ $m_X = 200 \text{ GeV}$	1.1%	8.3%	15.6%	52.2%
$V \rightarrow XH, m_V = 5 \text{ TeV}$ $m_X = 2.5 \text{ TeV}$	1.7%	8.2%	14.2%	44.4%
$V \rightarrow XH, m_V = 6 \text{ TeV}$ $m_X = 400 \text{ GeV}$	2.1%	11.0%	18.8%	52.7%
DJ, $m_{Z'} = 3.5 \text{ TeV}$	5.2%	22.4%	35.9%	80.4%
$W' \rightarrow WZ, m_{W'} = 3 \text{ TeV}$ $m_W = 200 \text{ GeV}, m_Z = 400 \text{ GeV}$	3.4%	13.5%	21.8%	61.4%

Table 6.8. WPs with corresponding event-level AS values, background rejection factors, background efficiencies and signal efficiencies.

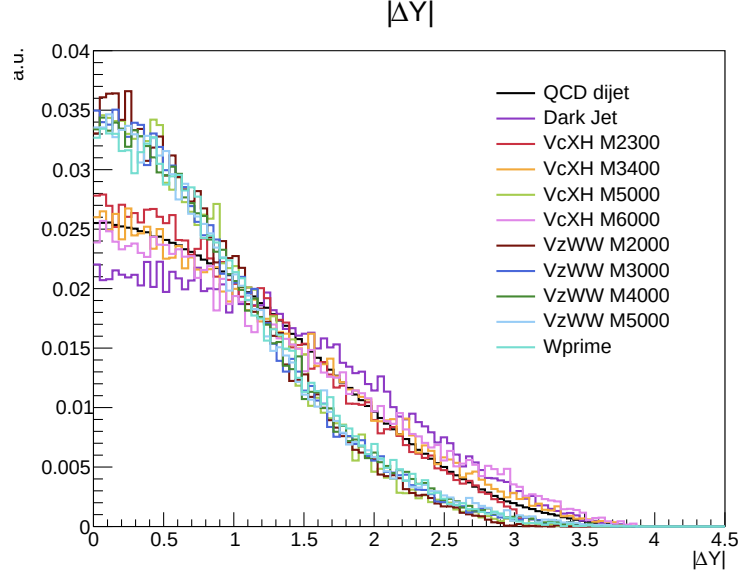


Figure 6.22. Distribution in absolute difference of rapidity between the two jets in the event, $|\Delta Y|$, for QCD dijet and signal events.

6.3 Results for the MC analysis

Once the WPs for the event-level AS have been set, the analysis on MC simulation follows a straightforward approach. In the final analysis using real ATLAS Run 3 data, the signal region (SR) will be defined by applying a cut on the absolute difference of jet rapidities, $|\Delta Y| = |y_{jet1} - y_{jet2}|$. One of the fundamental assumptions in this search is that the signal should be resonant and $|\Delta Y|$ is particularly sensitive to the decay topology of the process. The two jets originating from the BSM resonance produced from the s -channel are expected to exhibit lower $|\Delta Y|$ values compared to the predominantly t -channel background processes. This feature was also exploited in previous bump hunt dijet searches, such as Ref. [209]. The $|\Delta Y|$ distributions for the QCD dijet background and all signal samples are shown in Fig. 6.22. Signals corresponding to the $V \rightarrow WW$ topology and $W' \rightarrow WZ$ exhibit a pronounced peak at $|\Delta Y| = 0$ and lower tails, characteristic of resonant production. Conversely, $V \rightarrow XH$ signals behave more similarly to the QCD dijet background, while the DJ sample follows a different trend, featuring a lower peak but a more pronounced tail. A cut on $|\Delta Y| = 1.2$ is applied to define the analysis regions:

- **Signal Region:** $|\Delta Y| < 1.2$ and $AS_{ev} \leq threshold_{WP}$;
- **Control Region:** $|\Delta Y| \geq 1.2$ and $AS_{ev} \leq threshold_{WP}$.

The efficiencies for both signal and background, considering the event-level AS thresholds and the additional $|\Delta Y| < 1.2$ requirement, are summarised in Tab. 6.9. The additional requirement in $|\Delta Y|$ has the effect of increasing the background rejection factors of approximately 45-50% across all WPs. At the same time, some

Event-level AS + $ \Delta Y \leq 1.2$	808	1525	1954	3814
WP rejection factor	204	36.5	18.2	3.8
Background efficiency	0.5%	2.7%	5.5%	26.1%
Signal efficiency				
$V \rightarrow WW, m_V = 2 \text{ TeV}$	4.4%	22.3%	33.9%	63.2%
$V \rightarrow WW, m_V = 3 \text{ TeV}$	7.7%	29.7%	40.8%	64.6%
$V \rightarrow WW, m_V = 4 \text{ TeV}$	11.8%	33.8%	43.7%	64.4%
$V \rightarrow WW, m_V = 5 \text{ TeV}$	15.8%	37.5%	46.8%	65.3%
$V \rightarrow XH, m_V = 2.3 \text{ TeV}$ $m_X = 130 \text{ GeV}$	0.7%	5.8%	11.0%	35.5%
$V \rightarrow XH, m_V = 3.4 \text{ TeV}$ $m_X = 200 \text{ GeV}$	0.9%	6.2%	11.3%	33.9%
$V \rightarrow XH, m_V = 5 \text{ TeV}$ $m_X = 2.5 \text{ TeV}$	1.4%	6.6%	11.1%	34.0%
$V \rightarrow XH, m_V = 6 \text{ TeV}$ $m_X = 400 \text{ GeV}$	1.5%	8.1%	13.4%	33.2%
DJ, $m_{Z'} = 3.5 \text{ TeV}$	2.7%	11.6%	18.7%	43.9%
$W' \rightarrow WZ, m_{W'} = 3 \text{ TeV}$ $m_W = 200 \text{ GeV}, m_Z = 400 \text{ GeV}$	2.6%	10.3%	16.7%	45.5%

Table 6.9. WP with corresponding event-level AS values, background rejection factors, background efficiencies and signal efficiencies, all calculated with the additional requirement $|\Delta Y| \leq 1.2$.

signal events are inevitably removed with the $|\Delta Y| \leq 1.2$ selection. However, even in the worst-case scenario represented by the DJ signal, the residual number of raw events after applying the cut is still sufficient for a meaningful statistical analysis. To exploit the well-known PDF to fit signals and background samples for Bump Hunt, it is crucial that the invariant mass distributions retain similar shapes before and after applying cuts on the event-level AS and $|\Delta Y|$. As shown in Fig. 6.23, the invariant mass distributions after only the event-level AS selection have similar shapes, differing only by a multiplicative factor, as desired. However, the additional $|\Delta Y|$ cut introduces a slight dependence on the invariant mass. While this effect is inevitable, it does not pose a significant issue for the Bump Hunt search.

6.3.1 Limits on signal production cross-sections

To determine the upper limits on the signal production cross-sections, we preliminarily performed a template fit analysis on the dijet invariant mass using the HISTFACTORY package [213]. In this study, Asimov background-only MC simulation is employed in place of the real Run 3 ATLAS data. Nominal samples do account for a limited set of systematic uncertainties that do not include detector-related effects, which are studied separately. Specifically, background modelling and theoretical cross-section uncertainties on signals are introduced through the histogram normalisation parameters α_{bkg} and α_{sig} , with a 5% variation respectively, as a first attempt of introducing systematic uncertainties. Additionally, a 2% uncertainty is assigned to the luminosity estimation $\mathcal{L}_0$, as expected from Run 3 data [214]. Under these assumptions, the HISTFACTORY tool constructs the following probability density function:

$$\begin{aligned}
 P(n_i | \mu, \alpha_{\text{bkg}}, \alpha_{\text{sig}}) = & \prod_{i \in \text{bins}} [\text{Pois}(n_i | \mu \alpha_{\text{sig}} S_i + \alpha_{\text{bkg}} B_i)] \\
 & \cdot G(\alpha_{\text{sig}} | 1, 0.05) \cdot G(\alpha_{\text{bkg}} | 1, 0.05) \\
 & \cdot G(\mathcal{L}_0 | 1, 0.02)
 \end{aligned} \tag{6.5}$$

where n_i represents the observed event count in the i^{th} bin, while S_i and B_i correspond to the expected signal and background contributions, respectively. The parameter of interest is the signal strength μ , defined as the ratio of the observed cross-section to the predicted SM cross-section ($\mu = \frac{\sigma_{\text{obs}}}{\sigma_{\text{SM}}}$). All nuisance parameters included are assumed to follow Gaussian distributions.

The total likelihood for signal and background model is given by the product of the Poissonian probabilities $P(n_i | \mu, \alpha_{\text{bkg}}, \alpha_{\text{sig}})$ for all bins. A profile likelihood ratio test statistics is used to calculate the confidence intervals on the signal strength with the CL_S method. The CL_S method determines the confidence level intervals of the signal hypothesis as:

$$\text{CL}_S = \frac{\text{CL}_{S+B}}{\text{CL}_B} \tag{6.6}$$

where CL_{S+B} is the confidence level of the compatibility between the invariant mass distribution of the dataset (containing only background) and the signal plus background hypothesis and CL_B is the confidence level of the background only hypothesis. The upper limits on the signal cross-section are implemented with the

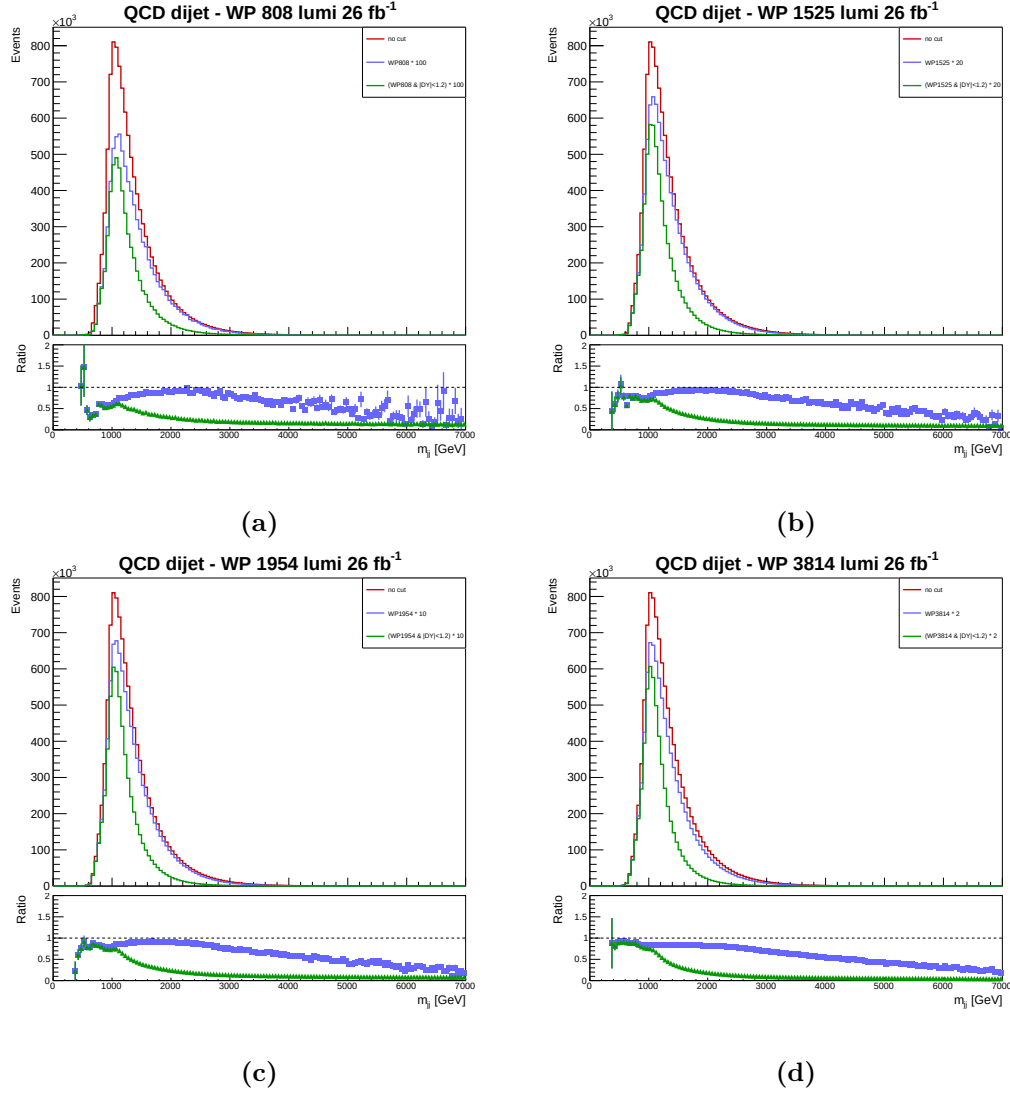


Figure 6.23. Invariant mass distributions for the QCD dijet without any cut (in red), with the cut on the event-level AS (in blue) and with the additional cut on $|\Delta Y|$ (in green). The plots are shown for different event-level AS thresholds, $\{808, 1525, 1954, 3814\}$ corresponding to rejection factors $\{100, 20, 10, 2\}$. To make them more visible, the histograms after the cuts are multiplied by the correspondent rejection factor. Below the distribution, the ratio plot between blue/green curves and red curve is shown.

StandardHypoTestInvDemo function [215] from ROOSTATS package and are determined at a 95% of confidence level. The expected limits are evaluated at different integrated luminosities: 26 fb^{-1} , approximately corresponding to the luminosity delivered during the 2023 data-taking period; 150 fb^{-1} representing the integrated luminosity collected so far in Run 3; 300 fb^{-1} which is the expected total Run 3 luminosity. The results are presented in terms of the expected signal production cross-section σ_{exp} , given by:

$$\sigma_{\text{exp}} = \mu \cdot \sigma_{\text{th}} \quad (6.7)$$

where σ_{th} is the theoretical cross-section. The theoretical cross-sections σ_{th} for the signals employed are shown in Tab. 6.10.

Signal	Theoretical cross-section σ_{th} [pb^{-1}]
$V \rightarrow WW, m_V = 2 \text{ TeV}$	$3.848 \cdot 10^{-3}$
$V \rightarrow WW, m_V = 3 \text{ TeV}$	$3.597 \cdot 10^{-4}$
$V \rightarrow WW, m_V = 4 \text{ TeV}$	$4.526 \cdot 10^{-5}$
$V \rightarrow WW, m_V = 5 \text{ TeV}$	$6.265 \cdot 10^{-6}$
$V \rightarrow XH, m_V = 2.3 \text{ TeV}$	$4.744 \cdot 10^{-1}$
$V \rightarrow XH, m_V = 3.4 \text{ TeV}$	$4.155 \cdot 10^{-2}$
$V \rightarrow XH, m_V = 5 \text{ TeV}$	$1.636 \cdot 10^{-3}$
$V \rightarrow XH, m_V = 6 \text{ TeV}$	$2.182 \cdot 10^{-4}$
DJ, $m_{Z'} = 3.5 \text{ TeV}$	$1.114 \cdot 10^{-9}$
$W' \rightarrow WZ, m_{W'} = 3 \text{ TeV}$	$6.968 \cdot 10^{-16}$

Table 6.10. Theoretical cross-section values used to calculate the expected cross-sections of the signals.

Recommendations for large R jet uncertainties in ATLAS Run 3 are not yet available. Therefore, a preliminary study has been conducted to assess the impact of the main sources of uncertainty on the signals: JES and JER (see Sec. 3.4.4). Based on Run 2 recommendations [133], we estimated a contribution of 5% for both JES and JER corrections, as illustrated in Fig. 6.24 for the DJ signal. In the resonance invariant mass distribution, the JES correction primarily shifts the resonance peak, while the JER correction broadens its resolution. The impact of the JES and JER corrections on the expected cross-sections is summarised in Tab. 6.11 for different signal hypotheses and luminosities. The nominal expected cross-section is computed without applying any selection on the AS or on $|\Delta Y|$. As expected, all cross-sections exhibit a dependence proportional to $1/\sqrt{\mathcal{L}}$. The performance deterioration due to the inclusion of JES-JER corrections is approximately 10-15%, increasing with the resonance mass. However, in most cases, the deterioration does not worsen with increasing luminosity, except for the VcXH signal with mass $m_V = 2.3 \text{ TeV}$, suggesting that this particular signal is not dominated by statistical uncertainties and that JES and JER corrections become significant. Since in most cases systematic uncertainties remain under control and statistical fluctuations dominate the total uncertainty, the following results are presented without incorporating the JES and JER corrections. These effects will be included once the ATLAS Run 3 recommen-

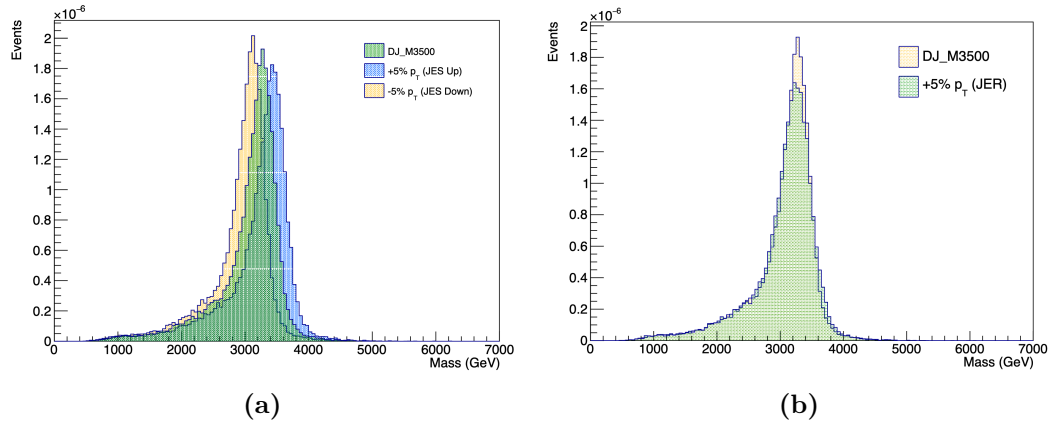


Figure 6.24. JES (on the left) and JER (on the right) corrections applied to the DJ signal p_T distributions at nominal luminosity 1 fb^{-1} . Here, the effects on the dijet invariant mass distributions are shown.

dations become available.

Signal	Luminosity 26 fb^{-1}		Luminosity 150 fb^{-1}		Luminosity 300 fb^{-1}	
	σ_{nom} [pb]	$\sigma_{JES+JER}$ [pb]	σ_{nom} [pb]	$\sigma_{JES+JER}$ [pb]	σ_{nom} [pb]	$\sigma_{JES+JER}$ [pb]
$V \rightarrow WW, m_V = 2 \text{ TeV}$	$9.41^{+3.69}_{-2.63} \cdot 10^{-2}$	$9.95^{+4.36}_{-2.99} \cdot 10^{-2}$	$3.91^{+1.53}_{-1.09} \cdot 10^{-2}$	$4.13^{+1.81}_{-1.24} \cdot 10^{-2}$	$2.77^{+1.08}_{-0.77} \cdot 10^{-2}$	$2.93^{+1.28}_{-0.88} \cdot 10^{-2}$
$V \rightarrow WW, m_V = 3 \text{ TeV}$	$3.60^{+1.41}_{-1.00} \cdot 10^{-2}$	$3.89^{+1.73}_{-1.19} \cdot 10^{-2}$	$1.50^{+0.59}_{-0.42} \cdot 10^{-2}$	$1.62^{+0.72}_{-0.50} \cdot 10^{-2}$	$1.06^{+0.41}_{-0.30} \cdot 10^{-2}$	$1.15^{+0.51}_{-0.35} \cdot 10^{-2}$
$V \rightarrow WW, m_V = 4 \text{ TeV}$	$1.48^{+0.58}_{-0.42} \cdot 10^{-2}$	$1.65^{+0.75}_{-0.52} \cdot 10^{-2}$	$6.16^{+2.42}_{-1.73} \cdot 10^{-3}$	$6.86^{+3.10}_{-2.15} \cdot 10^{-3}$	$4.35^{+1.71}_{-1.22} \cdot 10^{-3}$	$4.85^{+2.19}_{-1.52} \cdot 10^{-3}$
$V \rightarrow WW, m_V = 5 \text{ TeV}$	$6.55^{+2.60}_{-1.85} \cdot 10^{-3}$	$7.47^{+3.44}_{-2.38} \cdot 10^{-3}$	$2.72^{+1.07}_{-0.76} \cdot 10^{-3}$	$3.10^{+1.42}_{-0.98} \cdot 10^{-3}$	$1.92^{+0.75}_{-0.54} \cdot 10^{-3}$	$2.19^{+1.00}_{-0.70} \cdot 10^{-3}$
$V \rightarrow XH, m_V = 2.3 \text{ TeV}$	$9.32^{+3.65}_{-2.61} \cdot 10^{-2}$	$1.01^{+0.42}_{-0.31} \cdot 10^{-1}$	$3.88^{+1.52}_{-1.08} \cdot 10^{-2}$	$4.48^{+1.92}_{-0.97} \cdot 10^{-2}$	$2.74^{+1.07}_{-0.77} \cdot 10^{-2}$	$3.63^{+0.90}_{-0.49} \cdot 10^{-2}$
$V \rightarrow XH, m_V = 3.4 \text{ TeV}$	$3.19^{+1.25}_{-0.89} \cdot 10^{-2}$	$3.49^{+1.56}_{-1.07} \cdot 10^{-2}$	$1.32^{+0.52}_{-0.37} \cdot 10^{-2}$	$1.45^{+0.65}_{-0.44} \cdot 10^{-2}$	$9.36^{+3.67}_{-2.62} \cdot 10^{-3}$	$1.03^{+0.46}_{-0.30} \cdot 10^{-2}$
$V \rightarrow XH, m_V = 5 \text{ TeV}$	$6.35^{+2.50}_{-1.78} \cdot 10^{-2}$	$6.80^{+3.04}_{-2.06} \cdot 10^{-2}$	$2.64^{+1.04}_{-0.74} \cdot 10^{-2}$	$2.83^{+1.26}_{-0.86} \cdot 10^{-2}$	$1.87^{+0.73}_{-0.52} \cdot 10^{-2}$	$2.00^{+0.89}_{-0.60} \cdot 10^{-2}$
$V \rightarrow XH, m_V = 6 \text{ TeV}$	$3.35^{+1.36}_{-0.95} \cdot 10^{-3}$	$3.94^{+1.85}_{-1.28} \cdot 10^{-3}$	$1.38^{+0.55}_{-0.39} \cdot 10^{-3}$	$1.62^{+0.75}_{-0.52} \cdot 10^{-3}$	$9.70^{+3.84}_{-2.73} \cdot 10^{-4}$	$1.14^{+0.53}_{-0.70} \cdot 10^{-3}$
$DJ, m_{Z'} = 3.5 \text{ TeV}$	$3.54^{+1.39}_{-0.99} \cdot 10^{-2}$	$3.80^{+1.68}_{-1.16} \cdot 10^{-2}$	$1.47^{+0.58}_{-0.41} \cdot 10^{-2}$	$1.58^{+0.70}_{-0.48} \cdot 10^{-2}$	$1.04^{+0.41}_{-0.29} \cdot 10^{-2}$	$1.12^{+0.49}_{-0.34} \cdot 10^{-2}$
$W' \rightarrow WZ, m_{W'} = 3 \text{ TeV}$	$3.14^{+1.16}_{-0.88} \cdot 10^{-2}$	$3.37^{+1.52}_{-1.05} \cdot 10^{-2}$	$1.35^{+0.57}_{-0.23} \cdot 10^{-2}$	$1.40^{+0.63}_{-0.43} \cdot 10^{-2}$	$1.13^{+0.21}_{-0.44} \cdot 10^{-2}$	$9.93^{+4.47}_{-4.98} \cdot 10^{-3}$

Table 6.11. Expected upper limits on the signal cross-section with and without the JES and JER corrections for the different signals.

A study on the expected upper limits on signal cross-section is performed for various event-level AS WPs, with and without the additional $|\Delta Y|$ selection. Moreover, to compare the results with those obtained with a more standard discriminating variable cut, additional WPs have been considered. Since this analysis focuses on jet substructure, the D_2 variable (see Eq. 3.13) can be considered the best discriminative for 2-prong jets. Analogously to the transition from jet- to event-level AS, the D_2 of the single jets is multiplied to obtain new event-level quantity, the $D2_{ev}$. To perform a fair comparison with the AD analysis, the WPs for the $D2_{ev}$ are calculated so that the same rejection factor is achieved either cutting on the AS and $|\Delta Y|$ or cutting on $D2_{ev}$ and $|\Delta Y|$. The new WPs on $D2_{ev}$ are obtained cutting at 0.63, 0.97, 1.17 and 2.21. The results are summarised in Fig. 6.25-6.34.

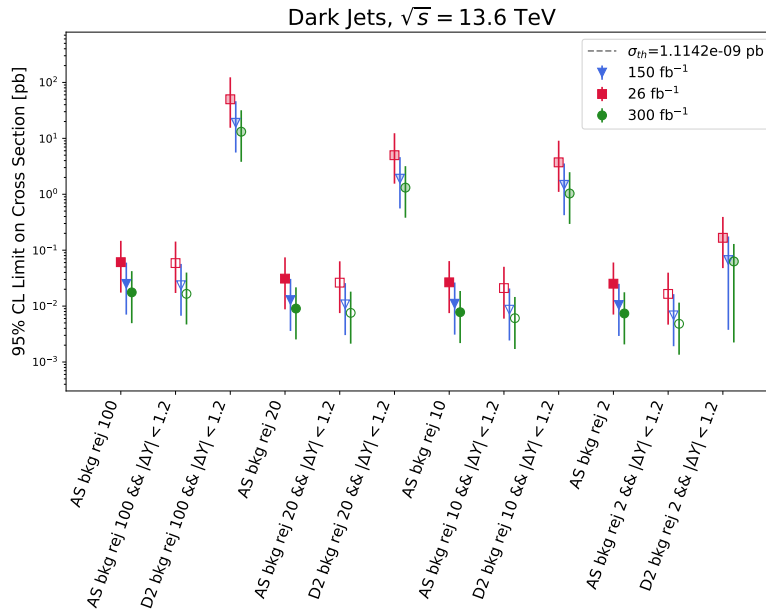


Figure 6.25. Expected upper limits on the DJ cross-section at different event-level AS WPs, with (open markers) and without (solid markers) the additional $|\Delta Y|$ selection. The WPs on $D2_{ev}$ are shown in transparent dots.

The $W' \rightarrow WZ$ signal (Fig. 6.26) and $V \rightarrow XH$ signals (Fig. 6.31-6.34) exhibit a strong dependence on signal efficiency. Consequently, the upper limits on the expected cross-section decrease as the background rejection factor is lowered. This is because a lower rejection factor allows more background events to pass the selection but simultaneously increases the number of selected signal events, improving sensitivity and leading to a tighter upper limit on the cross-section. However, this trend is not observed for the DJ (Fig. 6.25) and $V \rightarrow WW$ signals (Fig. 6.27-6.30), where reducing the background rejection results in only marginal improvements or, in some cases, a deterioration in detection performance. For $V \rightarrow WW$ signals, the WP corresponding to a background rejection factor of 20 appears to provide the optimal balance between efficiency and sensitivity. In all cases, the upper limits scales approximately as the inverse square root of the integrated luminosity. Additionally, the application of the $|\Delta Y|$ selection significantly enhances the performance.

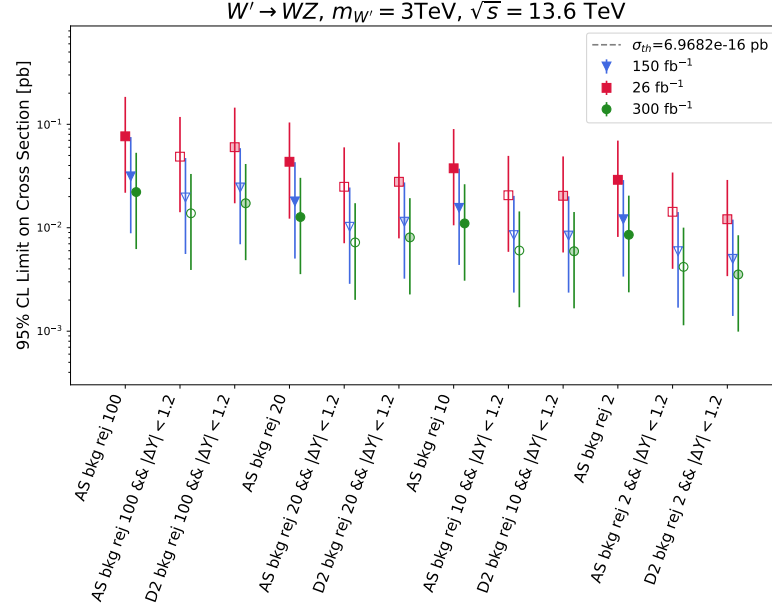


Figure 6.26. Expected upper limits on the $W' \rightarrow WZ$ cross-section at different event-level AS WPs, with (void dots) and without (solid markers) the additional $|\Delta Y|$ selection. The WPs on $D2_{ev}$ are shown in transparent dots.

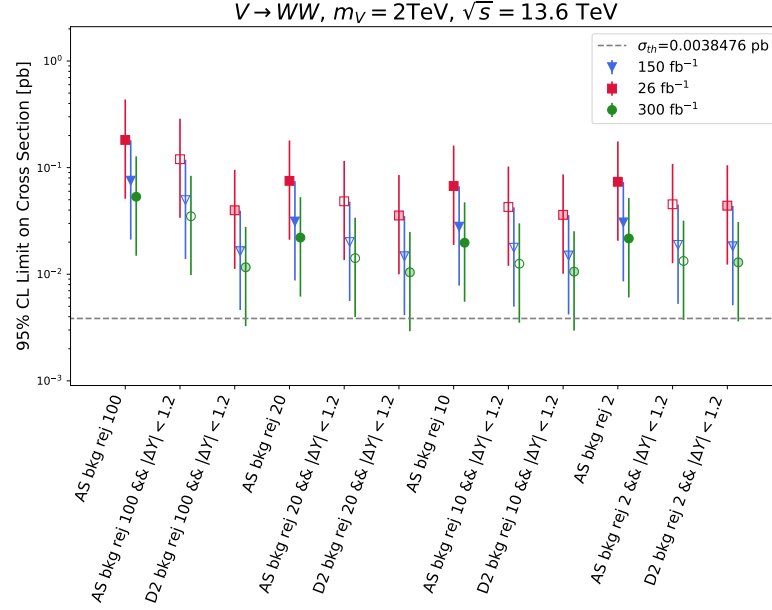


Figure 6.27. Expected upper limits on the $V \rightarrow WW$ ($m_V = 2 \text{ TeV}$) cross-section at different event-level AS WPs, with (open markers) and without (solid markers) the additional $|\Delta Y|$ selection. The WPs on $D2_{ev}$ are shown in transparent dots. The horizontal grey line represents the theoretical cross-section.

When compared to an analysis based on a more standard discriminating variable cut, for example performed on the $D2_{ev}$ variable, often the results obtained on $D2_{ev}$

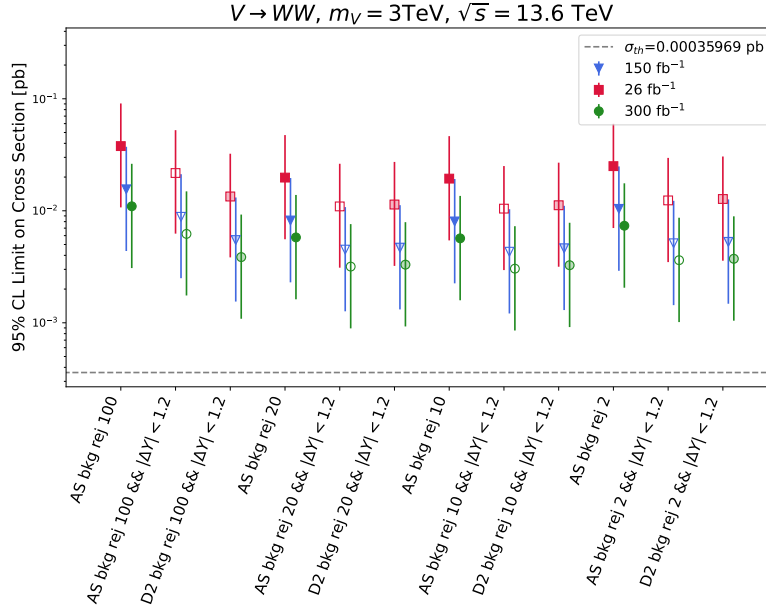


Figure 6.28. Expected upper limits on the $V \rightarrow WW$ ($m_V = 3 \text{ TeV}$) cross-section at different event-level AS WPs, with (open markers) and without (solid markers) the additional $|\Delta Y|$ selection. The WPs on $D2_{ev}$ are shown in transparent dots. The horizontal grey line represents the theoretical cross-section.

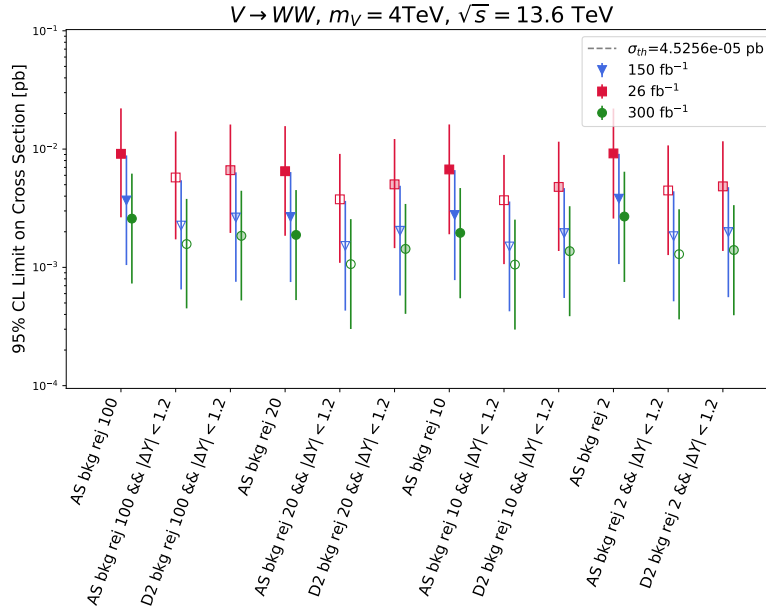


Figure 6.29. Expected upper limits on the $V \rightarrow WW$ ($m_V = 4 \text{ TeV}$) cross-section at different event-level AS WPs, with (open markers) and without (solid markers) the additional $|\Delta Y|$ selection. The WPs on $D2_{ev}$ are shown in transparent dots.

are similar or even better than those obtained cutting on the AS. This is expected, as the $D2$ variable discriminates 2-prong from 1-prong signatures. The majority

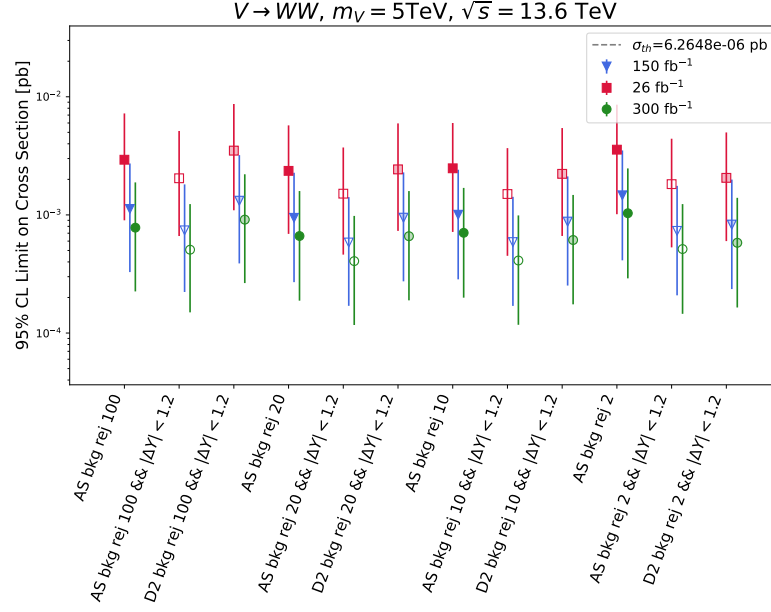


Figure 6.30. Expected upper limits on the $V \rightarrow WW$ ($m_V = 5$ TeV) cross-section at different event-level AS WPs, with (open markers) and without (solid markers) the additional $|\Delta Y|$ selection. The WPs on $D2_{ev}$ are shown in transparent dots.

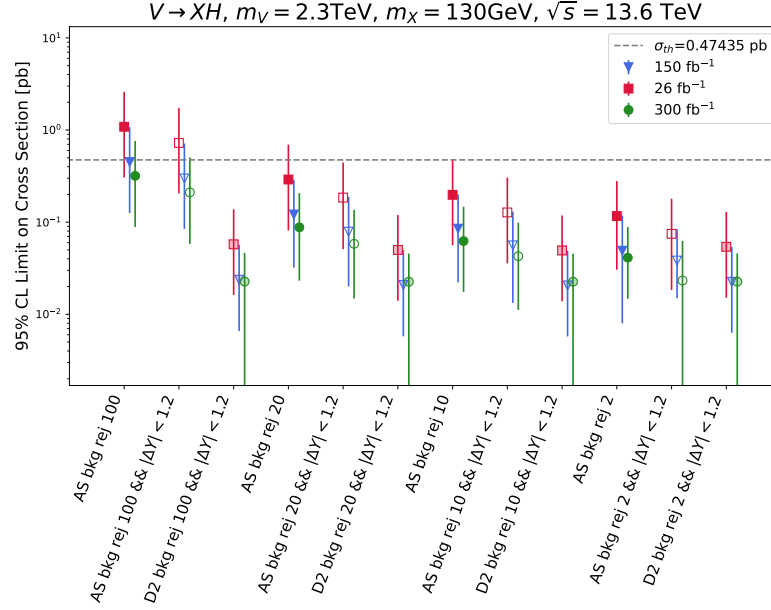


Figure 6.31. Expected upper limits on the $V \rightarrow XH$ ($m_V = 2.3$ TeV and $m_X = 130$ GeV) cross-section at different event-level AS WPs, with (open markers) and without (solid markers) the additional $|\Delta Y|$ selection. The WPs on $D2_{ev}$ are shown in transparent dots. The horizontal grey line represents the theoretical cross-section.

of the signals studied are 2-prong, with the exception of the $W' \rightarrow WZ$, which is 3-prong, and the DJ signal that has a more complex substructure. While for

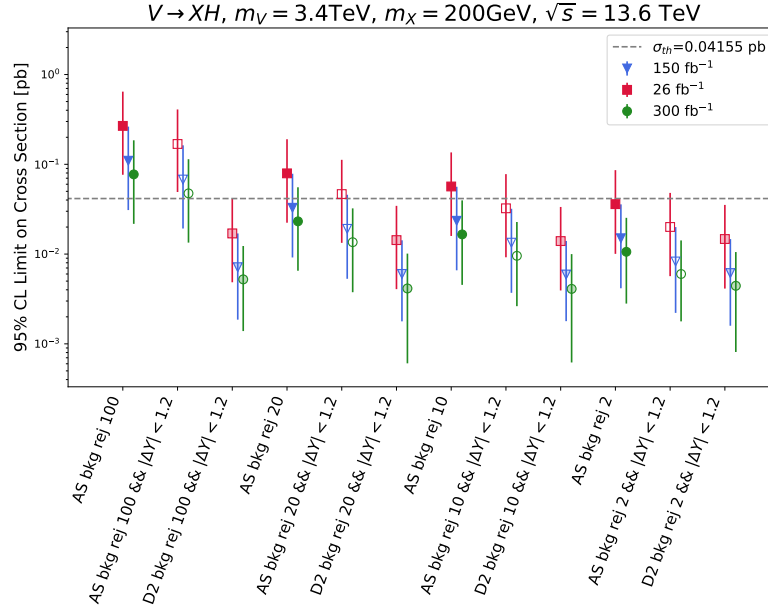


Figure 6.32. Expected upper limits on the $V \rightarrow XH$ ($m_V = 3.4$ TeV and $m_X = 200$ GeV) cross-section at different event-level AS WPs, with (open markers) and without (solid markers) the additional $|\Delta Y|$ selection. The WPs on $D2_{ev}$ are shown in transparent dots. The horizontal grey line represents the theoretical cross-section.

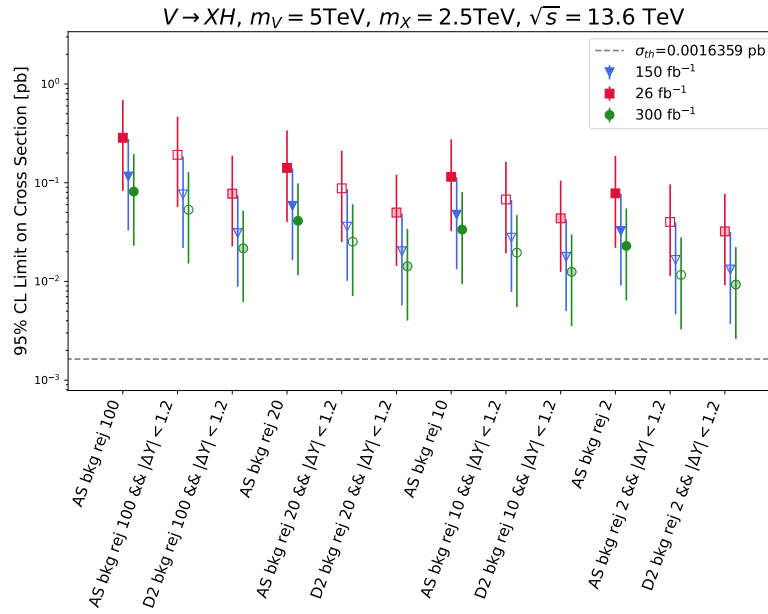


Figure 6.33. Expected upper limits on the $V \rightarrow XH$ ($m_V = 5$ TeV and $m_X = 2.5$ TeV) cross-section at different event-level AS WPs, with (open markers) and without (solid markers) the additional $|\Delta Y|$ selection. The WPs on $D2_{ev}$ are shown in transparent dots. The horizontal grey line represents the theoretical cross-section.

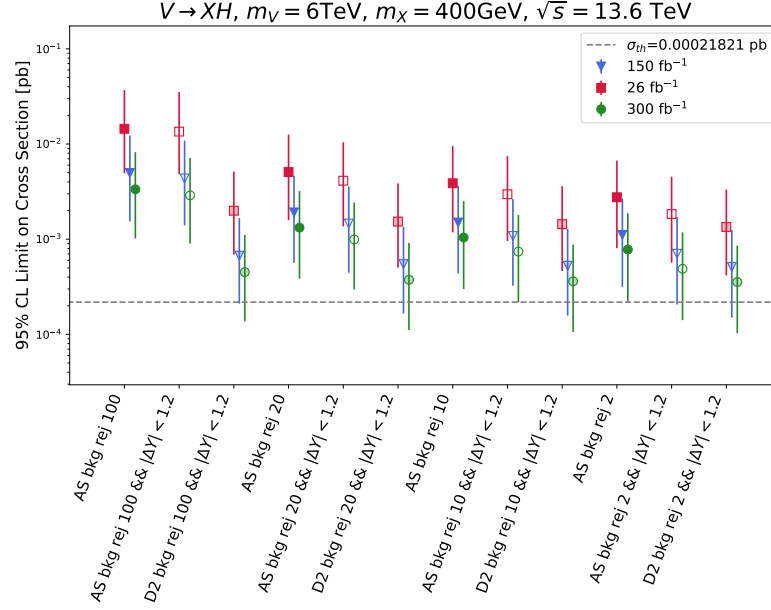


Figure 6.34. Expected upper limits on the $V \rightarrow XH$ ($m_V = 6$ TeV and $m_X = 400$ GeV) cross-section at different event-level AS WPs, with (open markers) and without (solid markers) the additional $|\Delta Y|$ selection. The WPs on $D2_{ev}$ are shown in transparent dots. The horizontal grey line represents the theoretical cross-section.

the $W' \rightarrow WZ$ the $D2_{ev}$ is still quite discriminative, resulting in similar results to the AS, for the DJ $D2_{ev}$ fails inevitably in identifying the BSM jets. In this case, the AD demonstrates its full power: it does not depend on the specific kinematic characteristics, like the prong-ness, of the signal.

Finally, the AD strategy described here demonstrates sufficient sensitivity to exclude low-mass $V \rightarrow XH$ signals and shows promising prospects for higher-mass $V \rightarrow XH$ and $V \rightarrow WW$ signals, bringing the observed limits closer to the theoretical cross-section predictions.

6.3.2 Towards a parametrisation of the observed QCD spectrum

The template fit method employed to estimate the upper limits on signal cross-section in Sec. 6.3.1 is not data-driven and systematic uncertainties are rigidly added through Gaussian constraints to the likelihood. For the transition to the analysis using Run 3 ATLAS data, it is crucial to develop an analytical approach directly based on data, where background uncertainties that are directly estimated from the parameters of the model itself. Therefore, an unbinned fit is performed on the dijet invariant mass distributions from the MC simulations.

The background shape is modelled using a QCD-inspired probability density function [216], described as:

$$f(m_{jj}) = a(1 - x^d)^c x^{b_0 + b_1 \log x + b_2 \log^2 x + \dots} \quad (6.8)$$

where x represents the mass normalised to the energy of the centre of mass, namely $x = m_{jj}/\sqrt{s}$. For this study, only the configuration with the WP at the event-level AS with background rejection factor of 100, the $|\Delta Y|$ cut and the full Run 3 luminosity of 300 fb^{-1} is considered. In this way, The fit is shown in Fig. 6.35 and is computed using the ROOFIT package [217] with 3 parameters: $b_0 = (-5.7993 \pm 0.0001) \cdot 10^{-1}$, $c = (8.2048 \pm 0.0042) \cdot 10^{-1}$ and $d = (-5.7291 \pm 0.0002) \cdot 10^{-1}$.

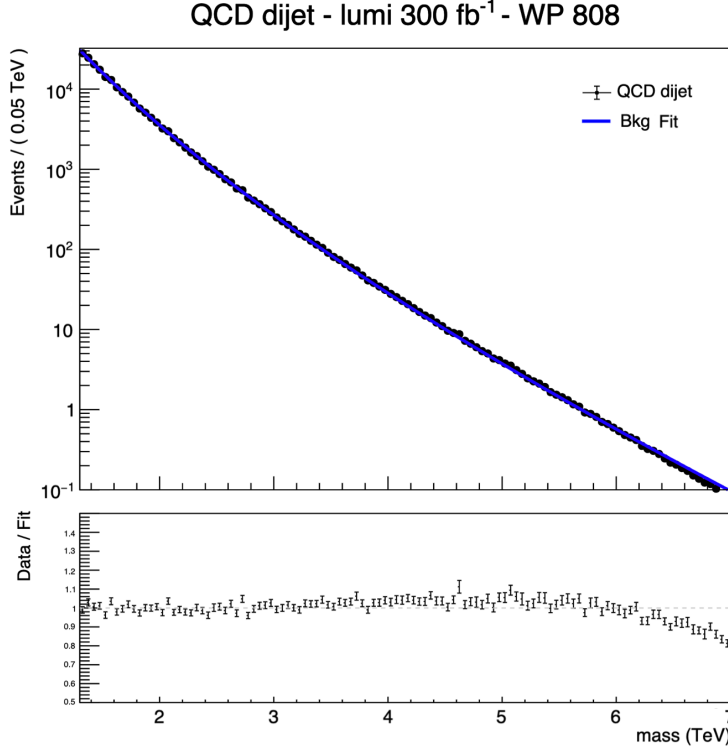


Figure 6.35. Fit of the QCD dijet MC background with the QCD-inspired probability density function described in Eq. 6.8. The events are selected with WP at rejection factor 100 and luminosity 300 fb^{-1} . The agreement of the two distributions is suboptimal because of the presence of steps in proximity to QCD slices boundaries which make the fit convergence difficult.

The normalisation parameter a is estimated subsequently, as the ROOFIT framework automatically normalises the probability density functions to unity. The fit is performed from 1.3 to 7 TeV, guaranteeing to lie on the trigger efficiency plateau. The agreement between the fitted curve and the invariant mass distribution is not optimal, as demonstrated by the ratio plot, which is slightly above the unity in the 3-6 TeV and slightly below in the high-mass tail. The fit's convergence was affected by the presence of residual steps corresponding to the QCD slices boundaries, visible as small bumps in the plot. Moreover, the dijet QCD MC simulation is generated as a proxy for data, so we expect that these discrepancies will disappear when using data. However, for the purposes of this study, the convergence of the fit can be considered satisfactory.

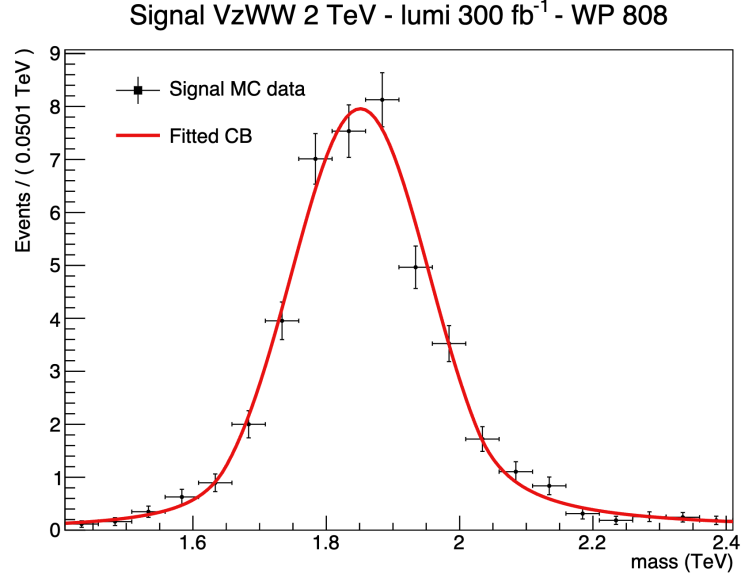


Figure 6.36. Fit of the $V \rightarrow WW$ signal at $m_V = 2$ TeV signal with a double-sided CB. The events are selected with WP at rejection factor 100 and luminosity 300 fb^{-1} .

The signal samples are fitted with a double-sided Crystal Ball (CB), expressed as:

$$f(m; m_0, \sigma, \alpha_L, n_L, \alpha_R, n_R) = \begin{cases} A_L \cdot \left(B_L - \frac{m-m_0}{\sigma_L} \right)^{-n_L} & \text{for } \frac{m-m_0}{\sigma_L} \leq -\alpha_L \\ \exp \left(-\frac{1}{2} \left(\frac{m-m_0}{\sigma_L} \right)^2 \right) & \text{for } 0 \leq \frac{m-m_0}{\sigma_L} \\ \exp \left(-\frac{1}{2} \left(\frac{m-m_0}{\sigma_R} \right)^2 \right) & \text{for } \frac{m-m_0}{\sigma_L} \leq \alpha_R \\ A_R \cdot \left(B_R + \frac{m-m_0}{\sigma_R} \right)^{-n_R} & \text{otherwise} \end{cases} \quad (6.9)$$

$$\text{where } A_i = \left(\frac{n_i}{|\alpha_i|} \right)^{n_i} \cdot \exp \left(-\frac{|\alpha_i|^2}{2} \right), \quad B_i = \frac{n_i}{|\alpha_i|} - |\alpha_i| \quad (6.10)$$

The CB probability density function is characterised by a Gaussian distribution centred in m_0 with right and left standard deviations $\sigma_{R,L}$. The right and left tails, instead, are exponentially falling and are regulated by the $n_{R,L}$ rate parameters. The $\alpha_{R,L}$ parameters indicate when the Gaussian distribution is substituted by the exponential distributions in the tails. The CB is used to model all the signal samples and it demonstrates to sufficiently describe the signal shapes for all the utilised samples, as shown in Fig. 6.36.

A new background sample is generated from the fitted distribution, to correctly account for the expected statistics in data. The agreement between the curve from the fit and the newly generated toy dataset is shown in Fig. 6.37a. Then, the toy dataset is fitted with a curve given by the sum of the signal and background distributions, as illustrated in Fig. 6.37b. The Gaussian systematic uncertainties on luminosity (expected 2%, as in [214]) and on number of signal events are added to the signal+background fit. The number of background events is left as an additional degree of freedom for the fit, while the parameter of interest is the signal strength

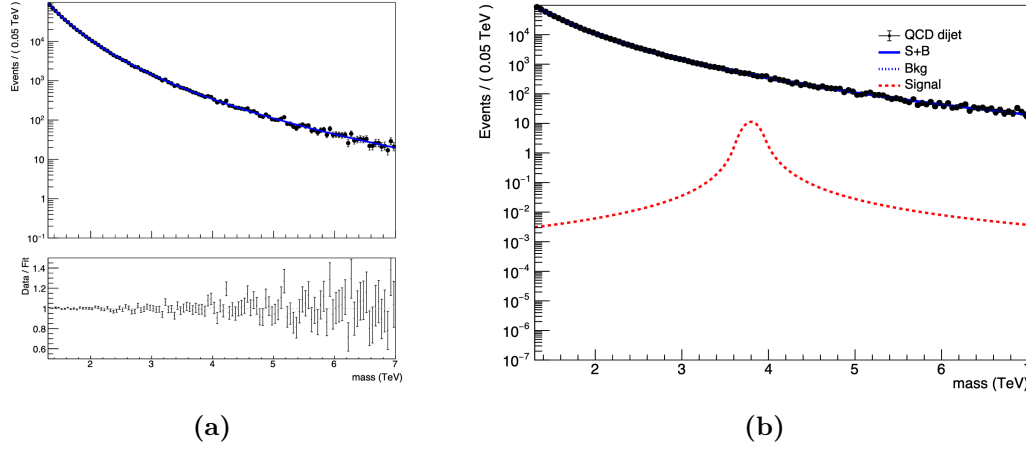


Figure 6.37. On the left, invariant mass distribution of the toy background dataset (black dots) generated from the QCD-inspired probability density function in Eq. 6.8 (in blue). On the right, signal plus background fit for the signal $V \rightarrow WW$ with $m_V = 4\text{ TeV}$. Both plots refer to the WP at rejection factor 100 and luminosity 300 fb^{-1} .

μ . After performing the fit, the CLs method at 95% confidence level is used to infer the upper limits on signal cross-sections for all the signals and the results are shown in Tab. 6.12. The upper limits are compatible with those found with the binned profile likelihood ratio, within their uncertainties¹. For the same signal topologies, the uncertainties on the upper limits decrease when the resonance mass increases: at high invariant mass, the background estimation appears to be more constrained and the uncertainties decrease. Signal injection tests will be performed to verify that this approach is more sensitive to signals at high mass.

¹The uncertainties here considered do not include the JES and JER corrections, that we estimated to have approximately 20% of impact, or a fit bias associated with the background model. In particular the fit bias, or spurious signal, is generally computed from the difference between the median fitted signal yield and the expected signal yield and provides a measurement of the flexibility of the background model to reabsorb a possible signal, partially or completely.

Signal	Expected upper limits on cross-section $\sigma_{\text{exp BH}}$ [pb]
$V \rightarrow WW, m_V = 2 \text{ TeV}$	$3.64^{+3.16}_{-2.27} \cdot 10^{-2}$
$V \rightarrow WW, m_V = 3 \text{ TeV}$	$7.85^{+3.30}_{-2.85} \cdot 10^{-3}$
$V \rightarrow WW, m_V = 4 \text{ TeV}$	$2.76^{+1.06}_{-0.75} \cdot 10^{-3}$
$V \rightarrow WW, m_V = 5 \text{ TeV}$	$1.75^{+0.53}_{-0.40} \cdot 10^{-3}$
$V \rightarrow XH, m_V = 2.3 \text{ TeV}$	$1.25^{+0.51}_{-0.20} \cdot 10^{-1}$
$V \rightarrow XH, m_V = 3.4 \text{ TeV}$	$4.51^{+4.03}_{-1.12} \cdot 10^{-2}$
$V \rightarrow XH, m_V = 5 \text{ TeV}$	$3.86^{+0.85}_{-1.71} \cdot 10^{-2}$
$V \rightarrow XH, m_V = 6 \text{ TeV}$	$1.03^{+0.40}_{-0.10} \cdot 10^{-2}$
$\text{DJ}, m_{Z'} = 3.5 \text{ TeV}$	$1.91^{+0.74}_{-0.15} \cdot 10^{-2}$
$W' \rightarrow WZ, m_{W'} = 3 \text{ TeV}$	$2.46^{+1.41}_{-0.65} \cdot 10^{-2}$

Table 6.12. Expected upper limits on signal cross-section calculated with the unbinned fit method.

6.4 Perspectives on the ATLAS analysis

The studies conducted thus far relied exclusively on MC simulations, which, while valuable for assessing the validity of the approach, have inherent limitations. In particular, QCD dijet events, which constitute the dominant background, are not accurately modelled in MC samples, as QCD processes are difficult to be simulated. Moreover, the final background estimation will be entirely data-driven, as the AD model is designed to be trained directly on real data rather than relying on simulated background distributions. Before transitioning to ATLAS Run 3 data, several pending steps must be completed. Specifically, final validation tests must be performed on the graph dataset and GNN models, which have yet to be fully implemented. Preliminary studies have demonstrated that training on small MC background datasets sampled according to the QCD dijet transverse momentum distribution is feasible. However, scaling up to a larger number of events remains necessary to ensure statistical robustness. Additionally, the approach adopted thus far has treated individual jets independently, subsequently combining their jet-level AS scores to derive an event-level anomaly score. A more structured approach, employing graph-structured data, could provide a more comprehensive representation of the entire event. By incorporating correlations among all jets, the model may exploit richer kinematic and topological features, ultimately enhancing its discrim-

inative power.

Once the optimal ML model has been identified, the transition to real ATLAS data can proceed. The SR and CR definitions based on $|\Delta Y|$ selection have proven to be both robust and effective. Therefore, we expect to apply the $|\Delta Y| = 1.2$ selection cut in the data analysis as well. The impact of detector-related systematic uncertainties, such as JES and JER, have been preliminary estimated, but not yet incorporated. However, the ATLAS collaboration is expected to release Run 3 recommendations for large R UFO jets in the near future, which will subsequently be included in the analysis.

Given the novelty of the approach and the potential discrepancies between results obtained from MC training and those derived from real ATLAS data, it is imperative to validate the AD model using well-established *SM candles*. AD model's discriminative power should be driven solely by the background sample on which it is trained. If trained primarily on QCD dijets, the architecture should be capable of re-observing or even effectively re-discovering known SM particles. For the 2-prong topology, W +jet events serve as a natural validation benchmark, whereas $t\bar{t}$ events provide an ideal test case for the 3-prong topology. These benchmarks will serve as a critical validation step, ensuring the reliability of the overall approach before searching for NP.

Ultimately, the primary objective of this analysis is the search for BSM resonances. This will be achieved through a Bump Hunt methodology, in which the background is modelled directly from data using the CR derived by inverting the $|\Delta Y|$ cut. The method so far tested has proven to be sensitive to a broad range of NP signatures without requiring anything but the constituent information of the jets.

Chapter 7

New algorithms for the L0 barrel Muon Spectrometer trigger

At the end of Run 3, in July 2026, LHC will enter a three-year shutdown period, known as Long Shutdown 3 (LS3), during which the LHC accelerators will undergo significant upgrades to increase the instantaneous luminosity to $5\text{--}7.5 \times 10^{34} \text{ cm}^{-2} \text{ s}^{-1}$. Simultaneously, several ATLAS sub-detectors will be upgraded or replaced to mitigate inefficiencies caused by ageing and to adapt to the increased luminosity. In particular, new muon chambers will be installed in the barrel region of the Muon Spectrometer, consisting of RPC triplets and smaller MDT detectors. The readout and trigger logic will be implemented using Field Programmable Gate Arrays, while the TDAQ system will be redesigned to manage the higher luminosity and associated PU.

For the RPC trigger logic, three approaches have been developed: two based on traditional algorithms and one using ML techniques. This chapter provides a detailed discussion of the challenges posed by the demanding experimental conditions, along with an in-depth description of the newly developed ML-based trigger algorithms.

7.1 ATLAS Phase II Upgrade

The LHC timeline was originally defined at the time of its project approval and was expected to have a lifespan from 2008 to approximately 2040. The schedule has been revised multiple times due to unexpected incidents and operational challenges; nevertheless, the most recent schedule is shown in Fig. 7.1.

7.1.1 HL-LHC plan

Run 3 will mark the end of the LHC in its original design configuration. Following this, a new phase, known as High-Luminosity LHC (HL-LHC) [219], will begin, with the primary objective of significantly increasing the collider’s luminosity. The goal

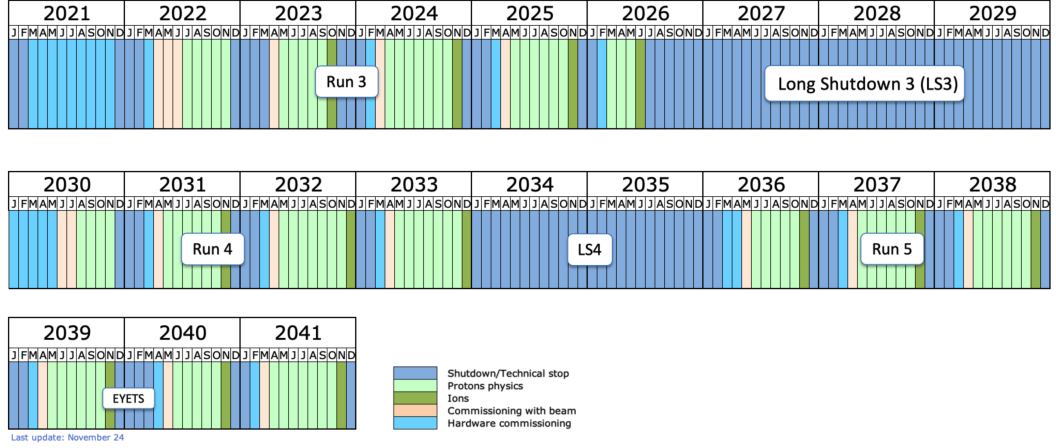


Figure 7.1. Long-term LHC schedule, updated in November 2024. Run 3 is expected to conclude in June 2026, allowing LS3 to begin in July 2026 and end with hardware commissioning three years later, in May 2030. Figure from Ref. [218].

is to achieve a peak luminosity of $5 \times 10^{34} \text{ cm}^{-2} \text{ s}^{-1}$ in Run 4, with the possibility of reaching $7.5 \times 10^{34} \text{ cm}^{-2} \text{ s}^{-1}$ during Run 5, as shown in Fig. 7.2. In this way, HL-

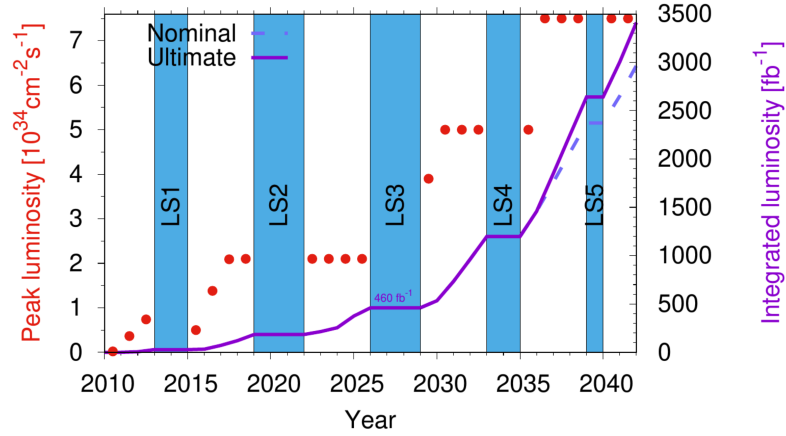


Figure 7.2. Expected instantaneous and integrated luminosity for proton-proton collisions during HL-LHC. in Figure from Ref. [220].

LHC is expected to collect an integrated luminosity of 250 fb^{-1} per year, ultimately reaching 3000 fb^{-1} by the end of Run 5. To accommodate the increased collision rates and higher detector PU, the HL-LHC will undergo substantial modifications. The installation of new superconducting magnets will improve beam separation, focusing and recombination, while the replacement of the insertion lines in correspondence with the ATLAS and CMS detectors will allow for enhanced operational efficiency. A key upgrade in this phase will be the introduction of superconducting RF *crab cavities* in the insertion lines, designed to enhance beam overlap and restore luminosity efficiency at the collision points. All upgrades and replacements will be implemented within a three-year period during LS3, between Run 3 and Run 4, as shown in Fig. 7.1. During this time, the ATLAS and CMS experiments will

carry out major detector upgrades to prepare for the so-called Phase II data taking, ensuring that the experiments can fully exploit the increased luminosity and extend the physics reach of the HL-LHC [221, 222].

7.1.2 ATLAS sub-detectors upgrade

The direct consequence of the luminosity increase during Phase II is a higher PU configuration, corresponding to an average number of collisions of $\langle\mu\rangle \simeq 140 - 200$, depending on the instantaneous luminosity. The majority of precision measurements and NP searches are highly dependent on the trigger and reconstruction performance, both of which are sensitive to $\langle\mu\rangle$. To maintain excellent detector efficiencies, the ID consisting of the pixel, strip and TRT detectors, will be completely replaced by the Inner TracKer (ITk) [223, 224]. The ITk will be entirely based on semiconductor technology and will extend the coverage up to $|\eta| < 4$. The calorimeters, however, will remain unchanged, except for their electronics, which will be replaced as they have reached their radiation tolerance limits and are not capable of operating at the latencies required for Phase II [225, 226]. In the MS, the NSW and the Big Wheel will not be affected by the upgrade. However, in the barrel region a new chamber consisting of an RPC triplet and a smaller MDT detector will replace the innermost layer, which is currently occupied by the BI MDT chambers [227]. The TDAQ system of ATLAS will be completely redesigned, while still maintaining the two-level trigger structure, consisting of low-level and high-level triggers. The new low-level trigger, known as Level 0 (L0), will perform the readout logic and preliminary trigger selections and is expected to operate at 1 MHz (compared to 100 kHz during Phase I), while the latency will increase to 10 μ s (from 3 μ s in Run 3) [228]. The high-level trigger, known as the Event Filter, will refine the trigger information, further reduce the rate to 10 kHz and make the final decision on event acceptance for permanent storage.

7.1.3 The BI chambers

During LS3, a new chamber will be inserted in the innermost layer of the barrel of the MS, the Barrel Inner (BI) station. In the place of the MDT chambers, the new BI station will consist of multilayer small MDT (sMDT) chambers integrated with an RPC triplet. Technological advancements have made it possible to reduce the thickness of both the RPC gas-gaps, from 2 to 1 mm, and the MDT resistive electrodes, from 1.8 to 1 mm. As a result, the new chambers will fit precisely into the space currently occupied by the innermost layer of MDTs. Prototypes of these chambers, the BIS78 stations [229], were produced and successfully installed in 2020 in the transition region between barrel and end-cap at $1.0 < |\eta| < 1.3$ in small sectors¹, as shown in Fig. 7.4. The new chambers will extend the RPC coverage, reaching 91% of the geometrical acceptance which, when combined with the MDT coverage, will result in 96% trigger acceptance. Moreover, due to the slimmer layout, the RPCs will operate at lower voltage (reduced from 9600 to

¹The large sectors are already covered by the EIL4 TGC chambers, installed as part of the NSW upgrade.

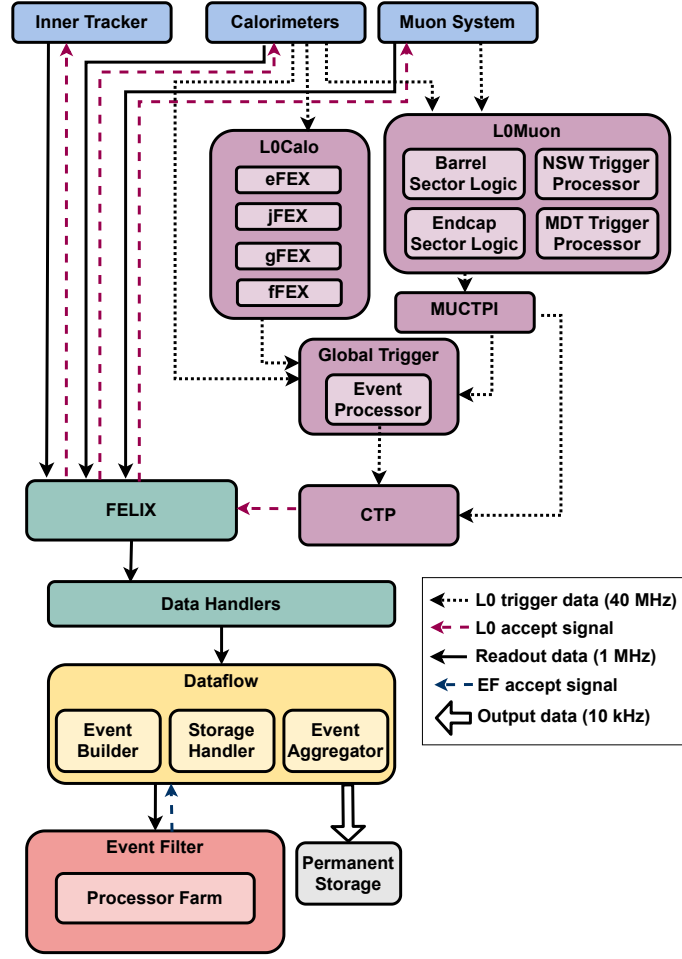


Figure 7.3. Schematics of the TDAQ system during Phase II of ATLAS. Figure from Ref. [228].

5400 V), which will not only enhance avalanche transfer efficiency and the signal-to-noise ratio but also improve the radiation resistance of the RPCs. Additionally, the insertion of three extra layers closer to the IP will provide a greater lever arm for muon track reconstruction and momentum determination, as described in Sec. 2.2.2. Each layer of the triplets will consist of orthogonal strips for the η measurement, with readout at both edges of the layer. The ϕ coordinate will be inferred from the timing difference between the two readout channels. The chambers will achieve improved timing and spatial resolutions of 0.4-0.5 ns and ≈ 1 mm, respectively. The cluster size of the new chambers during muon passage will remain between 1 and 2 strips, similarly to the older RPCs.



Figure 7.4. Longitudinal view of the MS in a large (Fig. 7.4a) and small sector (Fig. 7.4b). The transition barrel-end-cap region, where $1.0 < |\eta| < 1.3$, will be covered by the BIS78 chambers in the small sectors and by the EIL4 TGC in the large sectors. Figure from Ref. [227].

7.2 Level 0 RPC TDAQ system

The L0 trigger for the barrel of the MS is responsible for the identification of muon candidates using the signals from the RPC, MDT and Tile Calorimeter detectors. The schematics of the RPC TDAQ system is shown in Fig. 7.5. This strategy

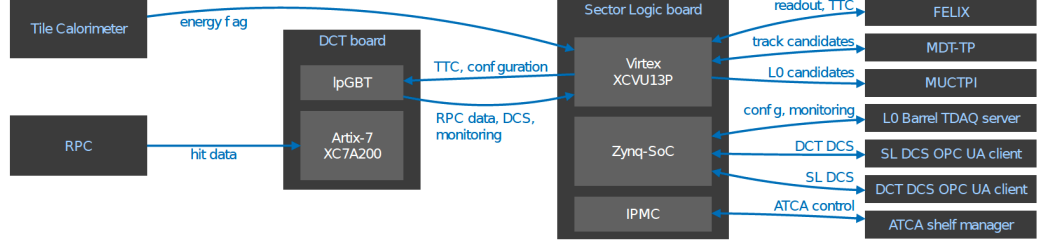


Figure 7.5. Schematic of the RPC TDAQ system. Signals from the RPC are transmitted to the DCT boards and SL. The data is then passed to the MDT Trigger Processor; if the MDT confirms the candidate, the signal is sent to the MUCTPI. Figure from Ref. [230].

combines the efficiency of custom integrated circuits with the flexibility of commercial Field Programmable Gate Arrays (FPGA) [231]. FPGAs are reconfigurable semiconductor devices, consisting of a matrix of interconnected logic blocks; the firmware is programmed in Hardware Description Language (HDL).

7.2.1 Data Collector and Transmitter boards

The RPC are read out by 1546 Data Collector and Transmitter (DCT) boards, 338 of which are dedicated to the BI region, while the remaining 1208 serve the BM and BO regions. Each DCT board consists of an FPGA, which performs the digitisation of the RPC signals, and a radiation-tolerant ASIC, responsible for communication with the other TDAQ components. In the BI region, the DCT also includes a Time-to-Digital Converter (TDC) which reads out the η and ϕ coordinates from the RPC triplets. In BI, the η and ϕ information is encoded in two 28-bit words. In contrast, the BM and BO outputs consists of 28-bit words containing either the η or ϕ information. The DCT boards also perform logic coincidences of hits arriving from the different layers of the same RPC plane, as described in detail in Sec. 7.3. The DCTs are located on the detector and, once they have assembled the 28-bit words, they transfer the signal to another board, the Sector Logic (SL), whose prototype is shown in Fig. 7.6.

7.2.2 Sector Logic boards

32 SL boards, one for each sector and each side, are located in the ATLAS counting room, USA15. The SL includes a Xilinx Virtex Ultrascale+ XCVU13P FPGA, which implements the trigger firmware, and a Xilinx Zynq Ultrascale+ SoC, responsible for communication with the detector control system. The barrel SL is divided

into four interconnected Super Logic Regions (SLR0, SLR1, SLR2 and SLR3) with each utilising one-fourth of the FPGA resources. This division has been optimised to reduce FPGA resource utilisation and minimise the number of signal transfers. In fact, the η strips in the BM and BO regions cover only half of the sector and can, therefore, be collected independently by SLR1 and SLR2. SLR0 receives signals only from the BI region which, unlike the other regions, has η strips that span the entire length of the chamber. All the readout logic is implemented in SLR3. The trigger algorithm, located in SLR1 and SLR2, selects muon candidates and sends them to the MDT Trigger Processor (MDT-TP). The MDT-TP receives five trigger candidates: three from the geometrically associated RPC SL and two from the neighbouring RPC SLs. Starting from the RPC candidates and utilising the MDT's precise muon transverse momentum estimation, the MDT-TP confirms or rejects the candidates. The candidates that pass the two-step selection are then sent from the SL to the Muon Central Trigger Processor Interface (MUCTPI), which resolves overlapping candidates. In the transition region, an energy flag from the Tile Calorimeter is sent to the SLR0 to assist in the selection of muon candidates. The BIS78 data collected in SLR0 will also be forwarded to the end-cap SL to ensure full coverage of all possible intersection areas.

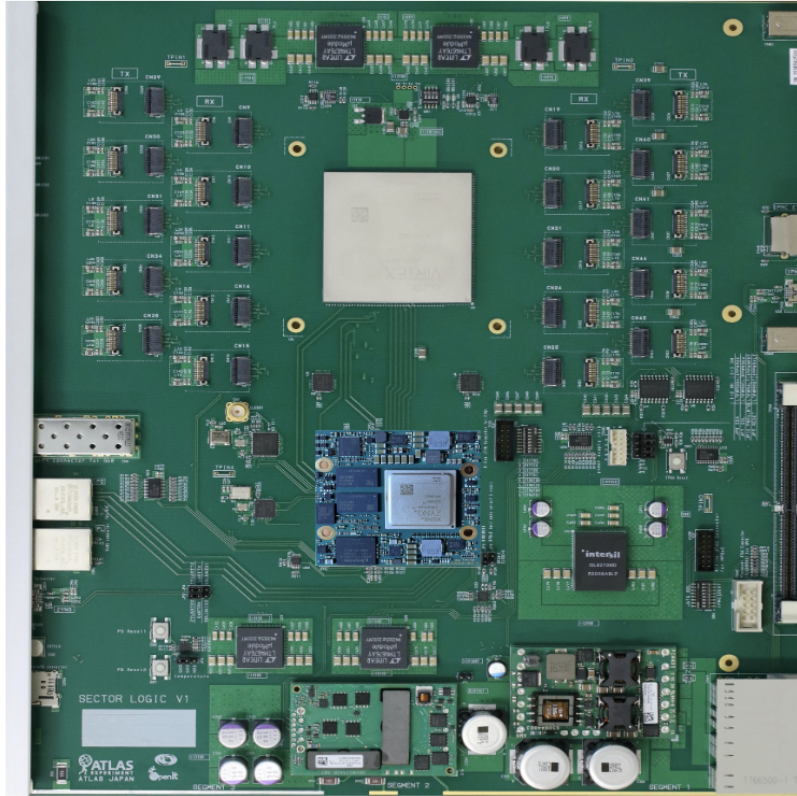


Figure 7.6. Prototype of the SL board for the L0 RPC trigger.

7.3 Level 0 RPC trigger algorithms

The L0 RPC trigger algorithm identifies muon candidates within a semi-sector, if present, by exploiting the fast RPC response. Candidates are then passed to sophisticated MDT trigger algorithms, which refine the selection using the more precise information from MDTs. The 28-bit words corresponding to hits in the detector are sent from the DCTs to the SL board, where they are grouped into arrays belonging to the same event, hence sharing the same BCID. These hits from the RPCs serve as the input for the trigger algorithm. However, they have been preprocessed to reduce resource occupancy and suppress noise. This preprocessing involves searching for coincidences within the doublets or triplets of the same plane, as follows:

- in the BI layers a 2/3 coincidence is required;
- a 1/2 coincidence is applied in the BM1, BM2 and BO layers.

The final format consists of four arrays, one for each super-layer, with length equal to the number of strips. The array elements contain 0-1 values, indicating the absence or presence of a hit.

Three different approaches currently under development for the L0 trigger algorithm: two, PatFinder and Pattern-matching, are traditional algorithms that search for coincidences within the RPC semi-sectors; the third is a ML-based algorithm. The three methods are still in development and the final choice of the L0 RPC trigger algorithm has not yet been made.

7.3.1 The PatFinder algorithm

The Pattern Finder algorithm, or PatFinder, is a revised version of the current L1 trigger algorithm, designed to incorporate the RPC BI triplets. PatFinder is a geometrical method that searches for track patterns within the RPC layers. It is based on the principle that muons originating from the IP with infinite transverse momentum follow straight-line trajectories with same η and ϕ coordinates across the RPC layers. The lower the muon's transverse momentum, the more its trajectory bends in the η plane, deviating from the straight-line path. By calculating the η and ϕ differences, PatFinder estimates the transverse momentum and position of the muon. It first computes the following quantities:

$$\eta_{PF} = \frac{z}{R}; \quad \phi_{PF} = \frac{y_L}{R} \quad (7.1)$$

where z and R are the global coordinates described in Sec. 2.2.1, while y_L is the local coordinate of the RPC, as shown in Fig. 7.7. The new coordinates are related to the global η and ϕ , by the expressions:

$$\eta \simeq \eta_{PF}, \quad \phi \simeq \phi_{\text{chamber}} + \phi_{PF} \quad (7.2)$$

where ϕ_{chamber} represents the global azimuthal coordinate of the RPC chamber in consideration. From each η bin in the BI super-layer, PatFinder opens a coincidence window in the subsequent super-layers and searches for hits, as illustrated

in Fig. 7.8a. This search is performed simultaneously in both the η and ϕ coordinates and the coincidence windows are optimised for each p_T threshold. If multiple

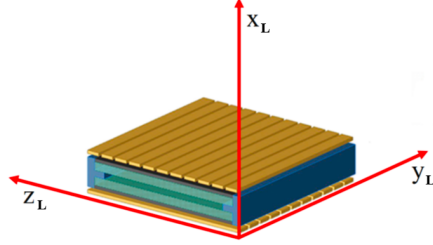


Figure 7.7. Representation of the local x_L , y_L , z_L coordinates with respect to the RPC layers. Figure from Ref. [232].

hits are found within the same coincidence window and in the same super-layer, PatFinder considers only the closest to the straight line connecting the IP to the hit in the BI super-layer. The coincidences are then combined to reconstruct the muon candidates according to the logic outlined in Tab. 7.1.

The PatFinder approach requires an initial seed to trigger the algorithm, which is typically identified in the BI layer, referred to as the *pivot*, and subsequently confirmed in the other layers. This choice arises from the new BI layers being fully efficient if compared to the older BM-BO chambers, whose declining performance will impact the overall trigger efficiency. However, the determination of the pivot super-layer is not unique.

Logic	BI	BM1	BM2	BO
3/3 chambers		✓	✓	✓
3/4 chambers	✓ ✓	✓	✓	✓ ✓
3/4 chambers + BI-BO	✓ ✓	✓	✓	✓ ✓

Table 7.1. Coincidence logic of the PatFinder algorithm. The 3/3 logic corresponds to the one used during Phase I.

7.3.2 The Pattern-matching algorithm

The Pattern-matching algorithm consists of a large look-up table containing all possible muon patterns. The hits in the RPCs are then compared to the expected patterns and, if a match is found, the hits are extracted and associated with the corresponding transverse momentum and coordinates, as shown in Fig. 7.8b. However, the correspondence between patterns and muon features is not bijective. A muon with the same p_T and spatial coordinates may leave different signatures inside the detector, corresponding to different patterns. The Pattern-matching method relies

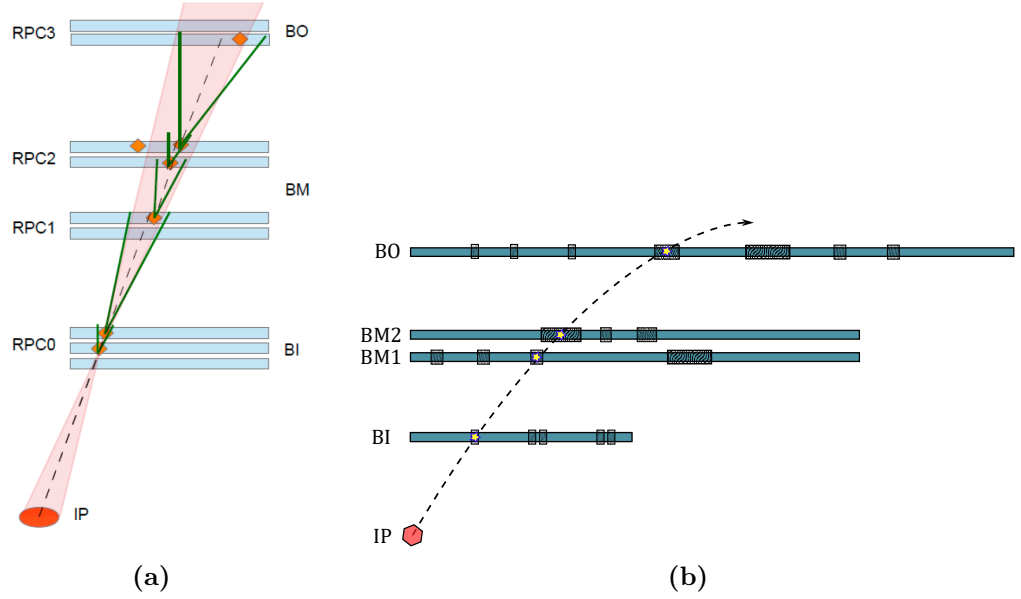


Figure 7.8. In Fig. 7.8a, sketch of the coincidence windows used in the PatFinder algorithm. Figure from Ref. [228]. Representation of the Pattern-matching algorithm in Fig. 7.8b. Among the different patterns matching the muon hits, a priority encoder assigns a quality score to each pattern and selects the best one accordingly.

heavily on the muon track simulations, which will be described in Sec. 7.5.1. From the dataset of all simulated muon tracks, the algorithm extracts a table of patterns and stores them in a database. Therefore, any simulation inefficiencies directly affect the algorithm's performance. The geometry of the MS detector, as modelled in the dataset simulation, is inherently transferred to the algorithm, leading to optimal performance. The size of the look-up table is strictly dependent on the number of possible muon track configurations. Using four super-layers instead of the separate nine RPC layers helps keep the size manageable. In fact, with four super-layers the number of possible patterns is approximately 100 k, which remains feasible for implementation on the specific FPGA. Finally, a priority encoder is implemented to resolve ambiguities when multiple patterns match a single sequence of hits.

7.3.3 Advantages and disadvantages of traditional approaches

The PatFinder algorithm is highly intuitive, as it exploits the fact that within the MS muons bend in the η direction with an intensity inversely proportional to their transverse momentum. The search for coincidences within the correspondent p_T -dependent window is both fast and efficient, requiring only few FPGA clock cycles for trigger selection computation. The generalisation from a single track to multiple muon candidates is straightforward and can be simultaneously implemented in the FPGA. Moreover, PatFinder is the upgraded version of the Phase I L1 RPC trigger algorithm, which has proven to be both reliable and effective. At the same time, the window optimisation is performed based on muons originating from the IP, but does not account for more exotic signatures. Additionally, the algorithm strongly

depends on the detector's geometry and efficiency. If certain regions of the MS, particularly the older BM or BO chambers, begin losing efficiency, the trigger will inevitably be affected.

The Pattern-matching approach is even simpler, as it requires only a memory call to the database. Multiple muon candidates can be reconstructed simultaneously, without adding extra clock cycles. On the other hand, this approach is not only dependent on the detector geometry, limiting its ability to trigger muons from IP, but also relies heavily on dataset simulation. If a muon track is not correctly simulated in the dataset, for any reason, it will not be selected. In such cases, the look-up table database must be continuously monitored and updated to account for detector inefficiencies.

7.4 ML-based L0 RPC trigger algorithms

Traditional algorithms have several limitations: they rely heavily on dataset simulation, meaning they can only reconstruct tracks pointing to the IP, and they depend on detector efficiency. As a result, partial tracks and muons from secondary decays that are not aligned with IP may be discarded from the trigger selection. Moreover, an approach that does not require continuous performance monitoring and parameter optimisation is highly desirable. An ML-based algorithm can overcome these limitations, since it learns how to connect hits across different super-layers and reconstruct muon candidates from them. This approach is significantly more robust against unseen patterns, whether they arise from exotic signatures or from detector inefficiencies. For these reasons, we have developed new ML-based algorithms designed for implementation on FPGAs. The first ML approach explored the use of DNNs. The geometry of the RPC semi-sectors was arranged in a fixed-shape, two-bidimensional grid, where the hits were positioned. From another perspective, this grid can be interpreted as a black-and-white image, where black represents hits in the detector. This image is then processed by a Convolutional Neural Networks (CNNs) to identify the muon track. The network output consists of the kinematic variables of the muon. The first results obtained with a CNN for single muon track identification and reconstruction of transverse momentum and pseudorapidity are discussed in Ref. [233]. This work represents the first successful attempt at implementing a simple ML trigger algorithm on FPGA for the ATLAS L0 system. Within the ML approach, the generalisation to multiple muon candidates is not straightforward, as the number of predicted muons alters the shape of the output vector, necessitating architectural adjustments to the algorithm. Therefore, we adopted an incremental development strategy with different architectures: first, we demonstrated the feasibility of the approach with a CNN that reconstructed the p_T , the η coordinate and the charge q of single muon tracks; subsequently we extended the model to predict multiple muons. However, the final CNN architecture exhibits performance limitations, primarily due to difficulty in resolving overlapping muon tracks. Since the model is forced to analyse the entire RPC image, it loses the ability to distinguish partially overlapping tracks, reducing overall performance. For this reason, a final approach using GNNs is currently under development and preliminary results will be presented. All developed ML approaches focus on the

evaluation of the muon features correlated with the bending of charged particles inside the ATLAS barrel toroidal magnet, specifically p_T , η and q . In fact, in the ϕ direction the muon track follows a straight line, allowing traditional algorithms to identify candidates quickly and efficiently. In the older BM and BO chambers, the η and ϕ readouts are separate and distinct. However, in the new BI the η strips will be read at both edges, while the ϕ coordinate will be inferred from the timing difference between the two signals. This feature could be exploited to associate muon candidates identified by the trigger algorithm with the correct ϕ coordinate, but this possibility has not yet been studied.

7.4.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a specialised type of DNN designed for processing data with a grid-like topology, such as images [234, 235]. At the core of CNNs are *convolutional layers*, which apply a set of filters, also called *kernels*, to the input data to detect local patterns. Mathematically, a convolutional layer performs a cross-correlation operation, which can be described as:

$$(I \times K)_{i,j} = \sum_m \sum_n I(i+m, j+n) K(m, n) \quad (7.3)$$

where I is the input feature map at position (i, j) , while K represents the kernel applied to the input. In Eq. 7.3, the convolution is two-dimensional (2D), referring to the number of possible directions the kernel can move. However, the operation can be similarly extended to one-dimensional (1D) or three-dimensional (3D) convolutions. The convolution operation enables the network to learn filters that activate when they detect specific types of features, such as edges or textures, in the input image. This process is position independent, making CNNs translation-invariant. The output of a convolutional layer is a set of *feature maps*, which indicate the presence of detected features at different spatial locations in the input. The *stride* defines number of pixels by which the filter moves across the input data. *Padding* is the process of adding extra pixels, usually zeros, around the input data to control the spatial dimensions of the output feature map and to preserve its original size. The *receptive field* refers to the specific region of the input data that influences a particular activation.

Typically, after a convolution layer, a *pooling layer*, such as max or average pooling, is applied. Pooling operates over a portion of the feature map and results in reducing its dimensions.

CNN architectures usually consist of multiple stacked convolutional blocks, followed by a flattening layer that converts the feature map into a one-dimensional array, which is then passed to one or more DNN blocks, as shown in Fig. 7.9. From 2010 to 2017, a competition called ImageNet Large Scale Visual Recognition Challenge (ILSVRC) was held annually and contributed significantly to advancements in CNN architectures. The most notable winners of the ILSVRC include AlexNet [237], Visual Geometry Group (VGG) [238], Inception [239] and ResNet [240].

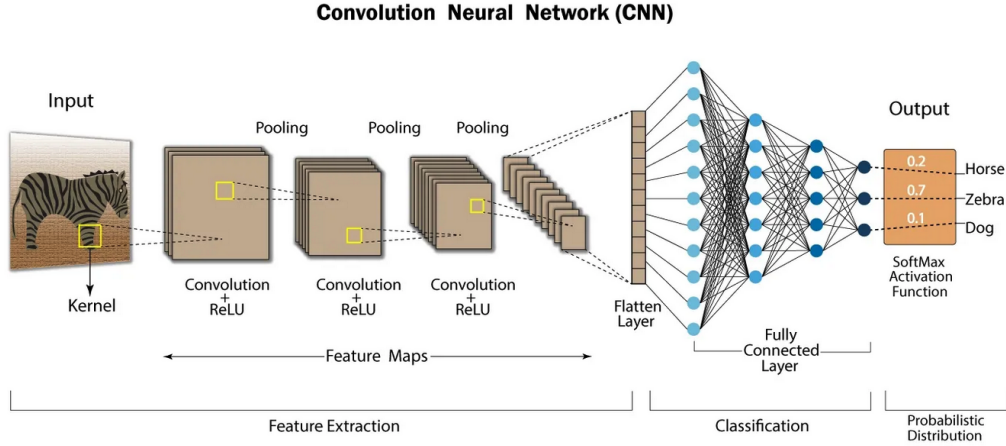


Figure 7.9. Sketch of a typical CNN. First convolutional layers are alternated with pooling layers; then there is a flatten layer and finally fully connected layers are used to produce the output. Figure from Ref. [236].

7.4.2 Experimental requirements

The L0 RPC trigger algorithm must meet several requirements, which can be grouped in two main categories: those related to the information needed for subsequent trigger steps and those dictated by experimental constraints. The RPC SL will provide 4/6 muon trigger candidates along with a flag that indicates the presence of more than 4/6 candidates. Three of these candidates, selected in decreasing priority order², will be sent to the MDT-TP for refined track reconstruction. After receiving the improved information from the MDT, the RPC SL will then transmit all 4/6 candidates to the MUCTPI. The data format of trigger candidates should contain the transverse momentum p_T , the pseudorapidity η , the azimuthal angle ϕ and the charge q of the muons.

Additionally the algorithm must satisfy the following conditions:

- **Limited resource occupancy:** the Xilinx Virtex Ultrascale+ XCVU13P, used for the SL, is a relatively large FPGA; however, it should accommodate all readout algorithms and the overall resource allocation, excluding the trigger algorithm, is reported in Tab. 7.2. Furthermore, since the SL is expected to remain operational until the end of Phase II, it is crucial to avoid occupying all available resources. Additional space should be reserved for potential modifications or logic updates that may be required to bypass sections of the algorithm in case of issues during data taking. For these reasons, it is recommended that overall SL FPGA occupancy does not exceed 50-60%, particularly for look-up tables (LUT in the Figure). Consequently, the trigger algorithm that will be located in SLR1 and SLR2 should ideally use only 35-45% of the available look-up tables.

²The ATLAS requirements do not specify the nature of the priority, which depends on the type of algorithm ultimately chosen for the trigger.

- **Maximum latency:** latency refers to the time required for the trigger algorithm to generate muon candidates. For the trigger pipeline to function correctly, the latency must remain fixed. The L0 RPC trigger algorithm is required to have a maximum latency of 390 ns; including all necessary preprocessing steps, if needed by the algorithm. The existing trigger logic operates at a frequency of 320 MHz, meaning that the maximum allowed latency corresponds to approximately 120 clock cycles. While it is possible to use multiple clock cycles within the same FPGA, this is not recommended.
- **Fake efficiency:** is defined as the probability of accepting events that contain only noise and no actual muons. Since the muons produced in pp collisions as minimum ionising particles, they deposit the little energy in the detector; however, muons with $p_T < 2.5\text{--}3\text{ GeV}$ do not reach the ATLAS MS. As a result, the majority of events processed by the trigger algorithms do not contain any muon. It is therefore crucial that the algorithm can accurately identify and discard events without muons. The fake efficiency needs to remain below 2‰.

Resource	SLR0	SLR1	SLR2	SLR3
LUT	24%	29%	29%	58%
FF	12%	15%	15%	13%
BRAM	1%	2%	2%	55%
CLB	44%	55%	52%	77%

Table 7.2. Resource utilisation of the FPGA for the RPC SL after the implementation of all the algorithms but the trigger, divided in the four SLR. LUT are the look-up tables, FFs are the flip flop, BRAM is the block RAM and CLBs are the configurable logic blocks. It is recommended not to overload the LUT occupancy, whose value should be kept below 50-60% for optimal and lasting operations.

7.5 Dataset for the ML trigger algorithm

A MC dataset has been generated and used for all trigger approaches. The dataset assumes an RPC efficiency of 100% and single muons originating from the IP have been simulated. Since digitisation of the BI RPC layers is not yet available, the first three layers use BI MDT hits instead. The simulated muons follow a uniform transverse momentum distribution, with $p_T \in [0, 50]\text{ GeV}$. However, due to the limited spatial resolution of RPCs (approximately 1 cm), muon tracks appear as a straight line for $p_T > 30\text{ GeV}$. Therefore, only muons with $p_T \in [0, 30]\text{ GeV}$ have been selected and their distribution is shown in Fig. 7.10. Muons with $p_T < 2.5\text{--}3\text{ GeV}$ do not reach the MS and at low p_T the distribution undergoes smearing. The single muons are uniformly generated inside the MS, then only large sectors are considered and one side is mirrored onto the other in order to increase statistics.

The ultimate goal is to train separate ML models for each sector, allowing the model to learn the specific characteristics of the detector. However, the available dataset contains 32 M events distributed across the entire detector, corresponding to 1 M events per sector in each side. For the initial feasibility studies using the ML approach, we decide to focus exclusively on muons from large sectors, as they are the most regular and similar among themselves. The η distribution of the selected events is shown in Fig. 7.10.

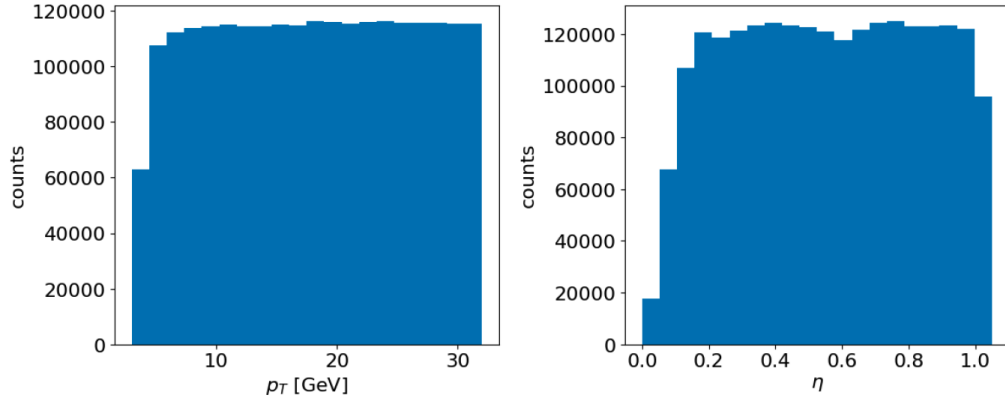


Figure 7.10. p_T distribution, on the left, and pseudorapidity distribution, on the right, of the muons in the dataset for the L0 RPC trigger algorithm.

Each dataset event contains all the hits left by the muon, including their global x - y - z coordinates, the RPC layer, the identifier number of the strip, the type of the strip, whether is an η or ϕ strip, altogether with the charge, p_T , η and ϕ of the muon. Two datasets have been created, one for the single muon studies and one for the multi-muon case. The single muon dataset contains approximately 6 M events; the multi-muon is constructed from the first by superimposing two or three single muon events into one. The number of single, di- and three-muons events is balanced within the dataset, ensuring that each category contains an equal number of events. As a result, the total number of events available for this dataset is 1 M events. For each dataset, 80% of the events are devoted to training, 10% to validation and 10% to test.

7.5.1 Noise modelling in the dataset

The main sources of noise are *cavern background* and *punch-through particles*. Cavern background originates from particles either slowed down by the interactions with the detector material and not arriving in time with the TDAQ acceptance window or accumulated from many collisions that finally reach the MS. These particles may have been produced in several collisions, both before and after the primary collision of interest, and primarily consist of low energy photons, neutrons and electrons. In particular, low energy neutrons, which have large elastic scattering cross-section, undergo multiple scattering inside the detector. This process creates a gas-like distribution of long lived neutrons inside the detector cavern. These neutrons can

excite detector atoms, which in turn emit photons that may be detected inside the MS. Punch-through particles, on the other hand, are charged particles either trapped within ATLAS magnet system or produced by highly energetic jets, whose decay products are capable of reaching the MS. The two types of noise leave distinct signatures: cavern background consists of randomly distributed hits in the detector, typically forming clusters of 1-2 strips, making it difficult to suppress; punch-through particles tend to produce short tracks, particularly in the BI stations. By requiring a 2/3 coincidence within the BI triplets, the noise from the punch-through particles is almost entirely eliminated. The expected hit rates in the BM and BO regions of the RPCs during Phase II are shown in Fig. 7.11 and 7.12, respectively. For the BI, it is reasonable to estimate the hit rate based on MDT measurements in the same region (Fig. 7.13). The tables show the total expected

Sector Φ Id.	RPC unit Id. along Z direction																												Average
	-6.2	-6.1	-5.0	-4.0	-3.2	-3.1	-2.2	-2.1	-1.2	-1.1	1.1	1.2	2.1	2.2	3.1	3.2	4.0	5.0	6.1	6.2									
01.01	342	280	301	225	145	114	128	101		61	61	71	113	122	127	136	214	276	269	285									180
01.02	293	281	303	218	159	133	129	143		76	71	75	127	143	140	148	215	295	278	297									188
2	168	204	188	138	109	90	77	63	55	56	45	48	61	79	94	104	140	207	196	152									115
03.01	297	296	281	198	148	128	119	119	68	67	65	71	131	125	114	125	207	329	268	290									177
03.02	300	243	277	210	151	129	155	122	85	75	75	70	122	127	152	132	207	315	243	299									179
4	112	166	158	151	101	83	65	77	41	53	46	41	73	68	92	108	160	196	175	112									101
05.01	171	173	263	138	105	102	140	127	68	60	60	69	124	177	102	137	185	290	173	171									149
05.02	227	198	237	158	109	105	136	143	77	61	63	71	111	136	108	141	200	267	255	282									159
6	175	186	208	163	105	95	90	95	77	59	52	59	84	81	106	124	189	200	227	167									131
07.01	305	263	288	191	154	129	131	114		76	78		122	139	124	148	185	261	268	305									183
07.02	327	258	216	203	141	112	129	108		74	77		105	114	112	152	184	278	276	279									175
8	146	196	195	161	103	85	80	70	50	57	54	54	67	74	85	108	168	196	194	156									118
09.01	319	246	301	206	155	117	149	119		46	64		106	134	124	135	197	283	262	297									181
09.02	347	258	287	205	143	95	107	103		58	67		99	112	95	137	188	285	265	292									174
10	174	201	207	147	99	86	68	71	46	43	41	50	64	69	80	103	148	193	201	170									115
11.01	308	244	237	157	97	84	81	87		40	43		83	94	92	94	148	227	215	278									132
11.02	196	193	157	105	78	66	55	57		33	31		50	62	58	71	98	151	160	185									98
12									80	81	66	51	36	36	51	75	87	80											64
13.01	291	278	253		140	102	96	84	43	41	47	50	87	95	99	123		249	263	319									149
13.02	299	264	262		104	97	105	86	49	48	50	56	93	103	97	110		252	227	294									146
14						142	68	64	52	41	40	49	63	68	136														76
15.01	196	221	148	113	76	71	67	49		38	36		50	59	86	87	104	156	173	196									104
15.02	183	159	246	164	116	98	103	75		44	43		75	106	112	107	158	248	159	183									133
16	173	214	216	173	108	89	54	75	56	59	50	50	77	54	87	103	177	209	208	154									124
Average	229	223	234	167	118	101	96	88	56	52	51	56	86	97	103	118	171	240	221	221									137

Figure 7.11. Expected hit rates in the BM chambers extrapolated for $7.5 \times 10^{34} \text{ cm}^{-2} \text{ s}^{-1}$ luminosity, including a safety factor of 2, from the analogous rates measured in Run 1. The rates are expressed in Hz cm^{-2} , sector by sector (rows), along the z direction (columns). Figure from Ref. [227].

hit rates, making it impossible to disentangle the uniform contribution of the cavern background from punch-through particles, which are more prominent near the end-caps.

Since the dataset only includes simulated muons, it is crucial that any ML model is trained with realistic background conditions. Otherwise, the model may exhibit unpredictable behaviour when applied to real experimental data. To address this, a noise simulation has been incorporated into the dataset, following the procedure outlined below:

- the RPC noise is simulated in clusters of strips following a normal distribution with an expected value of $\mu = 1.5$ and a standard deviation of $\sigma = 0.15$;
- the number of expected clusters in a single sector follows a Poisson distribution with a mean value of $\lambda = 20$;

Sector	RPC unit Id. along Z direction																																		Average
Φ id.	-8	-7	-6	-5	-4	-3	-2	-1	0	1	1.1	1.2	2	2.1	2.2	3	3.1	3.2	4	4.1	4.2	5	5.1	5.2	6	6.1	6.2	7	8						
1.1	233	213	210	218	181	198	151	130	105	90	83	23	60	76	89	95	125	193	196	183	217	227	224	183							154				
1.2	238	227	234	226	197	194	172	138	106	92	80	67	64	78	93	103	129	240	207	187	224	218	221	218							165				
2.1	155	221	141	136	118	121	97	71	65	63	47	33	41	47	60	60	71	85	114	115	140	150	116	311							107				
2.2	76	125	142	152	123	131	104	76	65	63	52	45	50	50	64	62	72	101	124	118	147	148	127	82							96				
3.1	207	251	204	202	168	221	206	127	95	106	82	69	69	79	98	103	151	212	171	172	184	201	282	218							162				
3.2	227	264	216	220	181	147	162	142	114	97	87	73	67	82	108	105	135	175	178	177	194	204	252	222							160				
4.1	302	120	135	135	119	121	108	78	73	73	59	46	58	59	66	69	76	98	111	119	136	136	111	296							113				
4.2	76	130	143	138	129	121	110	80	78	72	59	55	52	61	76	74	79	99	120	118	133	147	132	78							98				
5.1	218	191	199	191	179	208	159	137	118	102	102	95	85	97	125	123	119	142	194	179	208	198	226	191							158				
5.2	192	222	207	202	194	206	205	137	119	125	125	93	82	95	124	113	120	152	176	164	210	196	212	234							163				
6.1	224	103	123	115	105	108	90	63	77	73	58	47	48	56	66	76	68	87	101	101	117	124	100	276							100				
6.2	61	105	127	116	101	84	87	61	67	67	51	41	44	51	62	69	61	82	95	102	117	106	112	63							180				
7.1	209	209	196	183	171	169	148	144	98	84	79	64	65	75	88	93	117	156	169	169	186	208	228	241							148				
7.2	210	204	188	192	165	175	150	144	96	90	74	60	62	72	91	82	112	142	163	170	199	198	228	225							145				
8.1	283	116	136	135	118	109	99	72	62	62	51	44	42	51	59	67	79	98	114	111	146	161	124	310							110				
8.2	71	123	127	123	112	110	85	62	59	59	51	43	43	49	55	59	69	94	108	109	133	138	134	69							87				
9.1	202	203	206	198	178	179	158	120	93	86	91	68	63	80	86	94	116	173	181	188	189	202	207	285							152				
9.2	201	204	192	197	186	166	149	111	90	79	87	65	61	70	86	89	127	151	166	182	211	196	182	327							149				
10.1	299	124	139	130	104	113	95	70	58	55	46	38	38	42	53	59	66	89	110	105	128	142	117	318							106				
10.2	64	117	128	115	94	103	85	59	47	45	39	34	33	37	51	52	58	77	100	102	125	131	131	67							79				
11.1	203	188	180	168	172	173	121	96	73	72	62	49	48	63	66	80	101	143	156	164	171	200	206	212							132				
11.2	153	149	159	128	135	120	95	73	64	54	50	41	45	57	56	61	81	109	118	128	125	148	140	155							102				
12.1		36	104		72	86	83			49	66	59			29	31	26	27	28	28									89	40	56				
12.2		120	88		73	75	77			52	48	84			29	28	27	29	29	26									102	109	63				
13.1				236	230	189	176	169	148	103	83	82	79	62	49	59	68	94	77	81	107	150	163	170	187	227					225	133			
13.2				202	202	188	182	165	147	104	85	81	76	62	51	63	81	80	77	88	103	151	165	177	192	230					221	133			
14.1			36	86		77	78	79		56	45	75			31	33	23	32	28	25										101	120	60			
14.2			120	95		72	88	82		49	49	69			27	31	22	32	26	30										94	36	60			
15.1				177	164	160	143	133	122	100	80	70	59	55	48	46	56	57	65	84	107	119	123	134	143	169					164	107			
15.2				212	210	194	186	181	166	152	105	91	76	67	54	63	67	77	82	105	129	164	153	177	150	200					224	137			
16.1				145	115	123	133	103	99	90	63	54	53	43	35	34	43	50	57	62	89	107	101	125	129	125					278	94			
16.2				83	59	132	154	113	132	99	77	56	61	49	40	43	49	59	61	77	94	123	121	131	148	128					78	90			
Average	78	93	184	159	158	154	146	134	115	93	81	70	62	49	51	59	70	79	90	117	131	142	152	158	162	206	97	76			116				

Figure 7.12. Expected hit rates in the BO chambers extrapolated for $7.5 \times 10^{34} \text{ cm}^{-2} \text{ s}^{-1}$ luminosity, including a safety factor of 2, from the analogous rates measured in Run 1. The rates are expressed in Hz cm^{-2} , sector by sector (rows), along the z direction (columns). Figure from Ref. [227].

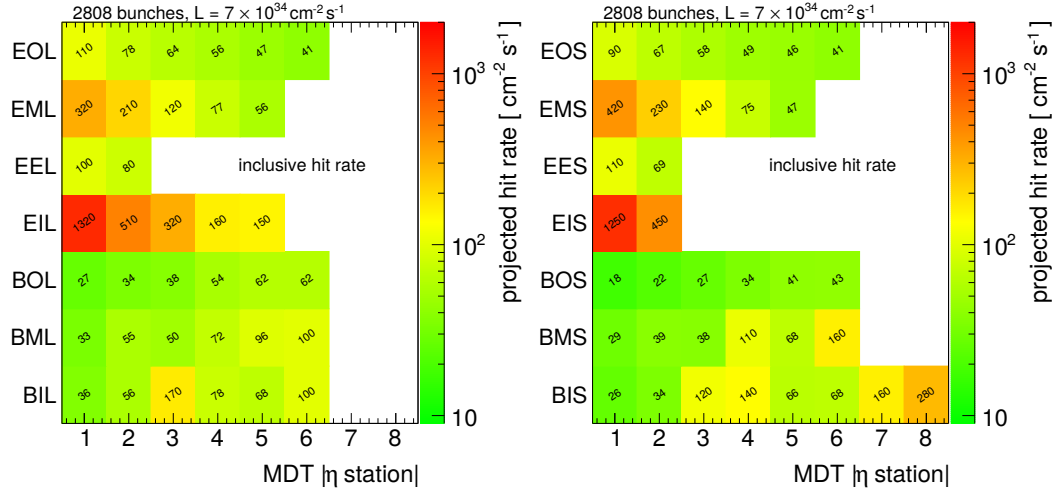


Figure 7.13. Expected hit rates in the MDT chambers extrapolated for $7.5 \times 10^{34} \text{ cm}^{-2} \text{ s}^{-1}$ luminosity from the analogous rates measured in Run 2. The rates are expressed for different chamber types (rows), along the η stations (columns). It is realistic to use the numbers obtained for the “BIL” and “BIS” MDT types for the BI large and small sectors of the RPC, respectively. Figure from Ref. [227].

- the spatial distribution of clusters is assumed to be uniform in the x - y - z coordinates.

The simulated noise hits are then superimposed onto events in each dataset. This approach serves as a preliminary method to account for an averaged noise contribution. However, future iterations will transition towards a more realistic simulation,

incorporating sector by sector and station by station values as presented in Fig. 7.11, 7.12 and 7.13.

7.5.2 Images dataset creation

To generate an image dataset for the CNN model, a fixed rectangular shape must be defined. The actual RPC stations used in the MS vary in size depending on experimental constraints, available space and detector mounting structures. However, since the pseudorapidity range for the MS barrel is well-defined, we discretised the η coordinate as follows:

$$\eta_{\text{index}} = 384 \cdot \frac{|\eta| - \eta_{\min}}{\eta_{\max} - \eta_{\min}} \quad (7.4)$$

where $\eta_{\min} = 0$ and $\eta_{\max} = 1.05$; 384 represents the average number of strips in a generic RPC layer; the actual number of strips can vary significantly.

The dataset has been created with the four super-layers obtained by applying logic coincidences between RPC doublets and triplets, as shown in Fig. 7.14. The im-

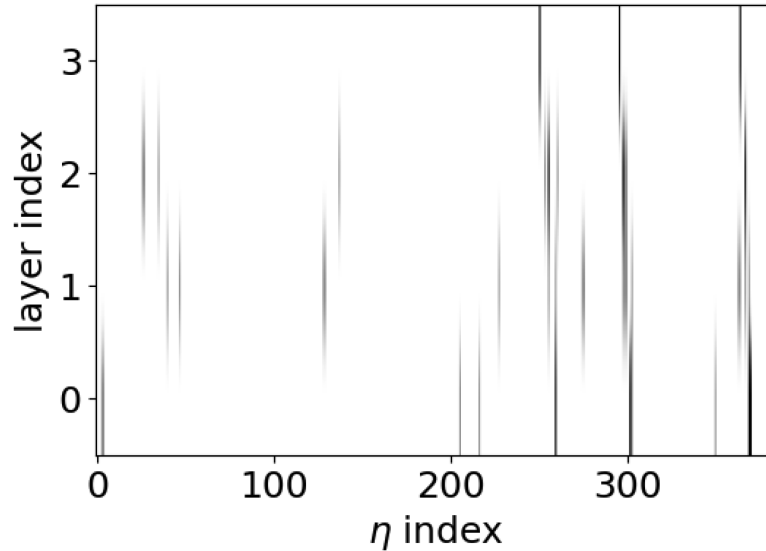


Figure 7.14. Image of the image dataset with three muons: the grid consists of 4 super-layers and 384 discretised η_{index} . Noise has been superimposed to images in the datasets.

ages were further reduced in size, by introducing a colour channel into the two-dimensional image representation. In this format, pixels corresponding to even η_{index} values are assigned to channel 0, while pixels corresponding to odd η_{index} values are assigned to channel 1. The final dataset format is (4, 192, 2).

7.5.3 Graph dataset creation

In the graph dataset, each node represents a hit in one of the four RPC super-layers; two nodes are connected by an edge if they satisfy a distance-based criterion. The node features consist of the hit distance from IP and its pseudorapidity, (r ,

η). The connection criterion is derived from the maximum curvature a muon can exhibit in the MS. Specifically, considering a muon that enters the RPC BI plane at longitudinal angle θ and exits the BO plane at an angle θ' (Fig. 7.15), the following relation can be derived:

$$d = R \sin\left(\frac{\pi}{2} - \theta'\right) - R \sin\left(\frac{\pi}{2} - \theta\right) = \frac{p_T [\cos \theta' - \cos \theta]}{0.3 B} \quad (7.5)$$

where d is the distance between the planes and the final step follows from Eq. 2.9. Since $\cos \theta = \tanh \eta$, the connection requirement between two nodes can be expressed as:

$$|\tanh \eta' - \tanh \eta| \leq k \frac{0.3 B d}{p_T} \quad (7.6)$$

Here the magnetic field is assumed to be uniform, with $B = 0.5$ T, while the minimum transverse momentum required for a muon to reach the MS is set at $p_T = 3$ GeV. The parameter k regulates the strictness of connection requirement for hits associated with muons of at least 3 GeV. To simulate real experimental conditions,

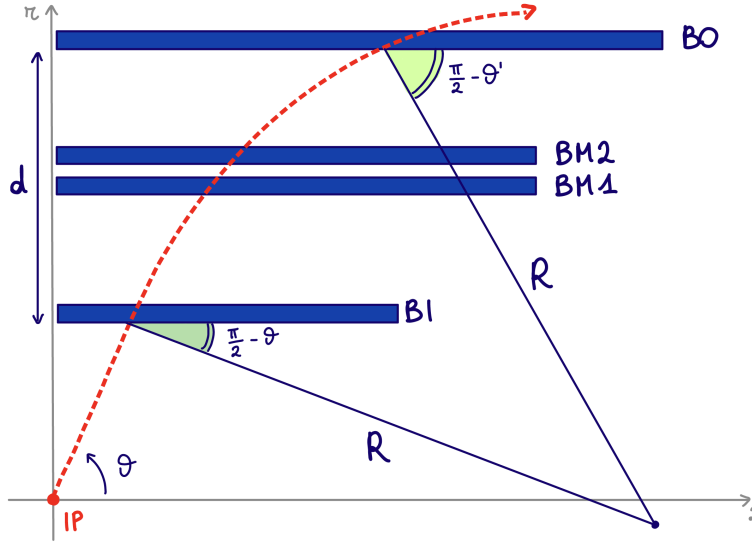


Figure 7.15. Visual representation of the relation between the initial and exit angle θ and θ' , the curvature radius R and the distance d between RPC planes.

noise hits have been included in the graph. Additionally, only events with a number of nodes in the range $[3, 50]$ have been considered. The node and edge distributions of the dataset for $k = 1$ are displayed in Fig. 7.16. On average, each graph contains 22 nodes, while the mean number of edges is 159. Finally, each node is assigned a label: 0 if the node corresponds to a noise hit, 1 if the node originates from a muon hit.

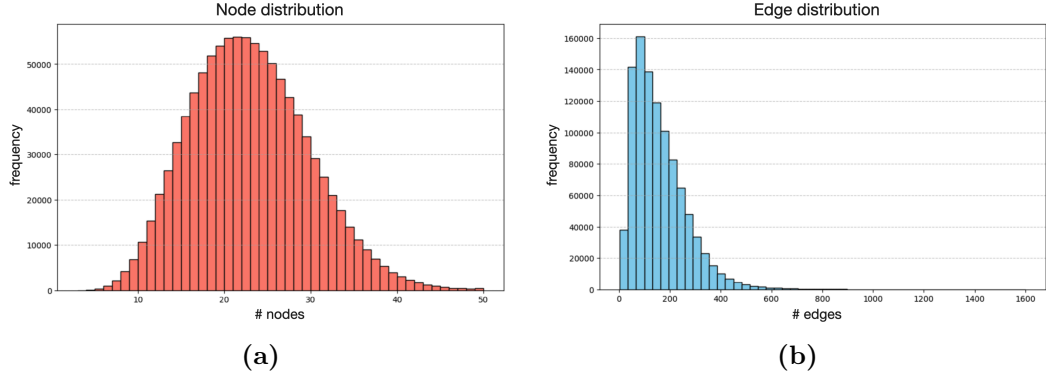


Figure 7.16. Distribution of nodes (Fig. 7.16a) and edges (Fig. 7.16b) of the events in the graph dataset for $k = 1$.

7.6 Single and multi-muon reconstruction with CNNs

An essential step in developing a high-performance L0 RPC trigger algorithm using CNNs is to begin with the single muon tracks dataset. The regression task of identifying the muon and predicting (p_T, η, q) from the dataset of images is relatively straightforward for a CNN architecture. However, its implementation on the FPGA, combined with the extreme experimental requirements described in Sec. 7.4.2, necessitates a highly lightweight model, constrained to only few hundreds of parameters. Unfortunately, drastic reductions in the model size inevitably lead to a decrease in performance. To address this, compression techniques have been employed to find an optimal balance between model performance and feasibility for FPGA implementation.

7.6.1 Compression techniques

Two different compression techniques have been applied simultaneously during the CNN training: Quantization Aware Training to reduce model size and Knowledge Distillation to mitigate performance degradation.

Quantization Aware Training

Quantization Aware Training (QAT) is a compression technique that restricts the model weights and activations to a reduced number of bits [241, 242]. This constraint is enforced during training, ensuring that the loss optimisation and gradient descent account for the bit-width limitations. As a result, QAT yields better performance compared to simply limiting the precision of the weights *after* training. QAT significantly reduces the model's memory occupancy without altering the number of parameters. The compression is applied using the `quantized_bits` function implemented in Python with the QKeras library [243]. The Quantization of the weights is implemented as $y_q = a_q x + b_q$, where:

$$a_q = 2^{\#bits-1}, \quad b_q = 0 \quad (7.7)$$

The resulting range of values that weights can assume is symmetric, given by $[-2^{\#bits}, 2^{\#bits-1}]$. During training, the weights and activations are computed with the desired precision, including compression. However, during back-propagation the quantisation is replaced by the identity since the quantiser is not differentiable.

Knowledge Distillation

Knowledge Distillation (KD) is a technique that employs two models to transfer information from one to the other [244]. First, a large model, referred to as the *Teacher*, is trained to achieve optimal performance on the task. Then, a model with smaller dimension, the *Student*, is guided during its training process to achieve faster convergence by leveraging knowledge from the Teacher. The guidance is provided in the form of a penalty term, added to the Student's loss, whenever its performance falls below a defined threshold compared to the Teacher's performance, as shown in Fig. 7.17. Mathematically, this is expressed as:

$$KD(x) = \begin{cases} S(x) + \sum_{i=1}^n \beta_i H_i(x) & \text{if } S(x) < (1 + \gamma) T(x) \\ S(x) & \text{otherwise} \end{cases} \quad (7.8)$$

where $S(x)$ and $T(x)$ represent the Student and Teacher loss, respectively; $H_i(x)$ denotes the n hints provided by the Teacher and regulated by the β_i parameters. The parameter γ controls the strength of the penalty imposed on the Student model. The specific choice of Teacher and Student architectures, as well as the positioning and nature of hints, their number n and the values of β and γ , must be optimised for the given application. Since Teacher and Student models may differ significantly, adaptation blocks may be required to ensure valid comparisons between their respective losses.

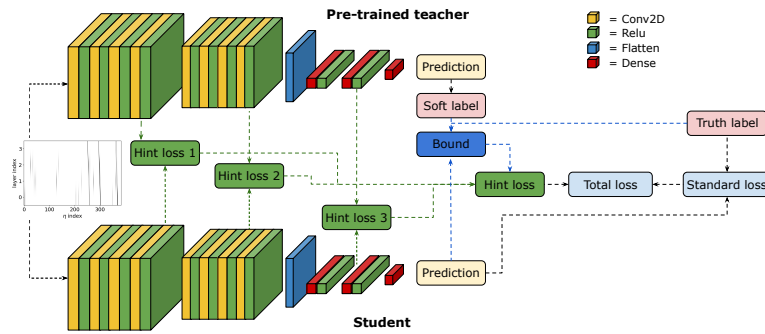


Figure 7.17. Representation of the KD used for the muon identification and reconstruction. Figure adapted from Ref. [233].

7.6.2 CNN models

The developed CNN models follow a simplified VGG architecture and use the Keras [245] and TensorFlow [246] libraries. Two Teacher and two Student models

have been created, one for the single muon reconstruction and one for the prediction of up to three muons. From these base architectures, two additional quantised Student models have been developed, using the analogous quantised convolutional and linear layers and activation functions in QKeras library. Teacher and Student models share a similar structure, with different kernel sizes and number of filters; in fact, it has been observed that the information distillation works better when the models are not so dissimilar. All models consist of two convolutional blocks and three parallel dense blocks, as shown in Fig. 7.18. In each convolutional block,

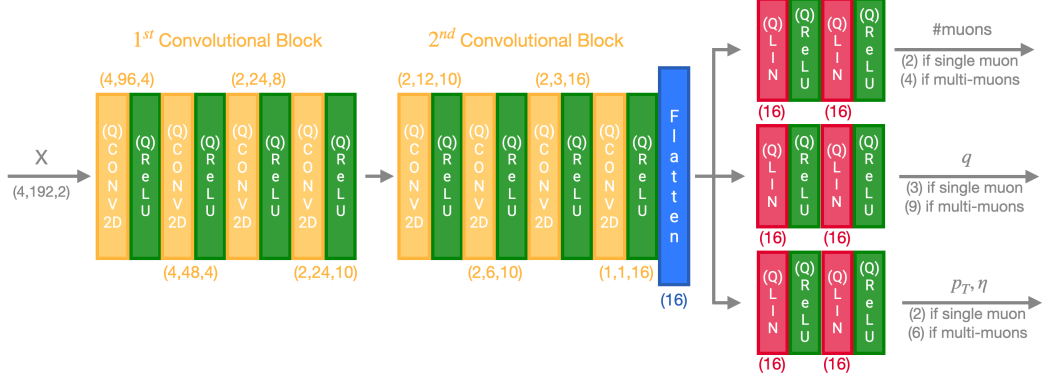


Figure 7.18. Architecture of the CNN Student model. After the flatten layer the hidden representation is passed to three separate but analogous block: one for the prediction of the number of muons, one for the charge and one for the regression task on p_T and η . The dimensions of the arrays at the end of each layer are shown (the batch dimension is omitted).

a zero padding layer, a two-dimensional convolution layer and a ReLU activation function are repeated four times. The convolutional layers are implemented with the Conv2D function [247], the kernel size is $(3, 3)$ but in the last convolution of the first block where the kernel has a $(2, 2)$ size. This change in size is due to the specific image shape that constraints the kernel dimension. The filter in the Teacher models is set to 32, while in the Student models varies from 4 in the first convolutional layers to 16 in the last ones. After the two convolutional blocks that incrementally decrease the image size, a flatten layer is used to create a one-dimensional array. The obtained array has length 16 in the case of the Student models and 32 for the Teachers. The output of the flatten layer is then transmitted to three independent dense blocks. One block is used for classification, predicting the number of muons, and exits with 2 or 4 neurons, depending on the single or three-muons task. Another block is employed for the charge prediction: for each muon, the charge can take the values $\{-1, 0, 1\}$; 0 is the value set when there is no muon in the image. The last block performs the regression task to predict the transverse momentum and pseudorapidity of each muon. All dense blocks in the Teacher architectures have 50 neurons, 16 in the Student models. The final number of parameters for each model is shown in Tab. 7.3.

Model	Single muon	multi-muon
Teacher	91 501	92 113
Student	8 177	8 381

Table 7.3. Number of parameters for the single muon and multi-muon Teacher and Student models. Quantisation do not add any parameter, therefore the values are valid also in the case of quantised Students.

7.7 Single muon CNN performance

The performance of the quantised Student CNN models after QAT is evaluated separately for each task. In the classification task (Fig. 7.19a), the quantised Student successfully identifies the presence of the muon with high accuracy, achieving a fake efficiency of 0.8‰ well below the 2‰ requirement. This demonstrates the model’s robustness in filtering out noisy event. The charge prediction is also performed effectively in the most cases, as illustrated in Fig. 7.19b. Events containing only noise, which correspond to a true charge value of 0, can be leveraged as an additional constraint to further limit fake efficiency. This constraint could be exploited if performance degradation occurs in the multi-muon scenario, or it can be omitted, if not needed. The model also performs well in pseudorapidity prediction; however,

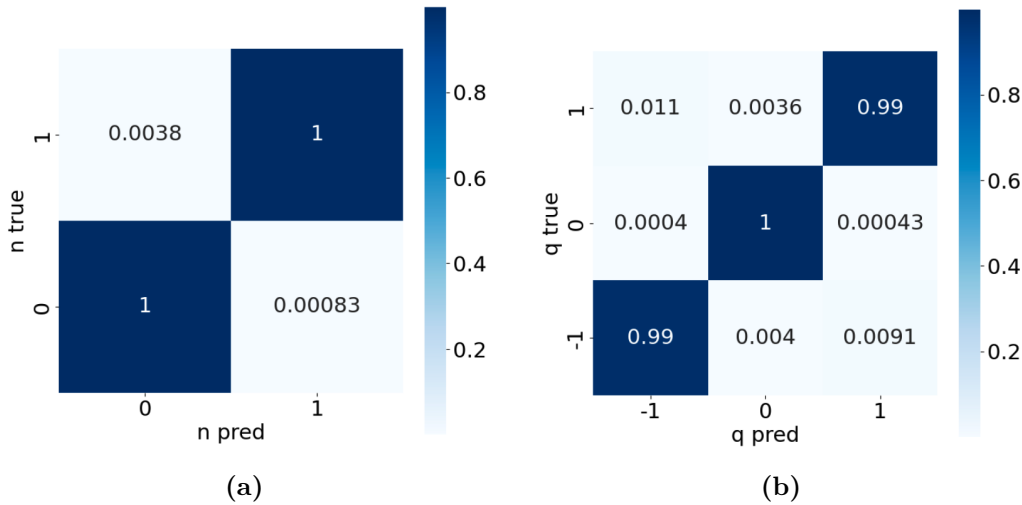


Figure 7.19. Confusion matrices regarding the classification performance of the single muon Student model after QAT and KD: in Fig. 7.19a, the prediction on the number of tracks; in Fig. 7.19b, the charge prediction.

the result is not particularly surprising, as η is largely determined by the muon entry trajectory into the MS. In contrast, the transverse momentum prediction is more complex and is best evaluated through efficiency curves. Given a threshold, the ideal behaviour is to exhibit low efficiency below threshold and high efficiency above. Fig. 7.20 and 7.21 show the efficiency plots at 6 and 14 GeV thresholds are shown for the different Student models. The Student without any compression

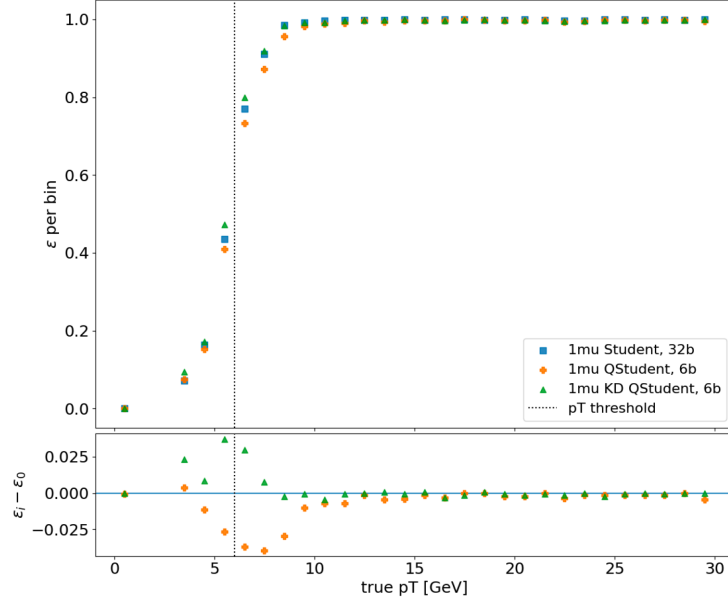


Figure 7.20. Single muon efficiency curves for the Student CNN models with p_T threshold at 6 GeV. In blue, the curve for the Student before any compression; in orange, the curve for the performance after the QAT with a 6 bits compression; in green, the efficiency curve after the QAT and the KD. In the plot below it is shown the difference between the efficiencies of the compressed models with respect to the reference Student model without any compression.

(blue squares) demonstrates the best performance; when quantised with a 6-bit precision (orange diamonds), the model performance is degraded, with a shallower slope and lower plateau values. KD (green triangles) effectively recovers the lost performance, restoring efficiency to a level close to that of the original uncompressed model.

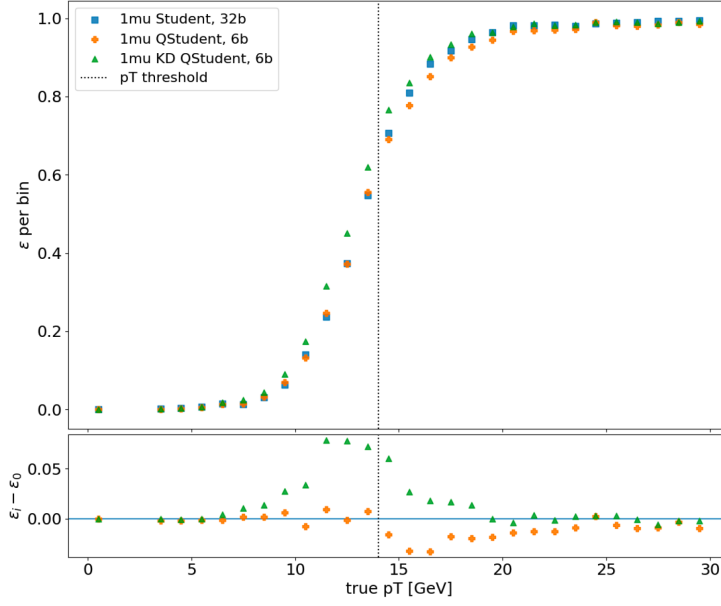


Figure 7.21. Single muon efficiency curves for the Student CNN models with p_T threshold at 14 GeV. In blue, the curve for the Student before any compression; in orange, the curve for the performance after the QAT with a 6 bits compression; in green, the efficiency curve after the QAT and the KD. In the plot below it is shown the difference between the efficiencies of the compressed models with respect to the reference Student model without any compression.

7.8 Multi-muon CNNs performance

Several challenges arise when transitioning from single to multi-muon track reconstruction, significantly impacting the model's performance. Unlike traditional algorithms, which evaluate the presence of tracks by considering all hits in the entire semi-sector or in the pivot plane, the ML approach requires an ordered representation of muons in the image. This ordering is crucial because a CNN could correctly predict the transverse momentum values but swap the positions of muons in the output neurons, causing the loss function to be wrongly penalised. To mitigate this issue, the best approach is to position the tracks in increasing pseudorapidity order. Since η is always well reconstructed by the models, this ordering ensures a consistent mapping between input tracks and output neurons, reducing misidentifications due to track swapping. The efficiency curves for the Teacher model are shown in Fig. 7.22, for each muon in the events. The points below threshold exhibit a noticeable degradation, with high efficiency values for the bins at 2-3 GeV in the prediction of second and third muon. This effect is partially due to the small statistics contributing to that part of the plot: those bins in general are sensitive to the smearing observed in Fig. 7.10. Additionally, an image with three (two) muons contributes three times (twice) in the plot, meaning that the green triangles (orange diamonds) have one-third (half) the statistics of the blue squares. At the same time, a degradation so evident in the Teacher's performance can only worsen in the Student's and should be investigated. A likely cause is that the two and three muon

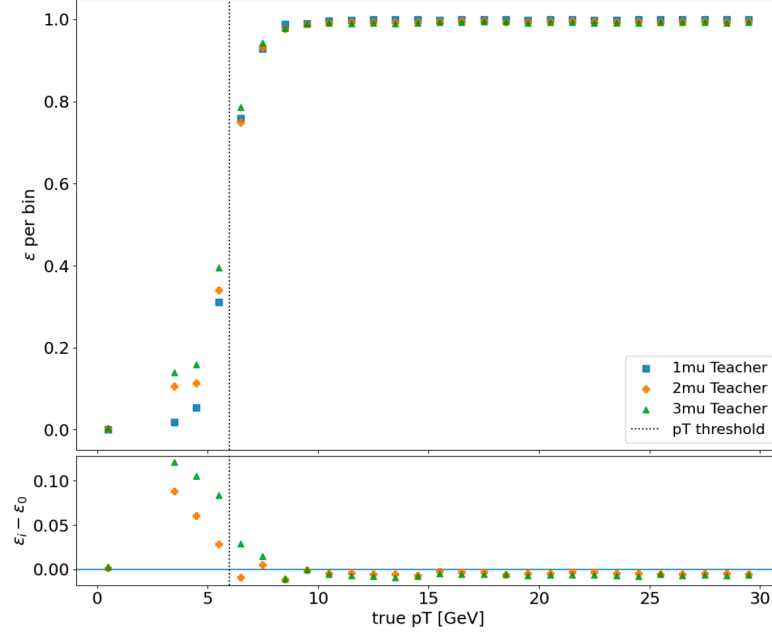


Figure 7.22. Efficiency curves at 6 GeV threshold for the Teacher model for the first (blue squares), second (orange diamonds) and third (green triangles) muons, ordered in descending pseudorapidity, in the image. At low p_T , the performance on the second and third muons deteriorates significantly, while reaching similar plateau efficiencies at high p_T . The plot below shows the difference between the single muon efficiency and the multi-muon case.

images are obtained from superimposing multiple single muon images, with possible overlapping of the trajectories. Therefore, we evaluated the Teacher's efficiency curves, training on the dataset from which only two muons have been considered and a criterion on the minimum pseudorapidity distance $|\Delta\eta|_{min}$ between the tracks has been set. The plots for $|\Delta\eta|_{min} = 0.02, 0.06, 0.1$ are shown in Fig. 7.23: when the requirement is more stringent, at higher $|\Delta\eta|_{min}$, less muons are overlapping and the performance is recovered.

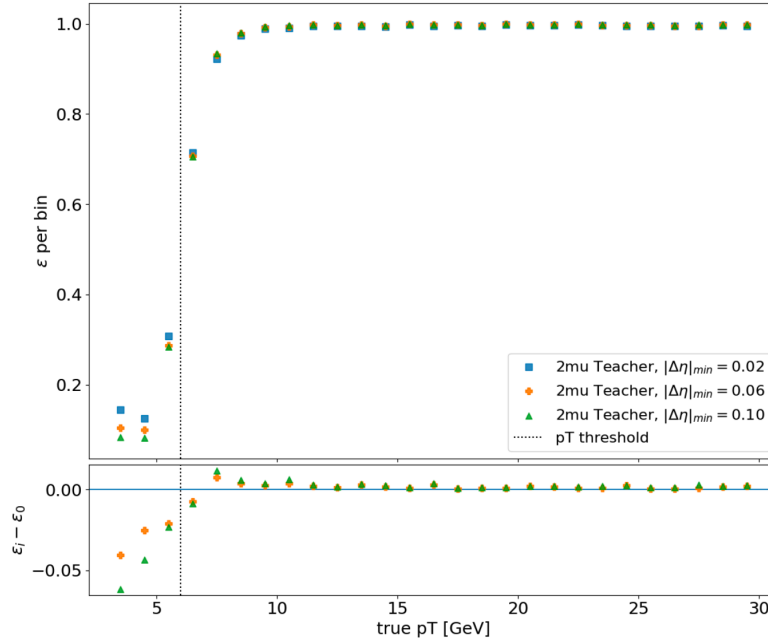


Figure 7.23. Efficiency curves for two muons in the events, when requiring different safety distances in pseudorapidity to avoid overlapping tracks.

7.9 GNN for single muon track identification

The CNN models developed have demonstrated great potential in reconstructing the muon kinematic variables. However, an intrinsic issue related to the image construction process makes this approach challenging to implement in practice. Beyond the overlapping track problem, another major challenge is the unusual shape of the images used as input. Typically, CNN architectures are optimised for more square-shaped images and deviating from this standard has complicated both the parameter tuning and the fulfilment of timing and resource occupancy constraints which are required for implementation on FPGA. A critical limitation arises in the first convolutional layer, which processes a (4, 192, 2) grid. Due to its elongated shape, reading the full image requires multiple clock cycles, making it the main bottleneck for real-time implementation. While this can be mitigated by optimising the HDL code and parallelising the input readout, it further highlights another fundamental inefficiency: most of the image is empty and has no hits. As a result, the CNN models waste valuable processing time and resources analysing image regions where only noise is present. A graph-based representation offers a more efficient alternative, as it allows the model to focus only on actual hits, significantly reducing the number of input features. Instead of processing 1536 input pixels, a graph-based model operates on only the few detector hits, making it computationally more efficient. Additionally, the track structure is suited to be represented as a graph, where nodes corresponds to detector hits and edges are defined based on spatial proximity. For these reasons, we transitioned from a CNN approach to a GNN model. GNN models can be highly complex and designing an architecture suitable for FPGA implementation is particularly challenging. However, it is not impossible, as

demonstrated in previous works [248, 249]. We structured the model as follows:

- a simple GNN block is used for edge prediction, determining which edges, and thus nodes, constitute a muon track;
- a graph pooling layer to transform the graph into a fixed-shape array;
- a dense block with a structure similar to that in the CNN (Fig. 7.18).

Inspired by the Interaction Network (IntNet) model from Ref. [250], we developed the GNN architecture and evaluated its performance before integrating it with the dense network, which is currently under development. The IntNet message passing mechanism consists of a node update function ϕ_O sandwiched between two edges update functions $\phi_{R,1}$ and $\phi_{R,2}$, as illustrated in Fig. 7.24:

$$\begin{aligned}
 \text{edge update 1:} \quad & a'_{ij} = \phi_{R,1}(x_i, x_j, a_{ij}) \\
 \text{edge aggregation:} \quad & \bar{a}_i' = \sum_{j \in \mathcal{N}(i)} a'_{ij} \\
 \text{node update:} \quad & x'_i = \phi_O(x_i, \bar{a}_i') \\
 \text{edge update 2:} \quad & a''_{ij} = \phi_{R,2}(x_i, x_j, a'_{ij})
 \end{aligned} \tag{7.9}$$

Both edge and node updates are implemented using an MLP consisting of a lin-

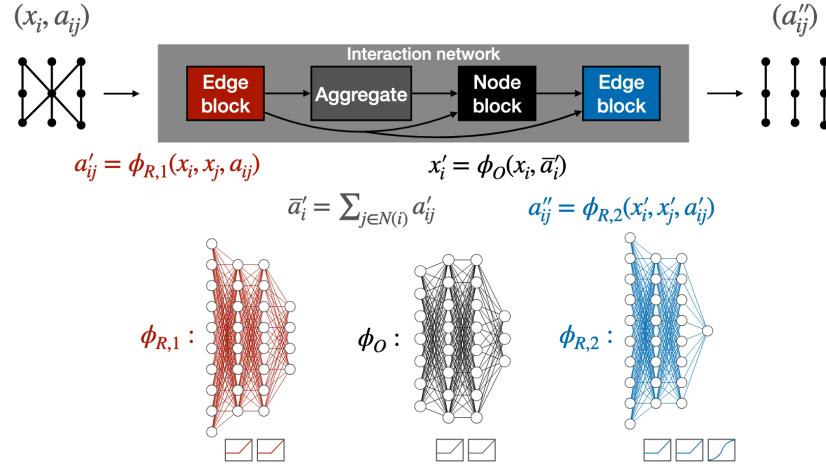


Figure 7.24. Schematics of the IntNet block. Figure from Ref. [250].

ear layer, batch normalisation and a ReLU activation function; this MLP block is applied twice. The final output representation takes the following terms:

- nodes are embedded in a $(n, 16)$ representation, where n is the number of nodes;
- edges are processed into a $(e, 2)$ representation, where e being the number of edges and the two final neurons represent the probability of an edge connecting two muon hits, as determined by a sigmoid activation function in the final layer.

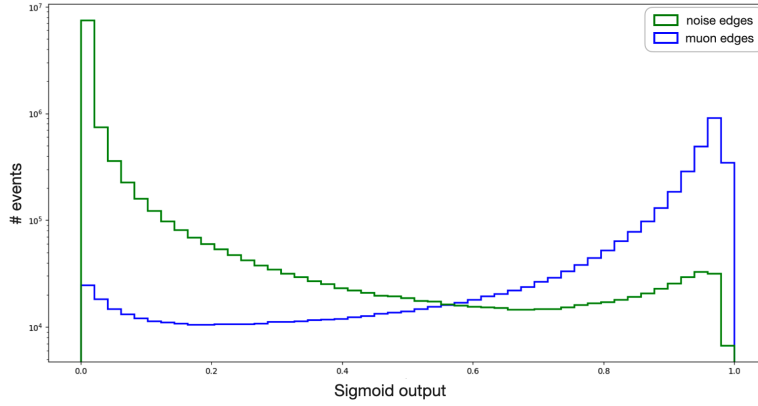


Figure 7.25. Noise and muon edge distributions of the output for the IntNet model. Being the final layer a sigmoid activation function, these values are interpreted as probabilities.

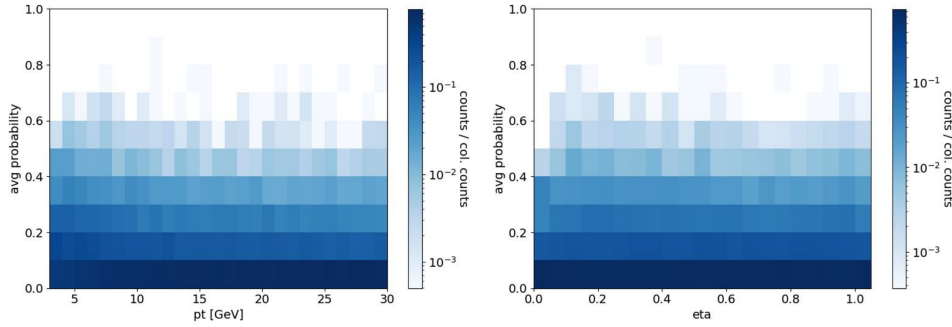


Figure 7.26. Correlation plots between the average edge probability distribution and the transverse momentum, on the left, and pseudorapidity, on the right, of the events containing only noise hits.

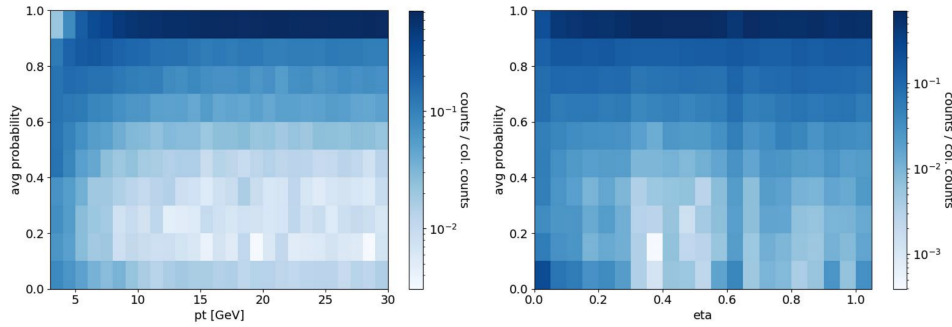


Figure 7.27. Correlation plots between the average edge probability distribution and the transverse momentum, on the left, and pseudorapidity, on the right, of the events containing single muon tracks.

As a result, the IntNet architecture successfully identifies 95.9% of noise edges and 90.6% of the muon edges, as shown in Fig. 7.28. However, the probability distributions for noise and muon events, shown in Fig. 7.25, reveal the presence of misclassified edges. This is indicated by the secondary peaks at low (high) probability values in the blue (green) curve, suggesting some muon (noise) edges are

mistaken for noise (muon). To further investigate this behaviour, correlation plots have been constructed between the average edge probability and the muon's p_T and η (Fig. 7.27 and 7.26). For noise-only events, no significant correlation with kinematic variables is observed. On the contrary, for muon events a slight dependence on transverse momentum is noticeable, especially at low p_T values and in specific problematic detector regions, such as the border of acceptance range and at $\eta = 0.6$. The correlation with low- p_T may be caused by the difficulty if the GNN in reconstructing highly curved trajectories. A case-by-case event analysis (Fig. 7.29) reveals that many misclassified events correspond to pathological muon signatures: hits from muon Bremsstrahlung emission or simulation artifacts, which create hits in unintended locations. However, in some rare cases the model completely fails to identify the muon (Fig. 7.30), although this happens with a very low frequency on the order of 10^{-4} .

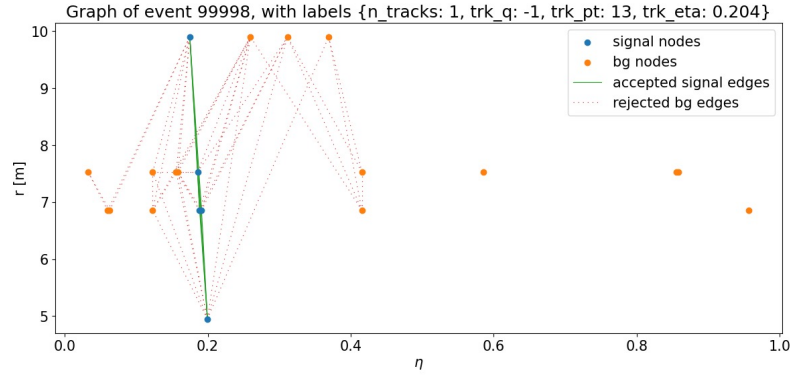


Figure 7.28. Muon track identification with the IntNet model: the muon (noise) nodes are represented in blue (orange), while solid lines are assigned to reconstructed edges and dotted lines to edges that have not been reconstructed.

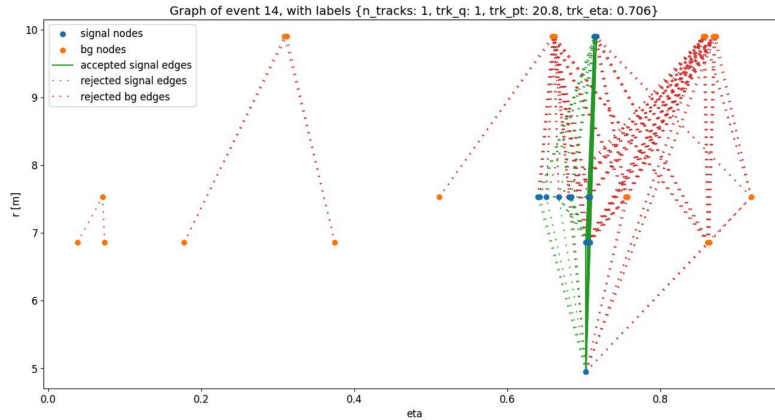


Figure 7.29. Representation of muon misidentification: the muon hits not associated with the final graph are due to Bremsstrahlung.

Overall, the obtained results with the GNN are highly promising and indicate that a graph-based approach is, in principle, a viable and effective solution for muon

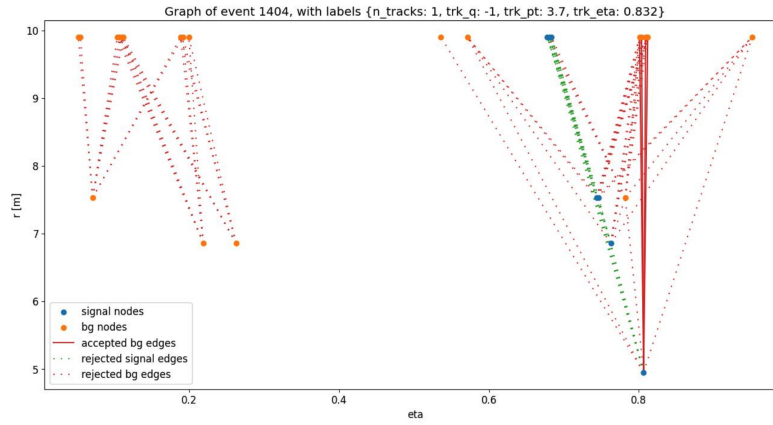


Figure 7.30. Representation of the worst case scenario for the GNN in reconstructing the muon: the muon is present in the image, but the model completely fails in its identification.

reconstruction.

7.10 Perspectives

With this work, we have demonstrated that an ML approach for low-level trigger implementation is feasible, but it comes with significant challenges and considerable efforts. The stringent timing and resource occupancy constraints severely limit the range of possible architectures and force the use of advanced compression techniques. Despite these optimisations, the multi-muon CNN model struggles to efficiently reconstruct transverse momentum, leading to a shift in perspective. A graph representation of the muon tracks is more natural and effective and preliminary studies indicate promising results for the GNN model in distinguish between noise edges and muon edges. The next immediate step is to complete the GNN architecture by integrating the dense blocks to reconstruct the kinematic variables required in the L0 trigger pipeline. Following this, the model must be synthesised and implemented on the FPGA used for the SL. The synthesis process will be conducted using HLS4ML [251], a tool which automatically converts a Python-based ML model into C++ code, which is then translated into HDL for hardware implementation.

A major limitation affecting performance is the dataset itself, which may partially contributed to the observed degradation in efficiency. Notably, the dataset lacks of a full simulation of the RPC BI geometry, which could lead to inaccuracies in training and evaluation. Additionally, it only simulates single muon tracks, whereas many challenges arise in the multi-muon scenario. The method used to generate multi-muon events, by superimposing single muon events, reduces the effective statistics, particularly at low transverse momentum where performance is already highly degraded due to increased trajectory curvature. Therefore, it is essential to generate a larger and more accurate dataset, incorporating a realistic simulation of the BI geometry, a more accurate noise simulation and a dedicated multi-muon dataset. Beyond dataset improvements, a direct performance comparison between ML models and traditional algorithms is crucial for assessing the advantages and disadvantages of each approach. At present, traditional algorithms have not been tested on noise, making any direct comparison unreliable. Once the resources usage of each algorithm on the FPGA is fully evaluated, it would be possible to decide on the most suitable approach for the ATLAS L0 barrel trigger. If resource constraints permit, multiple models could be implemented simultaneously. Given the strict HL-LHC timeline, it is possible that the ML approach will not be fully deployed by the time the first algorithm is implemented on the FPGA. However, we are confident that it will be ready for Run 5, providing a robust and scalable solution for future trigger algorithms.

Chapter 8

Conclusion

The research presented in this thesis explores the use of Machine Learning (ML) in two distinct fields within the ATLAS experiment: an unsupervised Anomaly Detection (AD) approach for Beyond the Standard Model (BSM) resonances searches in fully hadronic final states and the development of ML-based algorithms for the upgrade of the ATLAS Level 0 (L0) Muon Spectrometer trigger. The results of these studies demonstrate the enhanced discriminative power of model-independent frameworks for New Physics (NP) searches and the feasibility of integrating advanced deep learning architectures into real-time event selections.

The model-agnostic analysis is motivated by the lack of direct experimental evidence or strong theoretical guidance on where deviations from the Standard Model might appear. Fully unsupervised approaches, such as AD, are capable of identifying unexpected resonances without assuming any specific NP model. Several ML architectures have been developed and tested first on the LHC Olympics 2020 R&D dataset and then on ATLAS MC simulations. A feature transformation was applied to remove jet mass dependence and ensure invariance under translations and rotations, increasing the model's robustness. Transformers and Graph Neural Networks (GNNs) were explored and demonstrated how low-level variables, as jet constituents, can really be determinant in the separation between resonant signals and non-resonant backgrounds. The Transformer AD architecture was trained and validated exclusively on MC simulated QCD dijet events, ensuring in this way the independence from specific signal models. The model's performance was benchmarked on different signal topologies, including 2prong, 3prong Heavy Vector Triplets and Dark Matter signals. A template fit was performed to estimate preliminary contributions from systematic uncertainties and validate the model detection power over anomalous BSM simulations. The study was extended with an analytical fit over the QCD background, anticipating the data-driven approach that will be adopted when transitioning to ATLAS Run 3 data.

The High-Luminosity LHC (HL-LHC) will introduce higher Pile-Up conditions and significantly increased data rates; to face these issues, major upgrades are scheduled for the ATLAS detector, trigger and data acquisition system. In particular, a new Resistive Plate Chambers (RPC) triplet layer will be added and the trigger logic will be implemented on FPGA-based electronics. Convolutional Neural Networks

(CNNs) and GNN algorithms were explored as alternative to traditional pattern matching approaches. The CNN approach achieved efficient muon reconstruction but struggled with overlapping tracks, leading to the development of a GNN model as a more suitable alternative for the final implementation. FPGA constraints in terms of latency, resource allocation and power consumption were studied. Only through the use of compression techniques, such as Quantization Aware Training and Knowledge Distillation, the ML models can meet the strict timing and occupancy requirements.

Disclaimer

This thesis is the original work of the author. However, it should be noted that AI-based tools were utilized in certain sections for grammar correction and readability improvement. The use of these tools was intended to enhance the clarity and coherence of the text, without altering the originality and integrity of the research presented. All findings, interpretations and conclusions expressed in this thesis are those of the author.

Appendix A

Studies on RPCs timing calibration, detector and trigger efficiencies

When I started my PhD, the ATLAS experiment was concluding LS2 and preparing for the start of the Run 3 data taking, which officially began in July 2022. Before Run 3 could commence, extensive tests and calibrations had to be performed for each sub-detector, to ensure their proper functioning once the first proton-proton collisions were produced. This Appendix provides a brief overview of the operations required for the RPC sub-detector, to which I actively contributed, mainly the maintaining and development of the software providing all the plots for RPC trigger monitoring. Moreover, the software was completely automated few weeks after Run 3 start, to produce the plots as soon as data is available.

A.1 Trigger timing calibration

At the end of March 2022, the ATLAS detector was switched on following LS2 for hardware commissioning. During this period, the first cosmic ray data was collected and stored in the cosmic data stream. Shortly after, the first RPC trigger timing distributions were produced. The events selected for these plots were triggered exclusively by the TRT detector, ensuring that only cosmic rays pointing towards the interaction vertex were included. Most of the recorded events consisted of cosmic rays originating from above the ATLAS detector and passing through the IP, as shown in Fig. A.1a which represents the RPC trigger hits in one of the vertical sectors. However, a smaller fraction of cosmic rays arriving at large incidence angles was also observed, as illustrated by Fig. A.1b, showing one of the horizontal RPC trigger sectors. The timing distribution for the RPC trigger during a cosmic ray run in March 2022 is shown in Fig. A.2a, where time is expressed in ticks (3.125 ns per tick), the time unit used by the trigger logic. The PatFinder algorithm (see Sec. 7.3.1) correctly identifies muon candidates with hits within 30 ± 4 ticks. The global timing distribution exhibits two distinct peaks. The early peak corresponds

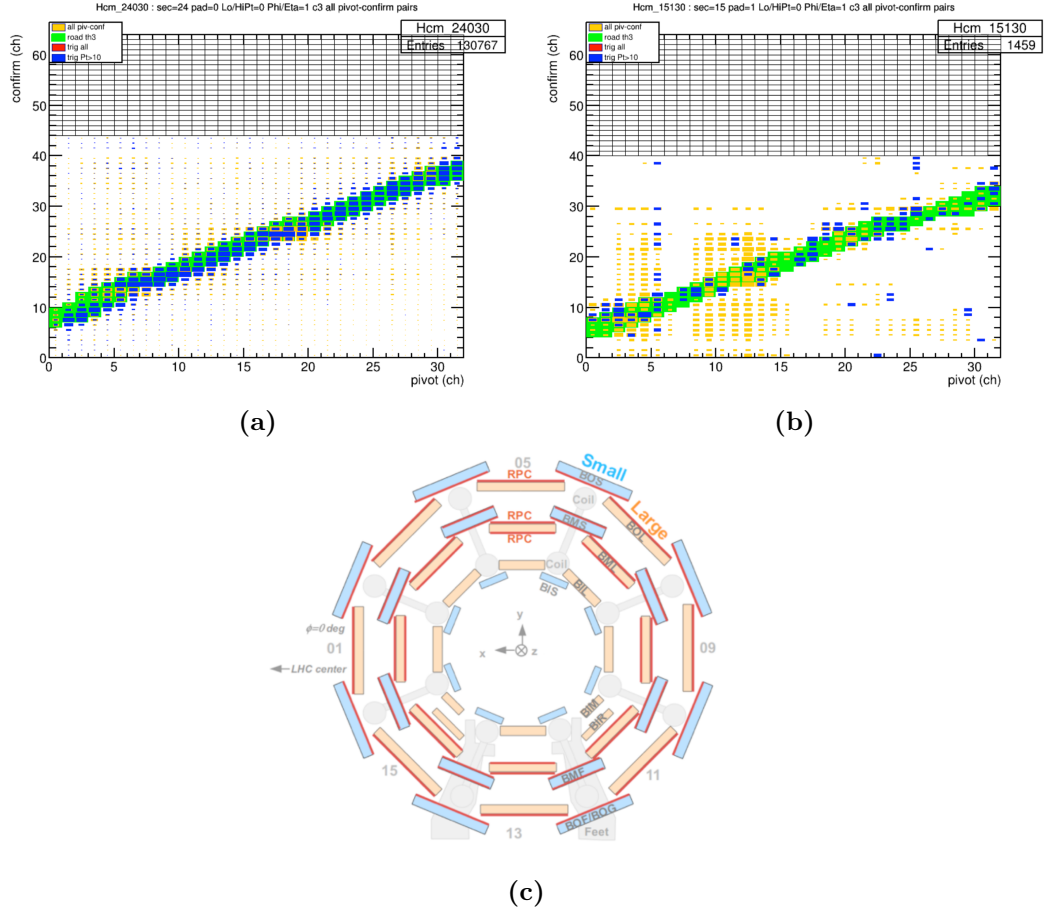


Figure A.1. RPC trigger hits for different sectors during cosmic ray runs, triggered by TRT and RPC sector enumeration (figure from Ref. [252]). Given a hit in one channel (strip) of the pivot layer, on the x -axis, corresponding hits in the confirm layers, on the y -axis, are searched. If the hits (in blue) lie within the green belt, the current PatFinder trigger algorithm selects the event. Fig. A.1a shows the hits from the large sector 13 in side A. Cosmic rays, which primarily come from above, are triggered by the TRT detector and reach the bottom of the MS, simulating muons from p - p collisions. Conversely, Fig. A.1b presents hits from the large sector 9 in side A, which lies in the same plane of the beam axis. In this case, fewer cosmic rays are detected, as it is rare for a cosmic ray to arrive at an incidence angle that allows it to both trigger the TRT and pass in RPC sector 9. Additionally, several hits in this sector fall outside the green belt, suggesting that they likely originate from more frequent vertical cosmic rays that arrived simultaneously with other particles responsible for triggering the TRT.

to cosmic rays passing through the upper sectors (sectors 4 to 6) before triggering the TRT, while the late peak is associated with cosmic rays in the lower sectors (sectors 12 to 14), which follow the opposite trajectory. The timing distribution for the lower sectors (second peak) closely resembles that of muons from proton-proton collisions, which originate from or near the IP, first encountering the TRT and then the MS. Both peaks are relatively broad; this is expected, as cosmic rays with large incidence angles are less frequent but still possible. However, the second peak spans a range

of 40-50 ticks, significantly deviating from the expected 8 ticks window. For this reason, it was decided to wait for the first p - p collisions to perform an accurate timing calibration. When the first proton-proton collisions at 13.6 TeV were recorded at

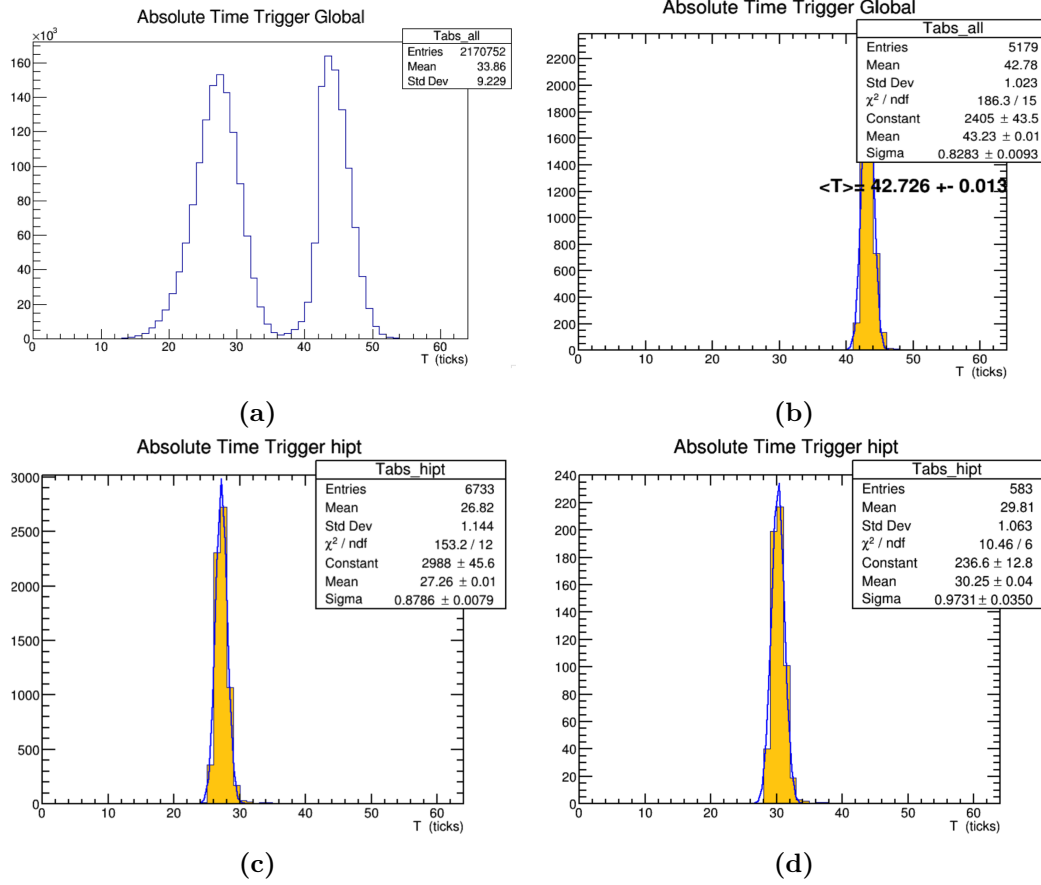


Figure A.2. Global timing distributions for the RPC trigger, with time expressed in ticks (1 tick = 3.125 ns) at different dates. Fig. A.2a shows the distribution for a cosmic ray run taken the 25th March 2022; the typical double peak of cosmic rays is clearly evident and RPCs are not calibrated. The other plots represent the distribution for 13.6 TeV collisions on the 7th July 2022 (Fig. A.2b), 14th July 2022 (Fig. A.2c) and 23rd July 2022 (Fig. A.2d). The timing calibration was progressively performed and within two weeks after the start of Run 3 RPCs were perfectly calibrated.

the beginning of July 2022, the trigger timing distribution appeared as a narrow Gaussian centred around 43 ticks (Fig. A.2b). A preliminary timing calibration was immediately performed, shifting the peak closer to the expected value. However, by mid-July, although the distribution had improved, approximately 8% of the hits still fell outside the expected window (Fig. A.2c). To fully correct the timing, three additional ticks were introduced in the trigger pipeline, bringing the trigger timing distribution to the expected 30 tick centre, as shown in Fig. A.2d.

The global RPC trigger timing was now correctly centred; however, certain trigger pads remained out of the expected bunch crossing window, as shown by the red and yellow boxes in Fig. A.3a. At that time, the overall L1 muon trigger efficiency

was between 30-40%, significantly lower than the Run 2 reference values of 70-80%. These inefficiencies were mitigated through a combination of targeted interventions, described in the following Section, which allowed for the recovery of most of the issues by October 2022 (Fig. A.3b).

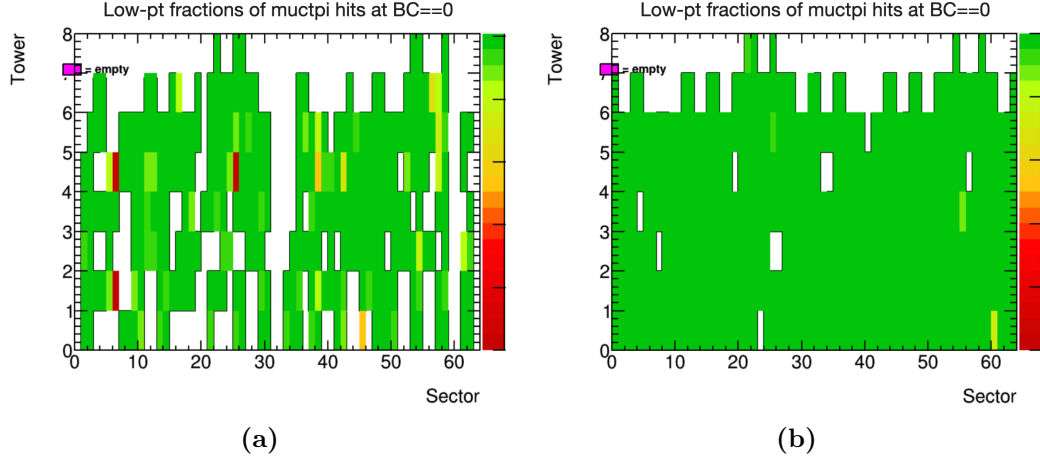


Figure A.3. Fraction of hits arriving at the correct bunch crossing by the end of July 2022 (Fig. A.3a) and at the beginning of October 2022 (Fig. A.3b). The plots display the 64 semi-sectors (32 per side) in the ϕ segmentation. Each sector consists of 6-7 towers in η , except for those in the ATLAS feet region which include an additional eighth tower originating from the BIS78. White boxes indicate towers where no hits were recorded, typically due to detector issues such as HV failures or RPC gas leaks.

A.2 L1 muon barrel trigger efficiencies

At the beginning of Run 3, several RPC towers were non-functional due to High Voltage (HV) issues and RPC gas leaks, while others exhibited low efficiency due to timing misalignments with the correct bunch crossing. The L1 muon barrel trigger efficiency was measured using reconstructed muons in events with triggers “orthogonal” to the muon trigger items. Orthogonal triggers are all the items that do not introduce any bias in the muon reconstruction, mainly jet or electron triggers. The trigger efficiency showed an overall performance between 30 and 40%, significantly lower than the 70-80% efficiency observed during Run 2. Although recoveries from HV failures and resynchronisation of problematic sectors were performed, they only partially mitigated the efficiency losses compared to Run 2. To address this issue, a gradual relaxation of the trigger majorities was studied and progressively implemented. The PatFinder algorithm identifies low transverse momentum muons by applying p_T -dependent coincidence windows, requiring a 3/4 hit coincidence between RPC layers in the two innermost stations, BM1 and BM2. However, due to detector ageing and inefficiencies, this requirement was too stringent in several sectors, leading to poor overall trigger efficiency. To improve efficiency, we initially relaxed the majority requirement by reducing it to a 1/2 coincidence in BM1, logically combined with a 1/2 coincidence in BM2. A potential side effect of this approach was an increase in the trigger rate, but the magnitude of this effect was

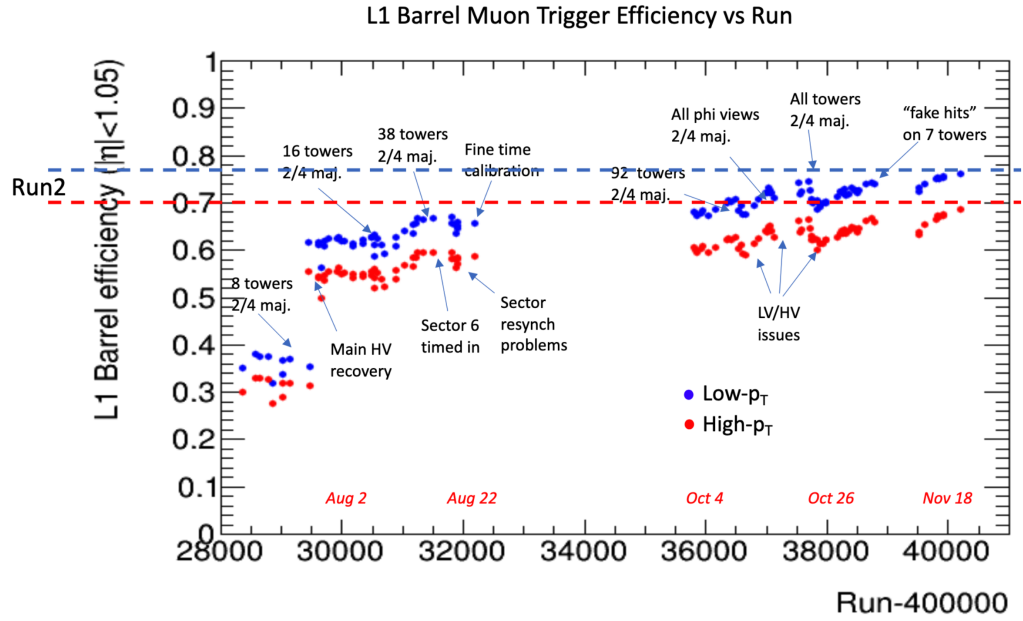


Figure A.4. L1 muon barrel efficiency as a function of the Run Number during 2022 data taking. The blue points represent the two-station low p_T trigger efficiency, used for multi muon L1 trigger items, while the red points correspond to the three-station high p_T trigger efficiency employed for single muon triggers. The efficiency timeline reflects the effect of different detector issues, interventions on hardware and changes in the trigger logic the impact of various detector issues, hardware interventions and modifications to the trigger logic.

unknown a priori. Through continuous monitoring of the trigger rate during the progressive application of the relaxed majorities to new sectors, we observed that the rate increase was not significant. Consequently, at the beginning of October 2022, the relaxation was applied to all ϕ towers and, by the end of October, to all ϕ and η towers. However, the relaxation of the majority requirement is only effective if at least one of the two RPC gas gaps is operational; otherwise, an alternative solution is necessary. From November 2022, problematic RPC planes were masked by introducing artificial “fake hits”. This procedure was applied selectively to the affected towers, as it requires continuous monitoring of the trigger rates. Thanks to these interventions, by November 2022, the L1 muon trigger efficiency successfully reached the expected 70-80% observed in Run 2 (Fig. A.4) and has maintained this performance throughout 2023 and 2024.

Appendix B

Details on the jet transformation

The transformation applied to each jet and its constituents to make the ML algorithm robust is inspired from the procedure described in [182]. There are few differences between the implementation in [182] and the actual implementation used in this thesis.

B.1 The original procedure

Given the i^{th} constituent of a jet of four momentum p_i^μ , we can define the jet four-momentum P_J^μ , the jet mass m_J^2 and the jet energy E_J as:

$$P_J^\mu = \sum_{i=0}^{N_J} p_i^\mu, \quad m_J^2 \equiv (P_J^0)^2 - (\vec{P}_J)^2, \quad E_J \equiv P_J^0 \quad (\text{B.1})$$

First, the mass rescaling is applied to all the jet and constituents' four-momenta, so that the final jet mass takes a fixed m_0 value. This corresponds to:

$$P_J^\mu \longrightarrow P_J'^\mu = \frac{m_0}{m_J} P_J^\mu, \quad p_i^\mu \longrightarrow p_i'^\mu = \frac{m_0}{m_J} p_i^\mu \quad (\text{B.2})$$

Then, a Lorentz boost is applied so that the final jet energy becomes fixed to a predetermined E_0 value. The direction in which apply the boost is determined by the energy the jet has at this point E_J' :

- if $E_0 < E_J'$: the boost is performed along the direction of the jet three-momentum;
- if $E_0 > E_J'$: the boost is performed along the direction *opposite* to the jet three-momentum.

The Lorentz factor γ of the boost can be easily obtained by solving the equation for which the energy after the boost E_J'' becomes equal to E_0 :

$$\gamma E_J' - \beta \gamma P_J' = E_J'' \equiv E_0 \quad \text{where } \beta^2 = 1 - \frac{1}{\gamma^2} \quad (\text{B.3})$$

This gives the following quadratic equation in γ :

$$\begin{aligned} m_0^2 \gamma^2 - 2[E'_J E_0] \gamma + [E_0^2 + (P'_J)^2] \\ \implies \gamma = \frac{1}{m_0^2} (E'_J E_0 - P'_J P_0) \end{aligned} \quad (\text{B.4})$$

where $P_0^2 = E_0^2 - m_0^2 \equiv (P''_J)^2$ and it comes from taking the smaller solution of the quadratic equation.

Lastly, a new set of basis vector $\{\hat{e}_1, \hat{e}_2, \hat{e}_3\}$ in the transverse plane is identified by the Gram-Schmidt orthonormal procedure and the residual rotation symmetry is removed from the constituents. In the new boosted reference frame the constituents can be ordered in descending order in energy, such that $p_1^0 \geq p_2^0 \geq \dots \geq p_N^0$. The new basis will be:

$$\begin{aligned} \hat{e}_1 &\equiv \frac{\vec{P}_J''}{|\vec{P}_J''|}, \\ \hat{e}_2 &\equiv \frac{\vec{p}_1 - (\vec{p}_1 \cdot \hat{e}_1) \hat{e}_1}{|\vec{p}_1 - (\vec{p}_1 \cdot \hat{e}_1) \hat{e}_1|}, \\ \hat{e}_3 &\equiv \frac{\vec{p}_2 - (\vec{p}_2 \cdot \hat{e}_1) \hat{e}_1 - (\vec{p}_2 \cdot \hat{e}_2) \hat{e}_2}{|\vec{p}_1 - (\vec{p}_1 \cdot \hat{e}_1) \hat{e}_1| - (\vec{p}_2 \cdot \hat{e}_2) \hat{e}_2} \end{aligned} \quad (\text{B.5})$$

B.2 The implemented procedure

The procedure implemented in the analysis code shown in Chap. 5 and 6 for applying the transformation is slightly different from that described just above, but the result is the same. For the implementation in Python, see Fig. B.1; the Scikit-HEP LorentzVector and Vector3D libraries [253] allow a built-in handling of four-vectors, three-vectors and Lorentz boosts.

Let's start by defining the jet initial four-momentum with Minkowskian coordinates, as well as in the (p_T, η, ϕ, m) basis:

$$P_J^\mu = \begin{cases} (P_x, P_y, P_z, E) \\ (P_T, \eta, \phi, m) \end{cases} \quad (\text{B.6})$$

First, a rotation of angle $-\phi$ is applied, followed by a boost along the z -direction and with $\beta = \frac{P_z}{E}$, so that the jet axis lies in $(\eta = 0, \phi = 0)$ and, more generally:

$$P_J^\mu \longrightarrow P_J'^\mu = \begin{cases} (P'_x = P_T, P'_y = 0, P'_z = 0, E' = \sqrt{E^2 - P_z^2}) \\ (P'_T = P_T, \eta' = 0, \phi' = 0, m' = m) \end{cases} \quad (\text{B.7})$$

As a consequence of the boost, the y and z momentum components are vanished, the x component has taken all the P_T value and the energy has changed accordingly. Then, the four-momentum of the jet is rescaled by the m_0 fixed mass, as:

$$P_J'^\mu \longrightarrow \tilde{P}_J^\mu = \frac{m_0}{m} P_J'^\mu = \begin{cases} (\tilde{P}_x = \hat{m} P_T, \tilde{P}_y = 0, \tilde{P}_z = 0, \tilde{E} = \hat{m} \sqrt{E^2 - P_z^2}) \\ (\tilde{P}_T = \hat{m} P_T, \tilde{\eta} = 0, \tilde{\phi} = 0, \tilde{m} = m_0) \end{cases} \quad (\text{B.8})$$

```

1  import numpy as np
2  from skhep.math.vectors import LorentzVector, Vector3D
3
4  def transformation(jet, const, m_zero=0.25, e_zero=1.):
5      bv = jet.boostvector
6      bv.x = 0
7      bv.y = 0
8
9      jet_transf = jet.rotatez(-jet.phi())
10
11     jet_transf = jet_transf.boost(bv)
12
13     m_rescale = m_zero/jet_transf.m if jet_transf.m else 1
14     jet_transf = m_rescale * jet_transf
15
16     A = (np.sqrt((np.abs(np.square(zero_E) - np.square(jet_transf.m)))/
17                 (np.abs(np.square(jet_transf.e) - np.square(jet_transf.m)))))
18     bv2 = A * jet_transf.px / e_zero
19
20     beta = (jet_transf.e - (jet_transf.px/bv2))/(jet_transf.p - (jet_transf.e/bv2))
21     jet_transf = jet_transf.boost(beta, 0, 0)
22
23     alpha = None
24     const_transf = []
25
26     for i in range(len(const)):
27         const[i] = const[i].rotatez(-jet.phi)
28         const[i] = const[i].boost(bv)
29         const[i] = m_rescale * const[i]
30         if alpha is None:
31             alpha = np.arctan2(const[i].phi(), const[i].eta)
32             const[i] = const[i].rotatex(alpha - np.pi/2)
33         const_transf.append(const[i])
34
35     return jet_transf, const_transf

```

Figure B.1. Implementation of the transformation procedure in Python, using the Scikit-HEP LorentzVector and Vector3D packages in [253].

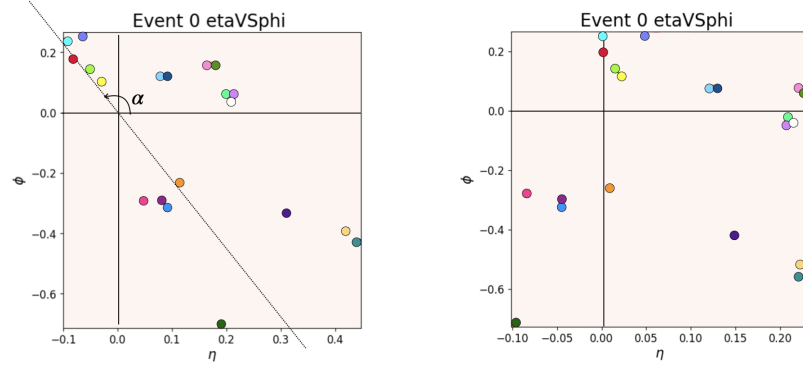


Figure B.2. Constituents of a jet in the η - ϕ plane before (on the left) and after (on the right) last rotation around the x axis. Note that after the rotation the relative distances of the constituents do change in the η - ϕ plane.

where $\hat{m} = \frac{m_0}{m}$ is the rescaling mass. Now, the jet undergoes a final boost in the x direction, that is the only spatial component with non-null values. The β'' of the Lorentz boost that transforms the energy of the jet into E_0 is given by:

$$\beta'' = \frac{P_0 \tilde{E} - \tilde{P}_x E_0}{P_0 \tilde{P}_x - E_0 \tilde{E}}, \quad \text{where } P_0 \equiv \sqrt{E_0^2 - m_0^2} \quad (\text{B.9})$$

It is easy to verify that Eq. B.9 corresponds to Eq. B.4, in fact:

$$\gamma'' = \frac{1}{\sqrt{1 - (\beta'')^2}} = \frac{|P_0 \tilde{P}_x - E_0 \tilde{E}|}{m_0^2} \quad (\text{B.10})$$

Therefore, after the transformation the jet four-momentum is given by:

$$P_J''^\mu = \begin{cases} (P_x'' = \gamma'' \tilde{P}_x - \beta'' \gamma'' \tilde{E}, P_y'' = 0, P_z'' = 0, E'' = E_0) \\ (P_T'' = P_x'', \eta'' = 0, \phi'' = 0, m'' = m_0) \end{cases} \quad (\text{B.11})$$

The same transformation is then applied to the constituents of the jet, with the addition of a further rotation to remove the remaining symmetry of the constituents position around jet axis. Assuming that the constituents in the jet are ordered in decreasing p_T values, the first hardest constituent is rotated of angle $-\pi/2$ around the x axis, so that its η and ϕ coordinates become: $(\eta_1 = 0, \phi_1 > 0)$. All the following constituents, instead, will be rotated of angle $\alpha - \pi/2$ around the x axis, where α is the angle taken by the first constituent in the η - ϕ plane. These last steps are analogous to the Gram-Schmidt orthogonalisation, but here we have decided that the coordinates of interest are η and ϕ . These coordinates are not the usual Cartesian ones and it would be difficult to implement a rotation in η and ϕ . Therefore, by noting that the x axis has $\eta_x = 0$ and $\phi_x = 0$ by definition and therefore is orthogonal to the η - ϕ plane, it is possible to perform the desired transformation by rotating the constituents around the x axis. Note that in the η - ϕ plane the relative distances of the constituents are not left the same. In Fig. B.2 the effects of the rotation on the constituents of an example jet are shown.

List of Acronyms

ACC average clustering coefficient

AD Anomaly Detection

Adam Adaptive Moment Estimation

AE AutoEncoders

AI Artificial Intelligence

ALICE A Large Ion Collider Experiment

AMVF Adaptive Multi-Vertex Finder

AOD Analysis Object Data

AS Anomaly Score

ASIC Application-Specific Integrated Circuit

ASPL average shortest path length

ATLAS A Toroidal LHC ApparatuS

AUC Area Under the Curve

BB Black Boxes

BCID Bunch Crossing IDentifier

BDT Boosted Decision Trees

BEH Brout-Englert-Higgs

BI Barrel Inner

BM Barrel Middle

BO Barrel Outer

BSM Beyond the Standard Model

CATHODE Classifying Anomalies THrough Outer Density Estimation

CB	Crystal Ball
CERN	European Organization for Nuclear Research
CL	confidence level
CM	Combined Muons
CMB	Cosmic Microwave Background
CMS	Compact Muon Solenoid
CNN	Convolutional Neural Networks
CP	conjugation-parity
CR	control region
CSC	Cathode Strip Chambers
CSSK	Constituent Subtraction and SoftKiller
CT	Calorimeter-Tagged Muons
CTP	Central Trigger Processor
CWoLa	Classification WithOut Labels
DAOD	Derived AOD
DCT	Data Collector and Transmitter
Deep SVDD	Deep Support Vector Data Description
DGL	Deep Graph Library
DJ	Dark Jets
DL	Deep Learning
DM	Dark Matter
DNN	Deep Neural Networks
ECal	Electromagnetic Calorimeter
ECF	Energy Correlation Functions
EGAT	Edge Graph ATtention network
EM	electromagnetic
EMEC	Electromagnetic End-cap Calorimeter
EW	Electroweak

FCal Forward Calorimeter

FE Front-End

FPGA Field Programmable Gate Arrays

FSR Final State Radiation

GAT Graph ATtention network

GD Gradient Descent

GELU Gaussian Error Linear Unit

GIN Graph Isomorphism Network

GINE Graph Isomorphism Network with Edge features

GLocalKD Graph Local Knowledge Distillation

GNN Graph Neural Networks

GPU Graphics Processing Unit

GRL good run list

HCal Hadronic Calorimeter

HDL Hardware Description Language

HEC Hadronic End-cap Calorimeter

HEP High Energy Physics

HL-LHC High-Luminosity LHC

HLT High Level Trigger

HV High Voltage

HVT Heavy Vector Triplets

IBL Insertable B-Layer

ID Inner Detector

ILSVRC ImageNet Large Scale Visual Recognition Challenge

IntNet Interaction Network

IP Interaction Point

IR Interaction Regions

ISR Initial State Radiation

ITk Inner TracKer

JER Jet Energy Response

JES Jet Energy Scale

KD Knowledge Distillation

KK Kaluza-Klein

L0 Level 0

L1 Level 1

L1Topo Level-1 Topological Processor

LCW Local Cell Weighted

LEE Look-Elsewhere Effect

LEIR Low Energy Ion Ring

LEP Large Electron-Positron

LHC Large Hadron Collider

LHCb Large Hadron Collider beauty

Linac3 LINear ACcelerator-3

Linac4 LINear ACcelerator-4

LLM Large Language Models

LLP Long Lived Particles

LR Learning Rate

LRT Large Radius Tracking

LS Long Shutdown

MC Monte Carlo

MDT Monitored Drift Tubes

MDT-TP MDT Trigger Processor

ME Extrapolated Muons

MET Missing Transverse Energy

MHA MultiHead Attention

MIP Minimum Ionising Particles

ML	Machine Learning
MLP	Multi-Layer Perceptron
MM	MicroMegas
mMDT	modified Mass Drop Tagger
MS	Muon Spectrometer
MSE	Mean Squared Error
MUCTPI	Muon Central Trigger Processor Interface
NN	Neural Network
NP	New Physics
NPLM	New Physics Learning Machine
NSW	New Small Wheel
OCGTL	One Class Graph Transformation Learning
OR	Overlap Removal
PDF	Parton Distribution Functions
PFlow	Particle Flow
PMT	PhotoMultiplier Tube
PReLU	Parametric Rectified Linear Unit
PS	Proton Synchrotron
PSB	Proton Synchrotron Booster
PU	Pile-Up
QAT	Quantization Aware Training
QCD	Quantum Chromodynamics
QED	Quantum Electrodynamics
QFT	Quantum Field Theory
QM	Quantum Mechanics
QUAK	QUasi Anomalous Knowledge
R&D	research and development

RDO	Raw Data Object
ReLU	Rectified Linear Unit
RF	Radio Frequency
RGE	Renormalisation Group Equation
RMS	Root Mean Squared
RNN	Recurrent Neural Networks
ROC	Receiver Operating Characteristic
ROD	ReadOut Drivers
RoI	Regions of Interest
ROS	ReadOut System
RPC	Resistive Plate Chambers
SAGE	SAMple and aggreGatE
SALAD	Simulation Assisted Likelihood-free Anomaly Detection
SB	sideband
SCT	SemiConductor Tracker
SD	Soft Drop
SGD	Stochastic Gradient Descent
SL	Sector Logic
SLR	Super Logic Region
SM	Standard Model
sMDT	small MDT
SoC	System on Chip
SPS	Super Proton Synchrotron
SR	signal region
ST	Segment-Tagged Muons
sTGC	small-strip Thin-Gap Chambers
TAE1	Transformer + AutoEncoder architecture 1
TAE2	Transformer + AutoEncoder architecture 2

TCC	Track Calo-Clusters
TDAQ	Trigger and Data Acquisition
TDC	Time-to-Digital Converter
TGC	Thin-Gap Chambers
TLA	Trigger-Level Analysis
TR	Transition Radiation
TRT	Transition Radiation Tracker
UFO	Unified Flow Objects
USA	Underground Service ATLAS
VAE	Variational AutoEncoders
VEV	vacuum expectation value
VGG	Visual Geometry Group
VRNN	Variational Recurrent Neural Network
WMAP	Wilkinson Microwave Anisotropy Probe
WP	Working Point
WTA	winner-takes-all
XAI	eXplainable Artificial Intelligence
YETS	Year-End Technical Stop

Bibliography

- [1] G. Russo. “Search for Anomalies in hadronic final states with GNNs in ATLAS”. In: *Il Nuovo Cimento* 47.3 (2024).
- [2] G. Aad et al. “Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC”. In: *Physics Letters B* 716.1 (Sept. 2012), pp. 1–29. ISSN: 0370-2693. DOI: [10.1016/j.physletb.2012.08.020](https://doi.org/10.1016/j.physletb.2012.08.020). URL: <http://dx.doi.org/10.1016/j.physletb.2012.08.020>.
- [3] S. Chatrchyan et al. “Observation of a New Boson at a Mass of 125 GeV with the CMS Experiment at the LHC”. In: *Phys. Lett. B* 716 (2012), pp. 30–61. DOI: [10.1016/j.physletb.2012.08.021](https://doi.org/10.1016/j.physletb.2012.08.021). arXiv: [1207.7235 \[hep-ex\]](https://arxiv.org/abs/1207.7235).
- [4] M. E. Peskin and D. V. Schroeder. “An Introduction to quantum field theory”. In: (1995). DOI: [10.1201/9780429503559](https://doi.org/10.1201/9780429503559).
- [5] S. Weinberg. “The Quantum theory of fields. Vol. 1: Foundations”. In: (June 2005). DOI: [10.1017/CB09781139644167](https://doi.org/10.1017/CB09781139644167).
- [6] Wikipedia The Free Encyclopedia. “Standard Model”. In: (2024). URL: https://en.wikipedia.org/wiki/Standard_Model.
- [7] A. Pich. “The Standard Model of Electroweak Interactions”. In: (2012). eprint: [1201.0537](https://arxiv.org/abs/1201.0537). URL: <https://arxiv.org/abs/1201.0537>.
- [8] E. Noether. “Invariant variation problems”. In: *Transport Theory and Statistical Physics* 1.3 (Jan. 1971), pp. 186–207. ISSN: 1532-2424. DOI: [10.1080/00411457108231446](https://doi.org/10.1080/00411457108231446). URL: <http://dx.doi.org/10.1080/00411457108231446>.
- [9] J. Huston, K. Rabbertz, and G. Zanderighi. “Quantum Chromodynamics”. In: (2023). arXiv: [2312.14015 \[hep-ph\]](https://arxiv.org/abs/2312.14015). URL: <https://arxiv.org/abs/2312.14015>.
- [10] W. Greiner, S. Schramm, and E. Stein. “Quantum chromodynamics”. In: (2002).
- [11] M. Gell-Mann. “Symmetries of baryons and mesons”. In: *Phys. Rev.* 125 (1962), pp. 1067–1084. DOI: [10.1103/PhysRev.125.1067](https://doi.org/10.1103/PhysRev.125.1067).
- [12] A. Pich. “The Standard Model of Electroweak Interactions”. In: (2012). arXiv: [1201.0537 \[hep-ph\]](https://arxiv.org/abs/1201.0537). URL: <https://arxiv.org/abs/1201.0537>.
- [13] S. Weinberg. “A Model of Leptons”. In: *Phys. Rev. Lett.* 19 (1967), pp. 1264–1266. DOI: [10.1103/PhysRevLett.19.1264](https://doi.org/10.1103/PhysRevLett.19.1264).

- [14] A. Salam and J. C. Ward. “Weak and electromagnetic interactions”. In: *Nuovo Cim.* 11 (1959), pp. 568–577. DOI: [10.1007/BF02726525](https://doi.org/10.1007/BF02726525).
- [15] N. Cabibbo. “Unitary Symmetry and Leptonic Decays”. In: *Phys. Rev. Lett.* 10 (1963), pp. 531–533. DOI: [10.1103/PhysRevLett.10.531](https://doi.org/10.1103/PhysRevLett.10.531).
- [16] M. Kobayashi and T. Maskawa. “CP Violation in the Renormalizable Theory of Weak Interaction”. In: *Prog. Theor. Phys.* 49 (1973), pp. 652–657. DOI: [10.1143/PTP.49.652](https://doi.org/10.1143/PTP.49.652).
- [17] A. Ceccucci, Z. Ligeti, and Y. Sakai. “The CKM quark-mixing matrix”. In: (2008).
- [18] P. W. Higgs. “Broken Symmetries and the Masses of Gauge Bosons”. In: *Phys. Rev. Lett.* 13 (16 Oct. 1964), pp. 508–509. DOI: [10.1103/PhysRevLett.13.508](https://doi.org/10.1103/PhysRevLett.13.508). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.13.508>.
- [19] R. Brout. “Ferromagnetic spin waves”. In: *Physica* 26 (1960), S215–S217. ISSN: 0031-8914. DOI: [https://doi.org/10.1016/0031-8914\(60\)90225-1](https://doi.org/10.1016/0031-8914(60)90225-1). URL: <https://www.sciencedirect.com/science/article/pii/0031891460902251>.
- [20] F. Englert and R. Brout. “Broken Symmetry and the Mass of Gauge Vector Mesons”. In: *Phys. Rev. Lett.* 13 (9 Aug. 1964), pp. 321–323. DOI: [10.1103/PhysRevLett.13.321](https://doi.org/10.1103/PhysRevLett.13.321). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.13.321>.
- [21] A. Abokhalil. “The Higgs Mechanism and Higgs Boson: Unveiling the Symmetry of the Universe”. In: (2023). arXiv: [2306.01019](https://arxiv.org/abs/2306.01019) [hep-ph]. URL: <https://arxiv.org/abs/2306.01019>.
- [22] S. Catani. “Aspects of QCD, from the Tevatron to the LHC”. In: *1st Les Houches Workshop on Physics at TeV Colliders*. May 2000, pp. 4–33. arXiv: [hep-ph/0005233](https://arxiv.org/abs/hep-ph/0005233).
- [23] J. C. Collins, D. E. Soper, and G. F. Sterman. “Factorization of Hard Processes in QCD”. In: *Adv. Ser. Direct. High Energy Phys.* 5 (1989), pp. 1–91. DOI: [10.1142/9789814503266_0001](https://doi.org/10.1142/9789814503266_0001). arXiv: [hep-ph/0409313](https://arxiv.org/abs/hep-ph/0409313).
- [24] G. Altarelli and G. Parisi. “Asymptotic Freedom in Parton Language”. In: *Nucl. Phys. B* 126 (1977), pp. 298–318. DOI: [10.1016/0550-3213\(77\)90384-4](https://doi.org/10.1016/0550-3213(77)90384-4).
- [25] T. Sjöstrand et al. “An introduction to PYTHIA 8.2”. In: *Computer Physics Communications* 191 (June 2015), pp. 159–177. ISSN: 0010-4655. DOI: [10.1016/j.cpc.2015.01.024](https://doi.org/10.1016/j.cpc.2015.01.024). URL: <http://dx.doi.org/10.1016/j.cpc.2015.01.024>.
- [26] J. Bellm et al. “Herwig 7.1 Release Note”. In: (2017). arXiv: [1705.06919](https://arxiv.org/abs/1705.06919) [hep-ph]. URL: <https://arxiv.org/abs/1705.06919>.
- [27] S. Alioli et al. “A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX”. In: *Journal of High Energy Physics* 2010.6 (June 2010). ISSN: 1029-8479. DOI: [10.1007/jhep06\(2010\)043](https://doi.org/10.1007/jhep06(2010)043). URL: [http://dx.doi.org/10.1007/JHEP06\(2010\)043](http://dx.doi.org/10.1007/JHEP06(2010)043).

- [28] R. Pöttgen. “Proton-Proton Collisions”. In: *Springer International Publishing* (2016), pp. 45–60. DOI: [10.1007/978-3-319-41045-6_4](https://doi.org/10.1007/978-3-319-41045-6_4).
- [29] B. A. Robson. “The Matter-Antimatter Asymmetry Problem”. In: *Understanding Gravity* (2018). URL: <https://api.semanticscholar.org/CorpusID:53992365>.
- [30] A. D. Sakharov. “Violation of CP Invariance, C asymmetry, and baryon asymmetry of the universe”. In: *Pisma Zh. Eksp. Teor. Fiz.* 5 (1967), pp. 32–35. DOI: [10.1070/PU1991v034n05ABEH002497](https://doi.org/10.1070/PU1991v034n05ABEH002497).
- [31] M. Cirelli, A. Strumia, and J. Zupan. “Dark Matter”. In: (2024). arXiv: [2406.01705](https://arxiv.org/abs/2406.01705) [hep-ph]. URL: <https://arxiv.org/abs/2406.01705>.
- [32] E. Hubble and M. L. Humason. “The Velocity-Distance Relation among Extra-Galactic Nebulae”. In: *Astrophys. J.* 74 (1931), pp. 43–80. DOI: [10.1086/143323](https://doi.org/10.1086/143323).
- [33] H. Andernach (translator) and F. Zwicky (author). “English and Spanish Translation of Zwicky’s (1933) The Redshift of Extragalactic Nebulae”. In: (2017). arXiv: [1711.01693](https://arxiv.org/abs/1711.01693) [astro-ph.IM]. URL: <https://arxiv.org/abs/1711.01693>.
- [34] M. Schwarzschild. “Mass distribution and mass-luminosity ratio in galaxies”. In: *Astronomical Journal, Vol. 59, p. 273 (1954)* 59 (1954), p. 273.
- [35] V. C. Rubin and W. K. Jr Ford. “Rotation of the Andromeda Nebula from a Spectroscopic Survey of Emission Regions”. In: *Astrophys. J.* 159 (1970), pp. 379–403. DOI: [10.1086/150317](https://doi.org/10.1086/150317).
- [36] P. Salucci et al. “The universal rotation curve of spiral galaxies–II. The dark matter distribution out to the virial radius”. In: *Monthly Notices of the Royal Astronomical Society* 378.1 (2007), pp. 41–47.
- [37] A. Einstein. “Lens-like action of a star by the deviation of light in the gravitational field”. In: *Science* 84.2188 (1936), pp. 506–507.
- [38] T. G. Brainerd, R. D. Blandford, and I. Smail. “Weak gravitational lensing by galaxies”. In: *Astrophysical Journal v. 466, p. 623* 466 (1996), p. 623.
- [39] D. Clowe et al. “A direct empirical proof of the existence of dark matter”. In: *The Astrophysical Journal* 648.2 (2006), p. L109.
- [40] C. L. Bennett et al. “Nine-year Wilkinson Microwave Anisotropy Probe (WMAP) observations: final maps and results”. In: *The Astrophysical Journal Supplement Series* 208.2 (Sept. 2013), p. 20. ISSN: 1538-4365. DOI: [10.1088/0067-0049/208/2/20](https://doi.org/10.1088/0067-0049/208/2/20). URL: <http://dx.doi.org/10.1088/0067-0049/208/2/20>.
- [41] N. Aghanim et al. “Planck2018 results: V. CMB power spectra and likelihoods”. In: *Astronomy and Astrophysics* 641 (Sept. 2020), A5. ISSN: 1432-0746. DOI: [10.1051/0004-6361/201936386](https://doi.org/10.1051/0004-6361/201936386). URL: <http://dx.doi.org/10.1051/0004-6361/201936386>.

- [42] M. Milgrom. “A modification of the Newtonian dynamics as a possible alternative to the hidden mass hypothesis”. In: *Astrophysical Journal, Part 1 (ISSN 0004-637X)*, vol. 270, July 15, 1983, p. 365–370. Research supported by the US-Israel Binational Science Foundation. 270 (1983), pp. 365–370.
- [43] T. Clifton et al. “Modified gravity and cosmology”. In: *Physics reports* 513.1–3 (2012), pp. 1–189.
- [44] Y. B. Zel’Dovich and I. D. Novikov. “The hypothesis of cores retarded during expansion and the hot cosmological model”. In: *Astronomicheskii Zhurnal* 43 (1966), p. 758.
- [45] B. Paczynski. “Gravitational microlensing by the galactic halo”. In: *Astrophysical Journal, Part 1 (ISSN 0004-637X)*, vol. 304, May 1, 1986, p. 1–5. 304 (1986), pp. 1–5.
- [46] A. Boyarsky, O. Ruchayskiy, and M. Shaposhnikov. “The role of sterile neutrinos in cosmology and astrophysics”. In: *Annual Review of Nuclear and Particle Science* 59.1 (2009), pp. 191–214.
- [47] J. Preskill, M. B. Wise, and F. Wilczek. “Cosmology of the invisible axion”. In: *Physics Letters B* 120.1–3 (1983), pp. 127–132.
- [48] G. Steigman and M. S. Turner. “Cosmological constraints on the properties of weakly interacting massive particles”. In: *Nuclear Physics B* 253 (1985), pp. 375–386.
- [49] E. Witten. “Dynamical breaking of supersymmetry”. In: *Nuclear Physics B* 188.3 (1981), pp. 513–554.
- [50] J. Ellis et al. “Supersymmetric relics from the big bang”. In: *Nuclear Physics B* 238.2 (1984), pp. 453–476.
- [51] B. Pontecorvo. “Mesonium and anti-mesonium”. In: *Sov. Phys. JETP* 6 (1957), p. 429.
- [52] B. Pontecorvo. “Neutrino experiments and the problem of conservation of leptonic charge”. In: *Sov. Phys. JETP* 26.984–988 (1968), p. 165.
- [53] Z. Maki, M. Nakagawa, and S. Sakata. “Remarks on the unified model of elementary particles”. In: *Progress of Theoretical Physics* 28.5 (1962), pp. 870–880.
- [54] C. Abel et al. “Measurement of the permanent electric dipole moment of the neutron”. In: *Physical Review Letters* 124.8 (2020), p. 081803.
- [55] R. D. Peccei. “The Strong CP problem and axions”. In: *Axions: Theory, Cosmology, and Experimental Searches*. Springer, 2008, pp. 3–17.
- [56] A. Crivellin and B. Mellado. “Anomalies in particle physics and their implications for physics beyond the standard model”. In: *Nature Reviews Physics* 6.5 (2024), pp. 294–309.
- [57] N. Zipper. “Testing a Neural Network for Anomaly Detection in the CMS Global Trigger Test Crate during Run 3”. In: (2023). arXiv: [2312.10009](https://arxiv.org/abs/2312.10009) [hep-ex]. URL: <https://arxiv.org/abs/2312.10009>.

- [58] D. Alves et al. “Simplified models for LHC new physics searches”. In: *Journal of Physics G: Nuclear and Particle Physics* 39.10 (2012), p. 105005.
- [59] D. Pappadopulo et al. “Heavy vector triplets: bridging theory and data”. In: *Journal of High Energy Physics* 2014.9 (2014), pp. 1–50.
- [60] The ATLAS collaboration. “Anomaly detection search for new resonances decaying into a Higgs boson and a generic new particle X in hadronic final states using $\sqrt{s} = 13$ TeV pp collisions with the ATLAS detector”. In: *Physical Review D* 108.2306.03637 (2023).
- [61] The CMS collaboration. “Search for new heavy resonances decaying to WW, WZ, ZZ, WH, or ZH boson pairs in the all-jets final state in proton-proton collisions at $\sqrt{s} = 13$ TeV”. In: *Physics Letters B* 844 (2023), p. 137813.
- [62] The ATLAS collaboration. “Search for diboson resonances in hadronic final states in 139 fb^{-1} of pp collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector”. In: *arXiv preprint arXiv:1906.08589* (2019).
- [63] G. Albouy et al. “Theory, phenomenology, and experimental avenues for dark showers: a Snowmass 2021 report”. In: *The European Physical Journal C* 82.12 (2022), p. 1132.
- [64] B. Holdom. “Two U (1)’s and epsilon charge shifts”. In: *Phys. Lett. B;(Netherlands)* 166.2 (1986).
- [65] B. Patt and F. Wilczek. “Higgs-field portal into hidden sectors”. In: *arXiv preprint hep-ph/0605188* (2006).
- [66] A. Falkowski, J. Juknevich, and J. Shelton. “Dark matter through the neutrino portal”. In: *arXiv preprint arXiv:0908.1790* (2009).
- [67] C. Englert, M. McCullough, and M. Spannowsky. “S-channel dark matter simplified models and unitarity”. In: *Physics of the Dark Universe* 14 (2016), pp. 48–56.
- [68] M. Park and M. Zhang. “Tagging a jet from a dark sector with jet substructures at colliders”. In: *Physical Review D* 100.11 (2019), p. 115009.
- [69] The ATLAS Collaboration. “Search for non-resonant production of semi-visible jets using Run 2 data in ATLAS”. In: *Physics Letters B* 848 (2024), p. 138324.
- [70] The CMS collaboration. “Search for resonant production of strongly coupled dark matter in proton-proton collisions at 13 TeV”. In: *Journal of High Energy Physics* 2022.6 (2022), pp. 1–51.
- [71] The CMS collaboration. “Search for dark QCD with emerging jets in proton-proton collisions at $\sqrt{s} = 13$ TeV. Search for new physics with emerging jets in proton-proton collisions at $\sqrt{s} = 13$ TeV”. In: *JHEP* 2407 (2024), p. 142. arXiv: [2403.01556](https://arxiv.org/abs/2403.01556).
- [72] The ATLAS collaboration. “Search for resonant production of dark quarks in the dijet final state with the ATLAS detector”. In: *Journal of High Energy Physics* 2024.2 (2024), pp. 1–35.

- [73] D. Pestre. “Prehistory of CERN: the first suggestions (1949-Jun 1950). The years following the war”. In: (1987). URL: <https://cds.cern.ch/record/151764>.
- [74] *Convention for the establishment of a European organization for nuclear research: Paris, 1st July, 1953 : as amended. Convention pour l'établissement d'une Organisation européenne pour la Recherche nucléaire. Paris, le 1er juillet 1953 : telle qu'elle a été modifiée.* Geneva: CERN, 1971. URL: <https://cds.cern.ch/record/330625>.
- [75] E. Mobs. “The CERN accelerator complex in 2019. Complexe des accélérateurs du CERN en 2019”. In: (2019). URL: <https://cds.cern.ch/record/2684277>.
- [76] L. Arnaudon et al. “Linac4 Technical Design Report”. In: (2006). URL: <https://cds.cern.ch/record/1004186>.
- [77] A. Newborough. “The Upgrade of the CERN Proton Synchrotron Booster Transfer Line Magnets”. In: *IEEE Transactions on Applied Superconductivity* 30.4 (2020), pp. 1–5. DOI: [10.1109/TASC.2020.2967334](https://doi.org/10.1109/TASC.2020.2967334).
- [78] S. Niang et al. “Performance of the CERN Proton Synchrotron internal dump: numerical simulation studies and comparison with beam measurements”. In: *Journal of Instrumentation* 19.6 (June 2024), T06002. DOI: [10.1088/1748-0221/19/06/T06002](https://doi.org/10.1088/1748-0221/19/06/T06002).
- [79] H. Bartosik and G. Rumolo. “Performance of the LHC injector chain after the upgrade and potential development”. In: (2022). arXiv: [2203.09202](https://arxiv.org/abs/2203.09202). URL: <https://cds.cern.ch/record/2847538>.
- [80] R. Bruce et al. “New physics searches with heavy-ion collisions at the CERN Large Hadron Collider”. In: *Journal of Physics G: Nuclear and Particle Physics* 47.6 (May 2020), p. 060501. ISSN: 1361-6471. DOI: [10.1088/1361-6471/ab7ff7](https://doi.org/10.1088/1361-6471/ab7ff7). URL: <http://dx.doi.org/10.1088/1361-6471/ab7ff7>.
- [81] O. S. Brüning et al. “LHC Design Report”. In: CERN Yellow Reports: Monographs (2004). DOI: [10.5170/CERN-2004-003-V-1](https://doi.org/10.5170/CERN-2004-003-V-1). URL: <https://cds.cern.ch/record/782076>.
- [82] D. Boussard and T. P. R. Linnecar. “The LHC Superconducting RF System”. In: (1999). URL: <https://cds.cern.ch/record/410377>.
- [83] L. Rossi. “The LHC superconducting magnets”. In: *Conf. Proc. C* 030512 (2003), p. 141.
- [84] B. J. Holzer. “Introduction to Transverse Beam Dynamics”. In: (2014). DOI: [10.5170/CERN-2013-007.27](https://doi.org/10.5170/CERN-2013-007.27). URL: <http://cds.cern.ch/record/1693320>.
- [85] S. Fartoukh et al. *LHC Configuration and Operational Scenario for Run 3*. Tech. rep. Geneva: CERN, 2021. URL: <https://cds.cern.ch/record/2790409>.
- [86] M. Ferro-Luzzi, W. Herr, and T. Pieloni. “LHC bunch filling schemes for commissioning and initial luminosity optimization”. In: (2008). URL: <https://cds.cern.ch/record/1114612>.

- [87] The ATLAS collaboration. “Total Integrated Luminosity in Run 3 (13.6 TeV p-p data only)”. In: (2023). URL: <https://cds.cern.ch/record/802466https://twiki.cern.ch/twiki/bin/view/AtlasPublic/LuminosityPublicResultsRun3>.
- [88] The ATLAS collaboration. “Interactions per Crossing 2022-2024”. In: (2023). URL: <https://cds.cern.ch/record/802466https://twiki.cern.ch/twiki/bin/view/AtlasPublic/LuminosityPublicResultsRun3>.
- [89] R. M. Bianchi and ATLAS Collaboration. “ATLAS experiment schematic or layout illustration”. In: (2022). URL: <http://cds.cern.ch/record/2837191>.
- [90] The ATLAS collaboration. “ATLAS magnet system: Technical Design Report, 1”. In: Technical design report. ATLAS (1997). DOI: [10.17181/CERN.905C.VDTM](https://doi.org/10.17181/CERN.905C.VDTM). URL: <https://cds.cern.ch/record/338080>.
- [91] Z. Charifouline. “Residual Resistivity Ratio (RRR) Measurements of LHC Superconducting NbTi Cable Strands”. In: (2006). URL: <https://cds.cern.ch/record/970393>.
- [92] The ATLAS Collaboration. “The ATLAS Experiment at the CERN Large Hadron Collider”. In: *Journal of Instrumentation* 3.08 (Aug. 2008), S08003. DOI: [10.1088/1748-0221/3/08/S08003](https://doi.org/10.1088/1748-0221/3/08/S08003). URL: <https://dx.doi.org/10.1088/1748-0221/3/08/S08003>.
- [93] H.H.J. Ten Kate. “The ATLAS superconducting magnet system at the Large Hadron Collider”. In: *Physica C: Superconductivity* 468 (2008), pp. 2137–2142. DOI: [10.1016/j.physc.2008.05.146](https://doi.org/10.1016/j.physc.2008.05.146). URL: <https://doi.org/10.1016/j.physc.2008.05.146>.
- [94] The ATLAS collaboration. “ATLAS inner detector: Technical Design Report, 1”. In: Technical design report. ATLAS (1997). URL: <https://cds.cern.ch/record/331063>.
- [95] S. Haywood et al. “ATLAS inner detector: Technical Design Report, 2”. In: Technical design report. ATLAS (1997). URL: <https://cds.cern.ch/record/331064>.
- [96] M. Mlinarevic. “The ATLAS inner detector trigger performance in Run 3”. In: *PoS LHCP2024* (2025), p. 220. DOI: [10.22323/1.478.0220](https://doi.org/10.22323/1.478.0220).
- [97] M. Capeans et al. “ATLAS Insertable B-Layer Technical Design Report”. In: (2010). URL: <https://cds.cern.ch/record/1291633>.
- [98] The ATLAS collaboration. “ATLAS calorimeter performance: Technical Design Report”. In: Technical design report. ATLAS (1996). URL: <https://cds.cern.ch/record/331059>.
- [99] The ATLAS collaboration. “ATLAS liquid-argon calorimeter: Technical Design Report”. In: Technical design report. ATLAS (1996). DOI: [10.17181/CERN.FWRW.F00Q](https://doi.org/10.17181/CERN.FWRW.F00Q). URL: <https://cds.cern.ch/record/331061>.
- [100] The ATLAS collaboration. “ATLAS tile calorimeter: Technical Design Report”. In: Technical design report. ATLAS (1996). DOI: [10.17181/CERN.JRBJ.7028](https://doi.org/10.17181/CERN.JRBJ.7028). URL: <https://cds.cern.ch/record/331062>.

- [101] The ATLAS collaboration. “ATLAS muon spectrometer: Technical Design Report”. In: Technical design report. ATLAS (1997). URL: <https://cds.cern.ch/record/331068>.
- [102] T. Kawamoto et al. “New Small Wheel Technical Design Report”. In: (2013). URL: <https://cds.cern.ch/record/1552862>.
- [103] The ATLAS collaboration. “Commissioning of the ATLAS Muon Spectrometer with Cosmic Rays. Commissioning of the ATLAS Muon Spectrometer with Cosmic Rays”. In: *Eur. Phys. J. C* 70 (2010), pp. 875–916. DOI: [10.1140/epjc/s10052-010-1415-2](https://doi.org/10.1140/epjc/s10052-010-1415-2). arXiv: [1006.4384](https://arxiv.org/abs/1006.4384). URL: <https://cds.cern.ch/record/1275998>.
- [104] Wolfgang Walkowiak. “ATLAS Plans for the High-Luminosity LHC”. In: *PoS BEAUTY2018* (2018), p. 055. DOI: [10.22323/1.326.0055](https://doi.org/10.22323/1.326.0055).
- [105] The ATLAS collaboration. “Technical Design Report for the Phase-I Upgrade of the ATLAS TDAQ System”. In: (2013). URL: <https://cds.cern.ch/record/1602235>.
- [106] The ATLAS collaboration. “ATLAS level-1 trigger: Technical Design Report”. In: Technical design report. ATLAS (1998). URL: <https://cds.cern.ch/record/381429>.
- [107] P. Jenni et al. “ATLAS high-level trigger, data-acquisition and controls: Technical Design Report”. In: Technical design report. ATLAS (2003). URL: <https://cds.cern.ch/record/616089>.
- [108] The ATLAS Collaboration. “Software and computing for Run 3 of the ATLAS experiment at the LHC”. In: (2024). arXiv: [2404.06335](https://arxiv.org/abs/2404.06335) [hep-ex]. URL: <https://arxiv.org/abs/2404.06335>.
- [109] Torbjorn Sjostrand. “Monte Carlo Generators”. In: *2006 European School of High-Energy Physics*. Nov. 2006, pp. 51–74. arXiv: [hep-ph/0611247](https://arxiv.org/abs/hep-ph/0611247).
- [110] J. Alwall et al. “The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations”. In: *Journal of High Energy Physics* 2014.7 (July 2014). ISSN: 1029-8479. DOI: [10.1007/jhep07\(2014\)079](https://doi.org/10.1007/jhep07(2014)079). URL: [http://dx.doi.org/10.1007/JHEP07\(2014\)079](http://dx.doi.org/10.1007/JHEP07(2014)079).
- [111] S. Höche et al. “QCD matrix elements + parton showers. The NLO case”. In: *Journal of High Energy Physics* 2013.4 (Apr. 2013). ISSN: 1029-8479. DOI: [10.1007/jhep04\(2013\)027](https://doi.org/10.1007/jhep04(2013)027). URL: [http://dx.doi.org/10.1007/JHEP04\(2013\)027](http://dx.doi.org/10.1007/JHEP04(2013)027).
- [112] S. Agostinelli et al. “GEANT4 - A Simulation Toolkit”. In: *Nucl. Instrum. Meth. A* 506 (2003), pp. 250–303. DOI: [10.1016/S0168-9002\(03\)01368-8](https://doi.org/10.1016/S0168-9002(03)01368-8).
- [113] T. M. Carter. “AtlFast3: The Next Generation of Fast Simulation in ATLAS”. In: (2022). URL: <https://cds.cern.ch/record/2815171>.
- [114] The ATLAS collaboration. “Software Performance of the ATLAS Track Reconstruction for LHC Run 3”. In: (2021). URL: <https://cds.cern.ch/record/2766886>.

- [115] G. Aad et al. “Software Performance of the ATLAS Track Reconstruction for LHC Run 3”. In: *Computing and Software for Big Science* 8.1 (Apr. 2024). ISSN: 2510-2044. DOI: [10.1007/s41781-023-00111-y](https://doi.org/10.1007/s41781-023-00111-y). URL: <http://dx.doi.org/10.1007/s41781-023-00111-y>.
- [116] The ATLAS collaboration. “Electron and photon performance measurements with the ATLAS detector using the 2015-2017 LHC proton-proton collision data”. In: *JINST* 14.12 (2019), P12006. DOI: [10.1088/1748-0221/14/12/P12006](https://doi.org/10.1088/1748-0221/14/12/P12006). arXiv: [1908.00005](https://arxiv.org/abs/1908.00005). URL: <https://cds.cern.ch/record/2684552>.
- [117] The ATLAS collaboration. “Electron and photon reconstruction and performance in ATLAS using a dynamical, topological cell clustering-based approach”. In: (2017). URL: <https://cds.cern.ch/record/2298955>.
- [118] S. Marzani, G. Soyez, and M. Spannowsky. “Looking Inside Jets: An Introduction to Jet Substructure and Boosted-object Phenomenology”. In: *Lecture Notes in Physics* (2019). ISSN: 1616-6361. DOI: [10.1007/978-3-030-15709-8](https://doi.org/10.1007/978-3-030-15709-8). URL: <http://dx.doi.org/10.1007/978-3-030-15709-8>.
- [119] G. F. Sterman and S. Weinberg. “Jets from Quantum Chromodynamics”. In: *Phys. Rev. Lett.* 39 (1977), p. 1436. DOI: [10.1103/PhysRevLett.39.1436](https://doi.org/10.1103/PhysRevLett.39.1436).
- [120] G. P. Salam and G. Soyez. “A practical seedless infrared-safe cone jet algorithm”. In: *Journal of High Energy Physics* 2007.05 (May 2007), pp. 086–086. ISSN: 1029-8479. DOI: [10.1088/1126-6708/2007/05/086](https://doi.org/10.1088/1126-6708/2007/05/086). URL: <http://dx.doi.org/10.1088/1126-6708/2007/05/086>.
- [121] S. D. Ellis and D. E. Soper. “Successive combination jet algorithm for hadron collisions”. In: *Physical Review D* 48.7 (Oct. 1993), pp. 3160–3166. ISSN: 0556-2821. DOI: [10.1103/physrevd.48.3160](https://doi.org/10.1103/physrevd.48.3160). URL: <http://dx.doi.org/10.1103/PhysRevD.48.3160>.
- [122] M. Wobisch and T. Wengler. “Hadronization Corrections to Jet Cross Sections in Deep-Inelastic Scattering”. In: (1999). arXiv: [hep-ph/9907280](https://arxiv.org/abs/hep-ph/9907280) [[hep-ph](https://arxiv.org/abs/hep-ph/9907280)]. URL: <https://arxiv.org/abs/hep-ph/9907280>.
- [123] Ma. Cacciari, G. P. Salam, and G. Soyez. “The anti-ktjet clustering algorithm”. In: *Journal of High Energy Physics* 2008.04 (Apr. 2008), pp. 063–063. ISSN: 1029-8479. DOI: [10.1088/1126-6708/2008/04/063](https://doi.org/10.1088/1126-6708/2008/04/063). URL: <http://dx.doi.org/10.1088/1126-6708/2008/04/063>.
- [124] P. Berta et al. “Particle-level pileup subtraction for jets and jet shapes”. In: *Journal of High Energy Physics* 2014.6 (June 2014). ISSN: 1029-8479. DOI: [10.1007/jhep06\(2014\)092](https://doi.org/10.1007/jhep06(2014)092). URL: [http://dx.doi.org/10.1007/JHEP06\(2014\)092](http://dx.doi.org/10.1007/JHEP06(2014)092).
- [125] M. Cacciari, G. P. Salam, and G. Soyez. “SoftKiller, a particle-level pileup removal method”. In: *The European Physical Journal C* 75.2 (Feb. 2015). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-015-3267-2](https://doi.org/10.1140/epjc/s10052-015-3267-2). URL: <http://dx.doi.org/10.1140/epjc/s10052-015-3267-2>.
- [126] D. Bertolini et al. “Pileup per particle identification”. In: *Journal of High Energy Physics* 2014.10 (Oct. 2014). ISSN: 1029-8479. DOI: [10.1007/jhep10\(2014\)059](https://doi.org/10.1007/jhep10(2014)059). URL: [http://dx.doi.org/10.1007/JHEP10\(2014\)059](http://dx.doi.org/10.1007/JHEP10(2014)059).

- [127] The ATLAS collaboration. “Topological cell clustering in the ATLAS calorimeters and its performance in LHC Run 1”. In: *The European Physical Journal C* 77.7 (July 2017). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-017-5004-5](https://doi.org/10.1140/epjc/s10052-017-5004-5). URL: <http://dx.doi.org/10.1140/epjc/s10052-017-5004-5>.
- [128] The ATLAS collaboration. “Jet reconstruction and performance using particle flow with the ATLAS Detector”. In: *The European Physical Journal C* 77.7 (). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-017-5031-2](https://doi.org/10.1140/epjc/s10052-017-5031-2). URL: <http://dx.doi.org/10.1140/epjc/s10052-017-5031-2>.
- [129] M. Schram. “Performance of the ATLAS forward calorimeter and search for the invisible Higgs via vector boson fusion at ATLAS”. PhD thesis. Carleton U., 2008.
- [130] The ATLAS collaboration. “Improving jet substructure performance in ATLAS using Track-CaloClusters”. In: (2017). URL: <https://cds.cern.ch/record/2275636>.
- [131] The ATLAS collaboration. “Optimisation of large-radius jet reconstruction for the ATLAS detector in 13 TeV proton–proton collisions”. In: *Eur. Phys. J. C* 81.4 (2021), p. 334. DOI: [10.1140/epjc/s10052-021-09054-3](https://doi.org/10.1140/epjc/s10052-021-09054-3). arXiv: [2009.04986](https://arxiv.org/abs/2009.04986). URL: <https://cds.cern.ch/record/2730141>.
- [132] The ATLAS collaboration. “New techniques for jet calibration with the ATLAS detector”. In: *The European Physical Journal C* 83.8 (Aug. 2023). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-023-11837-9](https://doi.org/10.1140/epjc/s10052-023-11837-9). URL: <http://dx.doi.org/10.1140/epjc/s10052-023-11837-9>.
- [133] The ATLAS collaboration. “In situ calibration of large-radius jet energy and mass in 13 TeV proton–proton collisions with the ATLAS detector”. In: *The European Physical Journal C* 79.2 (Feb. 2019). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-019-6632-8](https://doi.org/10.1140/epjc/s10052-019-6632-8). URL: <http://dx.doi.org/10.1140/epjc/s10052-019-6632-8>.
- [134] The ATLAS Collaboration. “Simultaneous energy and mass calibration of large-radius jets with the ATLAS detector using a deep neural network”. In: (2023). arXiv: [2311.08885 \[hep-ex\]](https://arxiv.org/abs/2311.08885). URL: <https://arxiv.org/abs/2311.08885>.
- [135] J. Thaler and K. Van Tilburg. “Identifying boosted objects with N-subjettiness”. In: *Journal of High Energy Physics* 2011.3 (Mar. 2011). ISSN: 1029-8479. DOI: [10.1007/jhep03\(2011\)015](https://doi.org/10.1007/jhep03(2011)015). URL: [http://dx.doi.org/10.1007/JHEP03\(2011\)015](http://dx.doi.org/10.1007/JHEP03(2011)015).
- [136] A. J. Larkoski, G. P. Salam, and J. Thaler. “Energy correlation functions for jet substructure”. In: *Journal of High Energy Physics* 2013.6 (June 2013). ISSN: 1029-8479. DOI: [10.1007/jhep06\(2013\)108](https://doi.org/10.1007/jhep06(2013)108). URL: [http://dx.doi.org/10.1007/JHEP06\(2013\)108](http://dx.doi.org/10.1007/JHEP06(2013)108).
- [137] The ATLAS Collaboration. “Identification and energy calibration of hadronically decaying tau leptons with the ATLAS experiment in pp collisions at $\sqrt{s} = 8$ s = 8 TeV TeV”. In: *The European Physical Journal C* 75.7 (July 2015). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-015-3500-z](https://doi.org/10.1140/epjc/s10052-015-3500-z). URL: <http://dx.doi.org/10.1140/epjc/s10052-015-3500-z>.

- [138] The ATLAS Collaboration. “The performance of missing transverse momentum reconstruction and its significance with the ATLAS detector using 140 fb⁻¹ of $\sqrt{s} = 13$ TeV pp collisions”. In: (2024). arXiv: [2402.05858](https://arxiv.org/abs/2402.05858) [hep-ex]. URL: <https://arxiv.org/abs/2402.05858>.
- [139] V. Balasubramanian. “Brain power”. In: *Proc Natl Acad Sci U S A* (Aug. 2021). DOI: [10.1073/pnas.2107022118](https://doi.org/10.1073/pnas.2107022118).
- [140] W. S. McCulloch and W. Pitts. “A logical calculus of the ideas immanent in nervous activit”. In: *The bulletin of mathematical biophysics* (Dec. 1943). DOI: [10.1007/BF02478259](https://doi.org/10.1007/BF02478259).
- [141] M. Krenn et al. “Forecasting the future of artificial intelligence with machine learning-based link prediction in an exponentially growing knowledge network”. In: *Nature Machine Intelligence* 5.11 (Oct. 2023), pp. 1326–1335. ISSN: 2522-5839. DOI: [10.1038/s42256-023-00735-0](https://doi.org/10.1038/s42256-023-00735-0). URL: <http://dx.doi.org/10.1038/s42256-023-00735-0>.
- [142] T. Mitchell. “Machine Learning”. In: (1997).
- [143] C. M. Bishop. “Pattern Recognition and Machine Learning”. In: (2006).
- [144] I. Goodfellow, Y. Bengio, and A. Courville. “Deep Learning”. In: *MIT Press* (2016). URL: <http://www.deeplearningbook.org>.
- [145] F. Rosenblatt. *Principles of Neurodynamics - Perceptrons and the theory of brain mechanisms*. New York: Cornell Aeronautical Laboratory, INC., 1961.
- [146] D. Hendrycks and K. Gimpel. “Gaussian error linear units (gelus)”. In: *arXiv preprint arXiv:1606.08415* (2016).
- [147] J. Yin, K. Y. Ngiam, and H. H. Teo. “Role of Artificial Intelligence Applications in Real-Life Clinical Practice: Systematic Review”. In: *J Med Internet Res*. (2021). DOI: [10.2196/25759](https://doi.org/10.2196/25759).
- [148] P. K. Diederik and B. Jimmy. “Adam: A Method for Stochastic Optimization”. In: (2017). arXiv: [1412.6980](https://arxiv.org/abs/1412.6980) [cs.LG]. URL: <https://arxiv.org/abs/1412.6980>.
- [149] Wikipedia The Free Encyclopedia. “Receiver operating characteristic”. In: (2024). URL: https://en.wikipedia.org/wiki/Receiver_operating_characteristic.
- [150] K. Albertsson et al. “Machine Learning in High Energy Physics Community White Paper”. In: (2019). arXiv: [1807.02876](https://arxiv.org/abs/1807.02876) [physics.comp-ph]. URL: <https://arxiv.org/abs/1807.02876>.
- [151] D. Guest, K. Cranmer, and D. Whiteson. “Deep Learning and Its Application to LHC Physics”. In: *Annual Review of Nuclear and Particle Science* 68.1 (Oct. 2018), pp. 161–181. ISSN: 1545-4134. DOI: [10.1146/annurev-nucl-101917-021019](https://doi.org/10.1146/annurev-nucl-101917-021019). URL: <http://dx.doi.org/10.1146/annurev-nucl-101917-021019>.
- [152] A. Radovic et al. “Machine learning at the energy and intensity frontiers of particle physics”. In: *Nature* 560 (2018), pp. 41–48. DOI: [10.1038/s41586-018-0361-2](https://doi.org/10.1038/s41586-018-0361-2). URL: <http://cds.cern.ch/record/2647378>.

- [153] A. Radovic et al. “Machine learning at the energy and intensity frontiers of particle physics”. In: *Nature* 560.7716 (2018), pp. 41–48.
- [154] A. Hoecker et al. “TMVA - Toolkit for Multivariate Data Analysis”. In: (2009). arXiv: [physics/0703039](https://arxiv.org/abs/physics/0703039) [[physics.data-an](https://arxiv.org/abs/physics/0703039)]. URL: <https://arxiv.org/abs/physics/0703039>.
- [155] The ATLAS collaboration. “Deep Learning for Pion Identification and Energy Calibration with the ATLAS Detector”. In: (2020). URL: <https://cds.cern.ch/record/2724632>.
- [156] Q. Portillo and M. Dilia. “Point Cloud Deep Learning Methods for Pion Reconstruction in the ATLAS Detector”. In: (2023). DOI: [10.22323/1.414.1076](https://cds.cern.ch/record/2839615). URL: <https://cds.cern.ch/record/2839615>.
- [157] G. Aad et al. “ATLAS flavour-tagging algorithms for the LHC Run 2 pp collision dataset”. In: *The European Physical Journal C* 83.7 (July 2023). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-023-11699-1](https://doi.org/10.1140/epjc/s10052-023-11699-1). URL: <http://dx.doi.org/10.1140/epjc/s10052-023-11699-1>.
- [158] The ATLAS collaboration. *Graph Neural Network Jet Flavour Tagging with the ATLAS Detector*. Tech. rep. Geneva: CERN, 2022. URL: <https://cds.cern.ch/record/2811135>.
- [159] ATLAS public results. “Jet Flavour Tagging with GN1 and DL1d. Generator dependence, Run 2 and Run 3 data agreement studies”. In: (2023). URL: <https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PLOTS/FTAG-2023-01/>.
- [160] G. Kasieczka, B. Nachman, and D. Shih. “LHC Olympics 2020”. In: (2020). URL: <https://lhco2020.github.io/homepage/>.
- [161] F. E. Grubbs. “Procedures for Detecting Outlying Observations in Samples”. In: *Technometrics* 11.1 (1969), pp. 1–21. DOI: [10.1080/00401706.1969.10490657](https://doi.org/10.1080/00401706.1969.10490657).
- [162] G. Pang et al. “Deep Learning for Anomaly Detection: A Review”. In: *ACM Computing Surveys* 54.2 (Mar. 2021), pp. 1–38. ISSN: 1557-7341. DOI: [10.1145/3439950](https://doi.org/10.1145/3439950). URL: <http://dx.doi.org/10.1145/3439950>.
- [163] V. Belis, P. Odagiu, and T. K. Aarrestad. “Machine learning for anomaly detection in particle physics”. In: *Reviews in Physics* 12 (Dec. 2024), p. 100091. ISSN: 2405-4283. DOI: [10.1016/j.revip.2024.100091](https://doi.org/10.1016/j.revip.2024.100091). URL: <http://dx.doi.org/10.1016/j.revip.2024.100091>.
- [164] T. Finke et al. “Autoencoders for unsupervised anomaly detection in high energy physics”. In: *Journal of High Energy Physics* 2021.6 (June 2021). ISSN: 1029-8479. DOI: [10.1007/jhep06\(2021\)161](https://doi.org/10.1007/jhep06(2021)161). URL: [http://dx.doi.org/10.1007/JHEP06\(2021\)161](http://dx.doi.org/10.1007/JHEP06(2021)161).
- [165] P. Jawahar et al. “Improving variational autoencoders for new physics detection at the LHC with normalizing flows”. In: *Frontiers in big Data* 5 (2022), p. 803685.

- [166] L. Vaslin, V. Barra, and J. Donini. “GAN-AE: an anomaly detection algorithm for New Physics search in LHC data”. In: *The European Physical Journal C* 83.11 (Nov. 2023). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-023-12169-4](https://doi.org/10.1140/epjc/s10052-023-12169-4). URL: <http://dx.doi.org/10.1140/epjc/s10052-023-12169-4>.
- [167] The ATLAS collaboration. “Search for new phenomena in two-body invariant mass distributions using unsupervised machine learning for anomaly detection at $\sqrt{s} = 13$ TeV with the ATLAS detector”. In: *arXiv preprint arXiv:2307.01612* (2023).
- [168] A. Hallin et al. “Classifying anomalies through outer density estimation”. In: *Physical Review D* 106.5 (2022), p. 055006.
- [169] L. M. Dery et al. “Weakly supervised classification in high energy physics”. In: *Journal of High Energy Physics* 2017.5 (2017), pp. 1–11.
- [170] E. M. Metodiev, B. Nachman, and J. Thaler. “Classification without labels: Learning from mixed samples in high energy physics”. In: *Journal of High Energy Physics* 2017.10 (2017), pp. 1–18.
- [171] O. Amram and C. M. Suarez. “Tag N’Train: a technique to train improved classifiers on unlabeled data”. In: *Journal of High Energy Physics* 2021.1 (2021), pp. 1–21.
- [172] M. F. Chen, B. Nachman, and F. Sala. “Resonant anomaly detection with multiple reference datasets”. In: *Journal of High Energy Physics* 2023.7 (July 2023). ISSN: 1029-8479. DOI: [10.1007/jhep07\(2023\)188](https://doi.org/10.1007/jhep07(2023)188). URL: [http://dx.doi.org/10.1007/JHEP07\(2023\)188](http://dx.doi.org/10.1007/JHEP07(2023)188).
- [173] G. Grosso et al. “Robust resonant anomaly detection with NPLM”. In: *arXiv preprint arXiv:2501.01778* (2025).
- [174] D. Sengupta et al. “CURTAINS flows for flows: Constructing unobserved regions with maximum likelihood estimation”. In: *SciPost Physics* 17.2 (2024), p. 046.
- [175] S. E. Park et al. “Quasi anomalous knowledge: searching for new physics with embedded knowledge”. In: *Journal of High Energy Physics* 2021.6 (2021), pp. 1–26.
- [176] G. Kasieczka et al. “The LHC Olympics 2020 a community challenge for anomaly detection in high energy physics”. In: *Reports on Progress in Physics* 84.12 (Dec. 2021), p. 124201. ISSN: 1361-6633. DOI: [10.1088/1361-6633/ac36b9](https://doi.org/10.1088/1361-6633/ac36b9). URL: <http://dx.doi.org/10.1088/1361-6633/ac36b9>.
- [177] K. S. Agashe et al. “LHC signals from cascade decays of warped vector resonances”. In: *Journal of High Energy Physics* 2017.5 (2017), pp. 1–51.
- [178] G. Kasieczka, B. Nachman, and D. Shih. “Official Datasets for LHC Olympics 2020 Anomaly Detection Challenge”. In: (2019). URL: <https://zenodo.org/records/4536624>.
- [179] G. Kasieczka, B. Nachman, and D. Shih. “RD Dataset for LHC Olympics 2020 Anomaly Detection Challenge”. In: (2019). URL: <https://zenodo.org/records/6466204>.

- [180] M. Cacciari, G. P. Salam, and G. Soyez. “FastJet user manual: (for version 3.0.2)”. In: *The European Physical Journal C* 72.3 (Mar. 2012). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-012-1896-2](https://doi.org/10.1140/epjc/s10052-012-1896-2). URL: <http://dx.doi.org/10.1140/epjc/s10052-012-1896-2>.
- [181] M. Dasgupta et al. “Towards an understanding of jet substructure”. In: *Journal of High Energy Physics* 2013.9 (2013), pp. 1–57.
- [182] T. S. Roy and A. H. Vijay. “A robust anomaly finder based on autoencoders”. In: *arXiv preprint arXiv:1903.02032* (2019).
- [183] L. Sauras-Altuzarra and E. W. Weisstein. “Adjacency Matrix from MathWorld—A Wolfram Web Resource”. In: (2025). URL: <https://mathworld.wolfram.com/AdjacencyMatrix.html>.
- [184] W. L. Hamilton. “Graph representation learning”. In: (2020).
- [185] Wikipedia The Free Encyclopedia. “Clustering coefficient”. In: (2024). URL: https://en.wikipedia.org/wiki/Clustering_coefficient.
- [186] W. Minjie et al. “Deep Graph Library: A Graph-Centric, Highly-Performant Package for Graph Neural Networks”. In: (2020). arXiv: [1909.01315](https://arxiv.org/abs/1909.01315) [cs.LG]. URL: <https://arxiv.org/abs/1909.01315>.
- [187] J. N. Morgan and J. A. Sonquist. “Problems in the Analysis of Survey Data, and a proposal”. In: *Journal of the American Statistical Association* 58.302 (1963). URL: <http://www.jstor.org/stable/2283276>.
- [188] Y. Coadou. “Boosted Decision Trees”. In: (Feb. 2022), pp. 9–58. DOI: [10.1142/9789811234033_0002](https://doi.org/10.1142/9789811234033_0002). URL: http://dx.doi.org/10.1142/9789811234033_0002.
- [189] scikit-learn. “RandomForestClassifier”. In: (2007). URL: <https://scikit-learn.org/1.5/modules/ensemble.html#forest>.
- [190] L. E. Raileanu and K. Stoffel. “Theoretical comparison between the gini index and information gain criteria”. In: *Annals of Mathematics and Artificial Intelligence* 41 (2004), pp. 77–93.
- [191] A. Vaswani et al. “Attention Is All You Need”. In: (2023). arXiv: [1706.03762](https://arxiv.org/abs/1706.03762) [cs.CL]. URL: <https://arxiv.org/abs/1706.03762>.
- [192] H. Qu, C. Li, and S. Qian. “Particle Transformer for Jet Tagging”. In: (2024). arXiv: [2202.03772](https://arxiv.org/abs/2202.03772) [hep-ph]. URL: <https://arxiv.org/abs/2202.03772>.
- [193] PyTorch Foundation. “MultiheadAttention”. In: (2023). URL: <https://pytorch.org/docs/stable/generated/torch.nn.MultiheadAttention.html>.
- [194] G. DeZoort. “Graph Neural Networks at the Large Hadron Collider Lecture”. In: (2022). URL: https://github.com/savvy379/codashep_ml_2022/blob/main/lecture4_gnns/gnn_lecture.pdf.
- [195] C. Liu et al. “Graph Pooling for Graph Neural Networks: Progress, Challenges, and Opportunities”. In: (2023). arXiv: [2204.07321](https://arxiv.org/abs/2204.07321) [cs.LG]. URL: <https://arxiv.org/abs/2204.07321>.

- [196] T. N. Kipf and M. Welling. “Semi-Supervised Classification with Graph Convolutional Networks”. In: (2017). arXiv: [1609.02907 \[cs.LG\]](https://arxiv.org/abs/1609.02907). URL: <https://arxiv.org/abs/1609.02907>.
- [197] Y. Wang et al. “Dynamic graph cnn for learning on point clouds”. In: *ACM Transactions on Graphics (tog)* 38.5 (2019), pp. 1–12.
- [198] W. Hamilton, Z. Ying, and J. Leskovec. “Inductive representation learning on large graphs”. In: *Advances in neural information processing systems* 30 (2017).
- [199] A. Leman and B. Weisfeiler. “A reduction of a graph to a canonical form and an algebra arising during this reduction”. In: *Nauchno-Technicheskaya Informatsiya* 2.9 (1968), pp. 12–16.
- [200] K. Xu et al. “How powerful are graph neural networks?” In: *arXiv preprint arXiv:1810.00826* (2018).
- [201] W. Hu et al. *Strategies for Pre-training Graph Neural Networks*. 2020. arXiv: [1905.12265 \[cs.LG\]](https://arxiv.org/abs/1905.12265). URL: <https://arxiv.org/abs/1905.12265>.
- [202] P. Veličković et al. “Graph attention networks”. In: *arXiv preprint arXiv:1710.10903* (2017).
- [203] K. Kamiński et al. “Rossmann-toolbox: a deep learning-based protocol for the prediction and design of cofactor specificity in Rossmann fold proteins”. In: *Briefings in bioinformatics* 23.1 (2022), bbab371.
- [204] J. Chen and H. Chen. *Edge-Featured Graph Attention Network*. 2021. arXiv: [2101.07671 \[cs.LG\]](https://arxiv.org/abs/2101.07671). URL: <https://arxiv.org/abs/2101.07671>.
- [205] L. Ruff et al. “Deep one-class classification”. In: (2018), pp. 4393–4402.
- [206] R. Ma et al. “Deep graph-level anomaly detection by glocal knowledge distillation”. In: (2022), pp. 704–714.
- [207] C. Qiu et al. “Raising the bar in graph-level anomaly detection”. In: *arXiv preprint arXiv:2205.13845* (2022).
- [208] PyTorch Foundation. “MSELoss”. In: (2024). URL: <https://pytorch.org/docs/stable/generated/torch.nn.MSELoss.html>.
- [209] The ATLAS collaboration. “Weakly supervised anomaly detection for resonant new physics in the dijet final state using proton-proton collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector”. In: (2025). arXiv: [2502.09770 \[hep-ex\]](https://arxiv.org/abs/2502.09770). URL: <https://arxiv.org/abs/2502.09770>.
- [210] The CMS collaboration. “Model-agnostic search for dijet resonances with anomalous jet substructure in proton-proton collisions at $\sqrt{s} = 13$ TeV”. In: (2024). URL: <https://cds.cern.ch/record/2892677>.
- [211] The SciPy community. “`scipy.stats.rv_histogram`”. In: (2025). URL: https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.rv_histogram.html.
- [212] P. Virtanen et al. “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”. In: *Nature Methods* 17 (2020), pp. 261–272. DOI: [10.1038/s41592-019-0686-2](https://doi.org/10.1038/s41592-019-0686-2).

- [213] K. Cranmer et al. “HistFactory: A tool for creating statistical models for use with RooFit and RooStats”. In: (2012). DOI: [10.17181/CERN-OPEN-2012-016](https://cds.cern.ch/record/1456844). URL: <https://cds.cern.ch/record/1456844>.
- [214] ATLAS collaboration. “Preliminary analysis of the luminosity calibration for the ATLAS 13.6 TeV data recorded in 2023”. In: ATL-DAPR-PUB-2024-001 (2024), p. 29. DOI: [10.3204/PUBDB-2024-07912](https://bib-pubdb1.desy.de/record/619807). URL: <https://bib-pubdb1.desy.de/record/619807>.
- [215] L. Moneta. “StandardHypoTestInvDemo.C File Reference”. In: (2025). URL: https://root.cern.ch/doc/master/StandardHypoTestInvDemo_8C.html.
- [216] The ATLAS collaboration. “Recommendations for the Modeling of Smooth Backgrounds”. In: (2020). URL: <https://cds.cern.ch/record/2743717>.
- [217] W. Verkerke and D. P. Kirkby. “The RooFit toolkit for data modeling”. In: *eConf C0303241* (2003). Ed. by L. Lyons and Muge Karagoz, MOLT007. arXiv: [physics/0306116](https://arxiv.org/abs/physics/0306116).
- [218] “Longer term LHC schedule”. In: (2024). URL: <http://lhc-commissioning.web.cern.ch/schedule/LHC-long-term.htm>.
- [219] O. Aberle et al. “High-Luminosity Large Hadron Collider (HL-LHC): Technical design report”. In: CERN Yellow Reports: Monographs (2020). DOI: [10.23731/CYRM-2020-0010](https://cds.cern.ch/record/2749422). URL: <https://cds.cern.ch/record/2749422>.
- [220] R. T. Garcia. “Luminosity forecasts”. In: (2024). URL: <http://lhc-commissioning.web.cern.ch/schedule/HL-LHC-plots.htm>.
- [221] The ATLAS collaboration. “Letter of Intent for the Phase-II Upgrade of the ATLAS Experiment”. In: (2012). Draft version for comments. URL: <https://cds.cern.ch/record/1502664>.
- [222] The ATLAS collaboration. “ATLAS Phase-II Upgrade Scoping Document”. In: (2015). DOI: [10.17181/CERN.7CRX.AJHP](https://cds.cern.ch/record/2055248). URL: <https://cds.cern.ch/record/2055248>.
- [223] The ATLAS collaboration. “Technical Design Report for the ATLAS Inner Tracker Pixel Detector”. In: (2017). DOI: [10.17181/CERN.FOZZ.ZP3Q](https://cds.cern.ch/record/2285585). URL: <https://cds.cern.ch/record/2285585>.
- [224] The ATLAS collaboration. “Technical Design Report for the ATLAS Inner Tracker Strip Detector”. In: (2017). URL: <https://cds.cern.ch/record/2257755>.
- [225] The ATLAS collaboration. “ATLAS Liquid Argon Calorimeter Phase-II Upgrade: Technical Design Report”. In: (2017). DOI: [10.17181/CERN.6QIO.YGH0](https://cds.cern.ch/record/2285582). URL: <https://cds.cern.ch/record/2285582>.
- [226] The ATLAS collaboration. “Technical Design Report for the Phase-II Upgrade of the ATLAS Tile Calorimeter”. In: (2017). URL: <https://cds.cern.ch/record/2285583>.
- [227] The ATLAS collaboration. “Technical Design Report for the Phase-II Upgrade of the ATLAS Muon Spectrometer”. In: (2017). URL: <https://cds.cern.ch/record/2285580>.

- [228] The ATLAS collaboration. “Technical Design Report for the Phase-II Upgrade of the ATLAS TDAQ System”. In: (2017). DOI: [10.17181/CERN.2.LBB.4IAL](https://doi.org/10.17181/CERN.2.LBB.4IAL). URL: <https://cds.cern.ch/record/2285584>.
- [229] L. Massa. “The BIS78 Resistive Plate Chambers upgrade of the ATLAS Muon Spectrometer for the LHC Run-3”. In: *Journal of Instrumentation* 15 (2020). DOI: [10.1088/1748-0221/15/10/C10026](https://doi.org/10.1088/1748-0221/15/10/C10026).
- [230] F. Morodei. “Development of the Barrel Sector Logic firmware of the ATLAS Muon Spectrometer for the High-Luminosity LHC”. In: *Nuovo Cimento C* 47.3 (2024), p. 128. DOI: [10.1393/ncc/i2024-24128-6](https://doi.org/10.1393/ncc/i2024-24128-6). URL: <https://cds.cern.ch/record/2904266>.
- [231] U. Farooq et al. “FPGA architectures: An overview”. In: *Tree-Based Heterogeneous FPGA Architectures: Application Specific Exploration and Optimization* (2012), pp. 7–48.
- [232] V. D’Amico. “Studio delle prestazioni del trigger per muoni nella zona centrale del rivelatore ATLAS per la fase ad alta luminosità di LHC”. In: (2017).
- [233] S. Francescato et al. “Model compression and simplification pipelines for fast deep neural network inference in FPGAs in HEP”. In: *The European Physical Journal C* 81.11 (2021), p. 969.
- [234] I. Goodfellow. “Deep learning”. In: (2016).
- [235] F. Chollet. “Deep learning with Python”. In: (2021).
- [236] N. Ahamed. “Convolutional Neural Networks”. In: (2023). URL: <https://blog.devgenius.io/convolutional-neural-networks-98e2067072a9>.
- [237] A. Krizhevsky, I. Sutskever, and G. E. Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems* 25 (2012).
- [238] K. Simonyan. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).
- [239] C. Szegedy et al. “Going deeper with convolutions”. In: (2015), pp. 1–9.
- [240] K. He et al. “Deep residual learning for image recognition”. In: (2016), pp. 770–778.
- [241] B. Jacob et al. “Quantization and training of neural networks for efficient integer-arithmetic-only inference”. In: (2018), pp. 2704–2713.
- [242] C. N. Coelho et al. “Automatic heterogeneous quantization of deep neural networks for low-latency inference on the edge for particle detectors”. In: *Nature Machine Intelligence* 3.8 (2021), pp. 675–686.
- [243] QKeras team. “QKeras repository”. In: (2021). URL: <https://github.com/google/qkeras>.
- [244] J. Gou et al. “Knowledge distillation: A survey”. In: *International Journal of Computer Vision* 129.6 (2021), pp. 1789–1819.
- [245] F. Chollet et al. “Keras”. In: (2015).

- [246] M. Abadi et al. “TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems”. In: (2015). URL: <https://www.tensorflow.org/>.
- [247] Keras team. “Conv2D layer”. In: (2025). URL: https://keras.io/api/layers/convolution_layers/convolution2d/.
- [248] S. Abi-Karam et al. “GenGNN: A Generic FPGA Framework for Graph Neural Network Acceleration”. In: (2022). arXiv: [2201.08475 \[cs.LG\]](https://arxiv.org/abs/2201.08475). URL: <https://arxiv.org/abs/2201.08475>.
- [249] Z. Que et al. “LL-GNN: Low Latency Graph Neural Networks on FPGAs for High Energy Physics”. In: *ACM Transactions on Embedded Computing Systems* 23.2 (Mar. 2024), pp. 1–28. ISSN: 1558-3465. DOI: [10.1145/3640464](https://doi.org/10.1145/3640464). URL: <http://dx.doi.org/10.1145/3640464>.
- [250] A. Elabd et al. “Graph Neural Networks for Charged Particle Tracking on FPGAs”. In: *Frontiers in Big Data* 5 (Mar. 2022). ISSN: 2624-909X. DOI: [10.3389/fdata.2022.828666](https://doi.org/10.3389/fdata.2022.828666). URL: <http://dx.doi.org/10.3389/fdata.2022.828666>.
- [251] FastML Team. “fastmachinelearning/hls4ml”. Version v0.8.1. In: (2023). DOI: [10.5281/zenodo.1201549](https://doi.org/10.5281/zenodo.1201549). URL: <https://github.com/fastmachinelearning/hls4ml>.
- [252] G. Aad et al. “Standalone Vertex Finding in the ATLAS Muon Spectrometer.” In: *JINST* 9 (2014), P02001. DOI: [10.1088/1748-0221/9/02/P02001](https://doi.org/10.1088/1748-0221/9/02/P02001). arXiv: [1311.7070](https://arxiv.org/abs/1311.7070). URL: <https://cds.cern.ch/record/1631701>.
- [253] H. Schreiner, J. Pivarski, and S. Chopra. “scikit-hep/vector git repository”. In: (2025). DOI: [10.5281/zenodo.5942082](https://doi.org/10.5281/zenodo.5942082). URL: <https://github.com/scikit-hep/vector>.

Acknowledgements

These past three and a half years have been the craziest of my life, a roller coaster of emotions and experiences. Nothing truly prepares you for nights spent setting up slides for the next day's presentation, crying over non-compiling code, navigating endless bureaucracy and pouring your heart and soul into improving a Neural Network's performance. But at every moment, I knew I could rely on many people to listen to my countless complaints and claims and help me with their wise advice. Although the experience was tough, I made new friends from around the world and strengthened older friendships and relationships that made the path easier and never let me feel alone. Today, as I reach the end of this tremendous achievement, I am sincerely grateful to all of you who enriched me as a person, beyond my academic growth. After reading this thesis, you should all know (and if you don't, go to Sec. 1.1.3) that life without colours is a misfortune and you have been the colour of these past years. (Bad) jokes aside, I truly want to thank you all, even those who, inevitably, are not mentioned in this acknowledgements section because my forgetfulness. Before Francica has a chance to complain, let me stress that the order of the acknowledgements simply reflects a scale from my academic career to my family, but I love you all.

First of all, I want to thank the Saletta crew, Lorenzo, Davide, Elena, Lox, Padi, Federico, Maria and Lucchetto, for all the support and meaningful conversations, in Saletta and outside. In particular, thank you to Elena that carefully monitored my daily water intake while being at CERN and always tried to cheer me up, even in the darkest PhD days. I have truly found a friend in you.

Thanks to Giuliano for your suggestions and comments, and sorry for being a constant presence in your Mattermost messages during Sergio's walk.

Thanks to Antonio, the Neapolitan angel, with whom I shared the heavy burden of the analysis. More than just a colleague, you have been a trusted analysis buddy and an excellent listener (even when I send to you long Whatsapp audios).

Thanks to the Seminarising organising committee, from the old guard with Michele, Beth and Stefano, to the newcomers Loris, Bea, Paolo, Carlo and Alessandro. You made the complex task of organizing the PhD Seminars feel easy.

Thanks to all the Cinefellow friends, Pino, Antimo, Francino, Elisabetta, Limone, Zuolo, Valery, Sebba, MJ, Friti, Ago and Bob, who brightened the grey Genevan days, the R1 lunches and the 18's rides.

Thanks to the wonderful people I met at ESHEP2023 in Grenaa, Ana, Arnau, Arthur, Chiara, Chris, Giovanni the young and Giovanni the wise (guess who is

who), Léo, Madri, Paula, Seulgi, Tizi and Erlend. You helped me realize that it does not matter how long you have known someone, deep, lasting friendships can form in just a few days. A special thanks to Alberto, also met in Grenaa, who has the courage to listen to my podcast-long Whatsapp audios. I know I can talk to you about anything and I will always find a sensitive, thoughtful listener even when we disagree.

Thanks to my flatmates from Encyclopédie: Mattia for being up all night and thereby letting me sleep more than humanly possible (according to the Encyclopédie Sleep Conservation Law) and for the sacred guanciale rule in the fridge; Matteo for supporting my strange addiction to Genevan-people-who-cannot-park and for the salsicci and Dario for your incredible generosity and for the granita recipe.

Thanks to Rappi, promoted to mum on the field, for the care and patience you showed in response to my most annoying requests, and for the lattuccio.

Thanks to my Tibbu flatmate Gege, for putting up with my pop-corn addiction and not overreacting every time I accidentally shut off the hot water.

Thanks to the Ozzezosi friends, Alice, Clappi, Ems, Silviu, Ric, Ric the Penguin, Rob, Sciotti, Jacopo and Puccio, with whom it has become nearly impossible to meet because you are all too talented and scattered around the world. Despite the distances, you have always rooted for me and believed in my success.

Thanks to my dear WeAreCool friends, Cami, Vinci, Francica, Ele, Ludo, Trava and Sara. You've been my pillars since high school (or even earlier) and continue to be so today. I know you are always there for me. In particular, thank you to Vinci for your extraordinary kindness and to Cami, my best friend, my wing.

Thanks to Flalla, for our insightful discussions, the Pindaric flights and the inner peace you have always transmitted to me.

Thanks to Andrea, my best friend and trusted confident.

Thanks to my family, who always made me feel cared for, and especially Cousin for our physics-related discussions.

Thanks to Aunt Lalla, more like a sister to me, for being there when I needed it most.

Thanks to Mum and Dad, you never doubted I could do it, even when I was convinced I could not. Thank you for all the opportunities you have given me and for letting me leave the nest.

Finally, I want to thank the person who truly walked this PhD journey with me, day by day. Thank you, Gio, my friend and my partner. You were by my side through every part of this journey, the tough moments and the happy ones. Over these three years, we learned how to stay close even when separated by hundreds of kilometers, we built up our relationship and support each other through everything. Thank you for providing me stability, a point of reference when everything else seemed to crumble.

And thank you all, once again, without you this PhD journey would have been far more difficult and grey!