Anomaly detection with Data Quality Early Warning Systems in ATLAS

Supervisor:

Prof. Sahal Yacoob

Co-supervisors:

Dr. Katharine Leney

Dr. James Catmore

Dr. James Keaveney

Author:

Senzo Msutwana

October, 2023

A dissertation submitted in fulfilment of the requirements for the degree of

Master of Science

in the

UCT-ATLAS Group

Department of Physics

Faculty of Science

University of Cape Town

Abstract

In this dissertation, the implementation of a Data-Quality Early Warning System (DQEWS) is explored. We use unsupervised Machine Learning (ML) methods to evaluate Data-Quality (DQ) in the ATLAS detector. We do so by observing and quantifying the evolution of Luminosity-Block (LB) data from Inner Detector (ID) tracking information, with a single LB towards the beginning of a run used as the reference. In this way, we obtain a trajectory that describes how the recorded LB data drift over the course of a run. Within the scope of this project thus far, the following will be shown. The version of the DQEWS algorithm defined as of the presentation of the results shown in this dissertation is shown to sufficiently flag good LBs as 'good', and bad LBs as 'bad' under the condition that the flagging criteria are evaluated on LB datasets that lie within a similar range of instantaneous luminosity as the LB datasets used to construct the criteria.

Acknowledgments

To my late father. Ndiyakukhumbula yonke imihla. Undifunde ukuba indoda eqinile. Ngoku ndizoqina xa ndikukhumbula. Le dissertation ayingowam, iyayakho. Lala namandla, tata.

To my main supervisor, Sahal. Thank you for all the conversations we've had, both about work and its complement. Your pragmatic approach to thinking has taught me to better streamline my own thoughts. I'll carry the lessons you've taught me, in practice, during my MSc, and will likely keep teaching me, in spirit, for the rest of my life. Thank you for this experience and for giving me the opportunity to show that I was capable of doing this.

To my co-supervisors. Thank you all for the various bits of assistance you've given me over the course of my MSc. I appreciate the help that you were all able to give.

To all my friends. Shout out to all of you. Too many to name individually but if you ever read this just know that I am, in fact, talking directly to you.

A special mention to my brother, Bob, for being there in ways that no one else could. I have an unending appreciation for the person you are and keep becoming. I can only hope that I ignite a similar feeling in you.

*

Abbreviations

AUC - Area Under the Curve

BCM - Beam Condition Monitor

BDT - Boosted Decision Tree

BSM - Beyond the Standard Model

CSC - Cathode Strip Chambers

CERN - European Organization for Nuclear Research

CM - Confusion Matrix

CTP - Central Trigger Processor

DAQ - Data Acquisition

DQ - Data Quality

DoF = Degree of Freedom

DQMD - Data Quality Monitoring Display

DQMF - Data Quality Monitoring Framework

DQEWS - Data Quality Early Warning System

ECal - Electromagnetic Calorimeter

EM - Electromagnetic

FCal - Forward Calorimeter

FN - False Negative

FP - False Positive

GBDT - Gradient Boosted Decision Tree

GRL - Good Runs Lists

HCal - Hadronic Calorimeter

HEP - High Energy Physics

HLT - High Level Trigger

ID - Inner Detector

IOV - Interval Of Validity

JER - Jet Energy Resolution

L1 - Level 1

LAr - Liquid-Argon

LB - Luminosity Block

LBN - Luminosity Block Number

LHC - Large Hadron Collider

MET - Missing transverse momentum

MBTS - Minimum Bians Trigger Scinitilator

MDT - Monitored Drift Tube

MS - Muon Spectrometer

QFT - Quantum Field Theory

ROC - Receiver Operating Characteristic

ROI - Region Of Interest

RPC - Resistive Plate Chamber

SCT - Silicon Tracker

SM - Standard Model

TDAQ - Trigger and Data Acquisition

TGC - Thin Gap Chambers

TN - True Negative

TP - True Positive

TRT - Transition Radiation Tracker

TTC - Timing Trigger and Control

ZDC - Zero Degree Calorimeter

List of Figures

List of Tables

Chapter 1

Introduction

The Standard Model (SM) of particle physics is a theoretical model which is the current best description of fundamental particles and their interactions. The particles which represent the basic constituents of matter are fermions, and those which mediate their interactions are the interaction particles known as vector bosons. Four fundamental forces (each described by an interaction particle) are postulated in nature namely the gravitational, electromagnetic, weak and strong forces. For this project, the latter three are relevant because the SM currently does not incorporate gravitational interactions. Four vector bosons exist that correspond to the three relevant forces. The electromagnetic force is mediated by photons, the weak force by the W and Z bosons, and the strong force by gluons. Additionally, the SM explains the origin of the fundamental particle masses through their interaction with a scalar boson called the Higgs boson [1].

Many predictions of the SM have been validated by various experiments around the world. For this reason, the SM is one of the most successful theories in physics. One such experiment which has the technological capability of validating theoretical predictions of the SM is the ATLAS experiment [2]. So far, observations made at ATLAS mostly conform to the SM, with one of the existing imperatives being to take more precise measurements of physical objects which exist within the experimental framework that aims to understand the SM interactions and particles, and to extend our understanding Beyond the Standard Model (BSM) as it currently stands.

In experiment, the process is such that the colliding particles are considered the initial state and primary vertex. All resultant decay products from this initial state present within the detector sub-systems prior to the measurement by the trigger menu are the intermediate states, and the final state particles are those which are measured directly by the detectors and selected for storage and analysis.

Experiments which aim to prove various theoretical predictions of the SM require the acquisition of data to be as precise as possible. In testing these theoretical predictions, measurements taken must be precise in that the uncertainties of measurements are reduced as much as possible.

Data are acquired with the Trigger and Data Acquisitions system (TDAQ) which is a sub-system in the ATLAS detector. The LHC provides data at rates that are too high for the ATLAS detector to read-out and store. Luckily most proton-proton collisions are not of interest and can be discarded. The trigger system makes a quick decision about whether or not an event is interesting and should be read out. Following a decision to read out an event by the trigger the data-acquisition process ensures that the many subsystems of the ATLAS detector correctly read out and store the correct information. The trigger in run 2 of the LHC is comprised of two independent levels, namely a hardware-based trigger (L1), and a software-based High-Level Trigger (HLT) [3].

In this dissertation, we present the Data Quality Early Warning System (DQEWS) which is designed to evaluate the validity of acquired information using Machine Learning (ML). Evaluation of information by the DQEWS is done on data having passed through L1 and accepted into the HLT using various tools to ensure that the acquired data do not contain unexpectedly different information (defects) from good data, and that if data is bad, the DQEWS can flag it correctly. We validate that the data do not have underlying distributions sufficiently different from others within a run of the detector in which the data are flagged as 'good' for physics analyses, where these differences of underlying distributions could be the results of things such as temporary electronics failures, detectors not being at the correct voltage, and mechanical failures of components in the detector. In particular, these defects denote situations where the detector conditions are not nominal within some interval of validity (IOV) [4]. The IOV is specifically a period during which the stored detector conditions are valid and applicable to the data taken. This period may be represented with either start and end timestamps (in ns) issued by the Central Trigger Processor (CTP). These time stamps are labelled with the Luminosity Block Number (LBN) [5]. In addition to this, there are other reasons for failure such as temporary electronics failures or detectors not being at the correct voltages etc. This can be identified, for example, by empty or zeroed out elements in the acquired datasets.

Regarding the structure of this dissertation, the following protocol will be followed. The relevant theory regarding the Standard Model, CERN and the Large Hadron Collider, the ATLAS detector, Data Acquisition and quality, and ML will be presented in that order. After this, the analyses will be presented.

The tools with which Data Quality (DQ) will be analysed will be ML-based with the eventual goal of monitoring DQ at ATLAS in real-time. This will be done by scanning reconstructed data Luminosity Block (LB) by LB and alerting shifters if the data contains any anomalous information in any of the scanned LBs. Anomalous information in this scope is any information that compromises the expected trajectory of changes in the underlying distributions between each LB for each considered run. Anomalous information may be detected in data at more than one level of analysis [6]. In this work I consider at tracking information from the Inner Detector (ID) that is acquired and stored.

The tools used to construct the DQEWS are binary classifiers. The particular type of binary classifier we use in this implementation of the DQEWS is the Gradient Boosted Decision Tree (GBDT). The idea is that this algorithm is used to quantify the separation between individual LB datasets with each instance of a pair-wise comparison being termed a LB comparison. With the set of LB comparisons, we then obtain fits of the trajectory of separation within a run. Additionally, we may calculate the variable attributions to determine whether variables attributed by the GBDT contained in LB datasets flagged as 'bad' by the Good Runs Lists (GRLs) provided by the DQ group are consistent with the variables that are flagged by the DQEWS. With the highest contributing, and potentially anomalous, variables computed by the DQEWS we have strong indications of locations of, and non-mechanical reasons for, failure within the ID sub-system that this anomalous variable comes from, and a time at which the failure occurred based on the LBN identifier.

The anomalous variables which we aim to detect emerge within a dynamically changing environment at the ATLAS detector during a particular run. Thus, the measurements taken are subject to effects resultant from the proton-proton collisions, as well as other effects which affect observations. These effects include but are not limited to pile-up, and background noise. Due to these sorts of effects, the underlying distributions of the acquired datasets are temporally subject to changes due to drifting variables which form a basis of the space of measurements taken. The set of all the changes that occur from all the comparisons is descriptive of drift occurring over the course of a run. By quantifying this drift, it is then possible to flag drifting variables as those most degrading the DQ of acquired information over the course of a run. The DQ is an integral consideration at ATLAS (and other experiments) as all physics measurements are taken from acquired data.

The analyses performed in this thesis make use of data collected from Run-2 of the ATLAS detector with proton-proton collisions with a centre-of-mass energy of up to $\sqrt{s}$ = 13 TeV.

2

The Standard Model of Particle Physics

This chapter motivates our need to be able to detect anomalous features in data that we analyse in our attempts to further understand the physics of our reality.

The standard model of particle physics provides the current most accurate description of our reality through the analysis of its fundamental constituents. These constituents are the particles defined at a fundamental level by Quantum Field Theory (QFT). QFT provides a unified framework for describing fundamental particles through the integration of relativity and quantum mechanics [7].

The dynamics of the system are determined using a slightly modified version of the action principle, S, and imposing the condition of zero variation of the action, δS with boundary conditions giving us the Euler-Lagrange equations of motion. This is done in this way because it allows us to generalise the dynamics of point particles over time to fields.

In QFT, there are two fields, each of which corresponds to the matter particles, and the force particles [7]. The fields are the solutions to Schrodingers equation when not considering relativity, and to the Klein-Gordon equation when considering relativity. Real valued fields, when quantised, generate wave-functions that represent free particles without charge while complex valued fields, when quantised, generate wave-functions that represent free particles with charge. The particles are represented in tabular form as shown in figure 2.1.

2.0.1 Fermions

The matter particles are half-integer spin particles called fermions. The fermions consist of two subsets each of which contains a collection of a type of particle. Each of these fermions has an antimatter counterpart which carries equivalent mass but opposite electric charge.

The first is the set of quarks which are particles that interact via the electroweak and strong forces. There are three generations of quarks which consist of the pairs: up and down, charm and strange, and top and

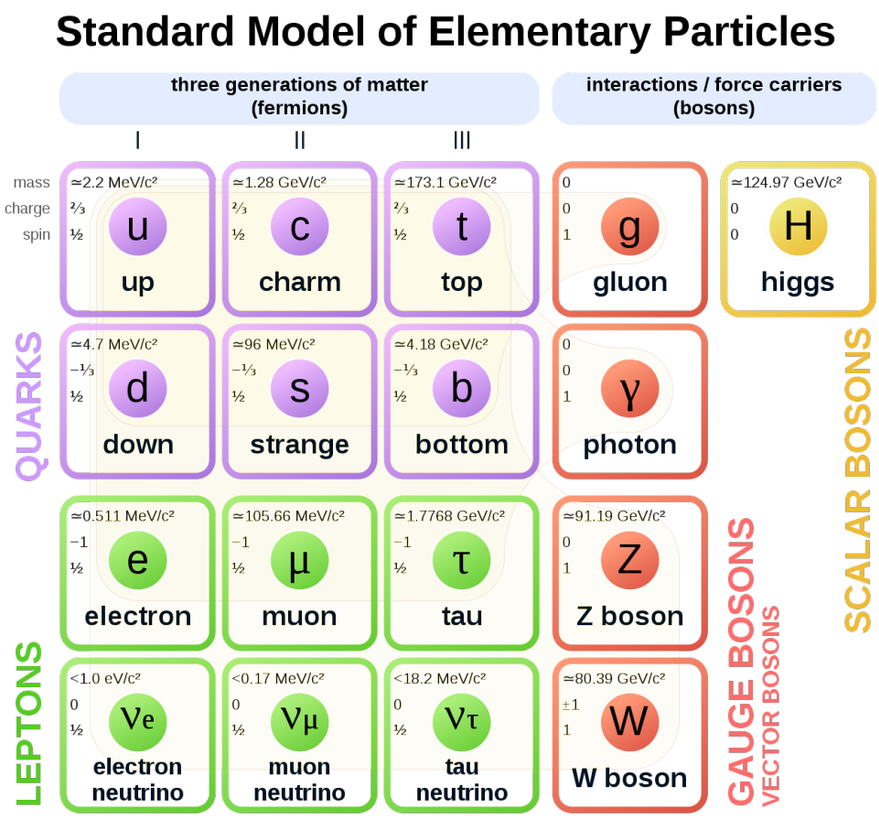


Figure 2.1: This figure shows the particles currently verified under the SM [8].

bottom.

The second is the set of leptons, which are also half-integer spin matter particles that interact via the electroweak force. There are also three generations of leptons which consist of the electron, muon and tau, along with their neutrino counterparts. [9]

2.0.2 Bosons

The force-carrying particles are integer spin particles. These particles are called the vector bosons. These vector bosons each mediate one of three fundamental forces. Namely, the electromagnetic, the weak, and the strong forces. Note that the gravitational force is not described by the SM. Four types of vector bosons exist for the three relevant forces in the SM. The electromagnetic force is mediated by photons, the weak force by the W and Z bosons, and the strong force by gluons.

An additional zero spin scalar boson exists called the Higgs-Boson. The Higgs boson has zero spin, zero charge and zero colour. It is highly unstable and as a result, decays into other particles almost instantaneously. It is responsible for the mass of the W and Z vector bosons by it's association with what is called the Higgs field. This Higgs field is what is responsible for the masses of all particles, including the Higgs boson, by their coupling to this field. The mass is acquired by a spontaneous symmetry-

breaking mechanism which is responsible for the near-instantaneous decay [10]. It is noted that there may be more than one candidate particles for the Higgs boson.

Photons and gluons are massless because they do not interact with the Higgs field.

2.1 Predictions

Consistency of SM theory with measurements taken by detector systems establishes an approach to a more consistent understanding of reality. It is through the measurement of the physical processes in such a way that theory and experiment converge, possibly leading to technological advancement, that it is possible to justify the high operational costs of these experiments to governments. This is in the sense that the governments that fund large experiments that employ thousands of scientists need to be incentivised to maintain this funding. These incentives include but are not necessarily limited to the possibility of creating new technologies through the discovery of new knowledge, and the training of individuals able to make use of the cutting edge technologies required to do these tasks entering their nations' work-forces. This is important because without this funding, the progress of our understanding of particle physics, and resultantly the SM, will come to rely only on private individuals that are able to do expensive High-Energy Physics (HEP) research without governmental funding. This would lead to an exodus of trained scientists into fields like software development and consulting should the private funding be insufficient to compensate scientists with living wages. This may, for some time, lead to a slowing of the rate of development within the field until such a time that these private individuals can fund the compensation of these scientists needed to reclaim the current rate of advancement within this field. For a reason such as this, predictions that may lead to discoveries are helpful to the rate of progress of our understanding of reality.

One such discovery is the famous Higgs-Boson particle. While no new technology has yet become of the discovery, its measurement validates prior theory. This principle of using experiment to validate theory is akin to painting the bulls-eye first then shooting the arrow, rather than the other way around. This is in the sense that in the latter case, by painting the bulls-eye after the arrow pierces some surface, one may analogously be considered as first configuring some piece of machinery for no particular problem and then obtaining the best solution to a particular problem using that machinery. It is not disputed that this methodology may work, but it does not include an initial purpose. It is this initial purpose that politicians may look at and see confirmation, through experiment, of theories that may make it easier to justify the continued operation of the facilities that perform these experiments. This approach ensures that there is a tangible utility in scientific research in the form of technological advancement.

2.2 Beyond the Standard Model

The SM is currently an incomplete system of physics [11]. The compulsion to continue to theorise new physics stems from this incompletion defined by the inability of the SM to describe all of what we can indirectly observe through experiment. This is the search for Beyond Standard Model (BSM) physics. Numerous theories aim to explain beyond standard model (BSM) physics. There are Grand Unified Theories, Technicolor, String Theory, and more recently, the Computational Universe [12, 13].

Evidence for BSM physics is currently observational in the sense that measurements have not directly been taken but suggestions of various phenomena exist through discrepancies seen in what is expected by theory. There are numerous examples from models that contain data that the SM cannot describe [14].

One such example is that we are currently unable to account for a large portion of the matter in our universe. Estimates of the amount of mass we are unable to account for are generally approximately 70% of the total mass density in our universe. This unaccounted-for matter has been suggested by various astrophysical observations that would be impossible without including this unaccounted-for matter. This unaccounted-for matter is chargeless, colourless, non-baryonic but is massive [14]. The term for this unaccounted for matter is Dark Matter (DM). The goal of the search for BSM physics is to find experimental evidence for candidate particles described by the theories mentioned above. Achieving this goal requires that we be able to theoretically describe candidate particles (or possibly a set of particles), and be able to construct experiments that can take precise and repeatable measurements of DM.

A necessary condition for candidate particles for DM is that they have an extremely small interaction with the three fundamental forces of the SM in order to be compatible with existing observations. Experimental searches for BSM physics, can be targeted to a specific effect predicted by a particular model, or designed to broadly search for deviations from the Standard Model expectations, which can be targeted for detailed study. It is important that when effects beyond the SM are measured, that the system taking these measurements is robust to errors that may mistakenly be seen as evidence, and that failures within the measurement system be swiftly identified. To ensure such robustness, a measurement system's ability to detect both anomalous acquired information and anomalous self-diagnostic information is important.

3

CERN and the LHC

3.1 CERN overview

CERN is the European Organization for Nuclear Research. It is an international research organisation whose initial goal was to represent peaceful world co-operation at the forefront of science. In the context of the era of the organisations' inception, this makes sense as it shortly followed the end of the second world war when our world was starved of peace. [15]

CERN was founded in 1954. It provided, and continues to provide, the infrastructure and resources needed to initiate science research that may guide humanity forwards by advancing the frontiers of possible engineering, and technological applications in general. It was founded by 12 nations.[15]

As of today, CERN has 22 member states, not necessarily within Europe's borders. These are states who contribute financially and are represented on the CERN council. Several other states are associate members, observers, or applicants for either full or associate membership. [15]

Primarily, CERN exists now for the same reasons as it did at the time of its founding. Open access has been culturally practised at CERN since its founding, achieved through using a mass mailing system. With the invention of the World Wide Web by Tim Berners Lee, while he was working at CERN, more efficient distribution of information was possible. CERN has an Open Access Policy that continues to reflect these values. This policy communicates that authors publishing as affiliates of CERN are required to publish their peer-reviewed primary research articles with open access [16].

In addition to being a bastion for sharing and improving knowledge, CERN provides the infrastructure needed for high-energy particle physics research. There are various accelerators at CERN which are collectively referred to as the accelerator complex, shown in figure 3.1.

This complex is a sequence of machines that can accelerate particles to increasingly higher energies. Each machine injects the particle beam into the next one. The Large Hadron Collider (LHC) is the last

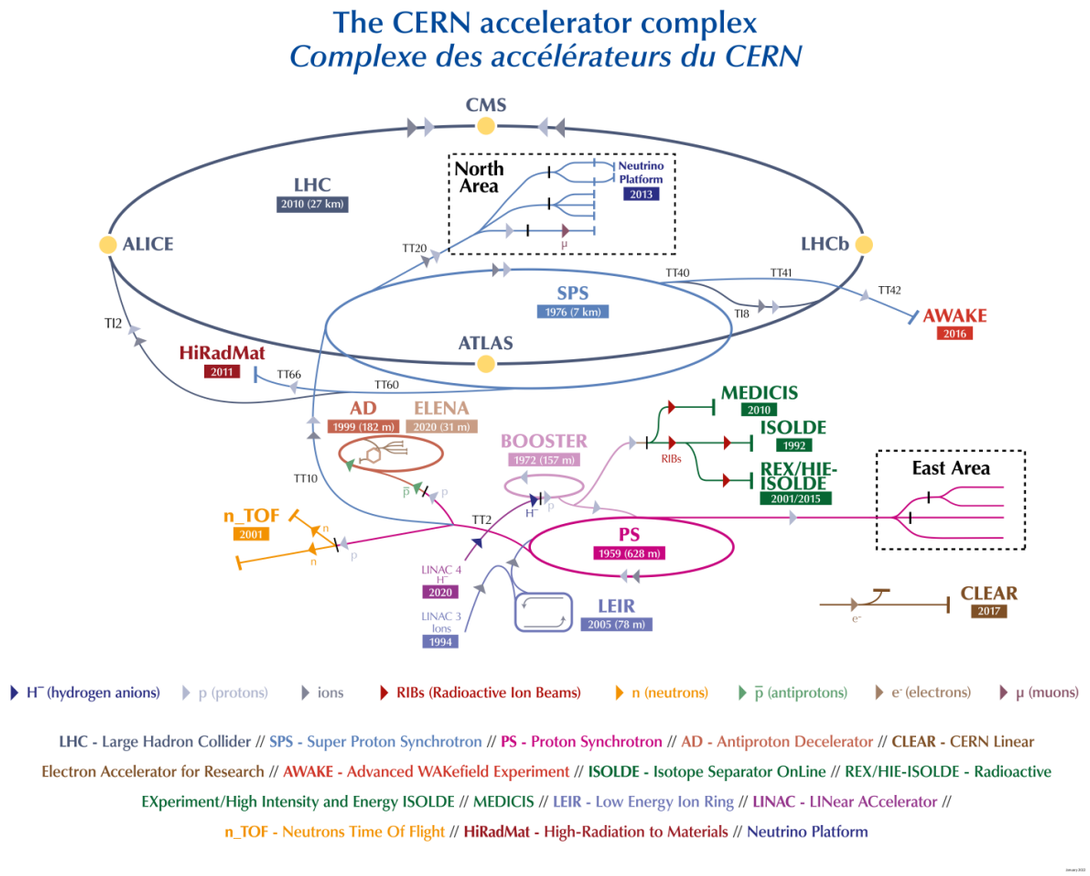


Figure 3.1: The CERN accelerator complex with the various accelerators and experiments are shown [17].

in this sequence of machines [18].

The LHC houses various experiments which aim to take measurements of different processes with the overarching goal of us furthering our collective physical understanding of reality. It is the largest, and most powerful, particle accelerator on Earth.

3.2 LHC overview

The LHC project was approved for construction by the CERN council in December 1994 and was built during a period between 1998 - 2008. As a result of its construction, the world was ushered into an age in which we saw an increase in the possibility of probing fundamental material structure for the proposed Higgs-Boson (which has, at the time of the writing of this dissertation, been verified), and the postulated range of particles predicted by theories of super-symmetry. [19]

The LHC has infrastructure for four major experiments. Each of these experiments occupies a particular location along the circumference of the LHC ring, each known as an interaction point. The four experiments are ALICE, ATLAS, CMS and the LHCb experiments. Each of these experiments focuses on

some form of physics study such as the testing of theoretical physics models, like the SM, and are capable of probing for possible candidates for BSM particles like the postulated dark matter. The ATLAS and CMS experiments are high luminosity experiments in search of dark matter [20] and other things such as identifying the difference between matter and antimatter, and the reasons why subatomic particles have such different masses [21, 22].

ATLAS and CMS are the largest of the experiments which both house general-purpose detectors [2]. ALICE is specifically a heavy-ion experiment in which Pb-Pb interactions that produce quark-gluon plasma are analysed, and LHC-B is an experiment dedicated to the study of charge-parity violations and other phenomena regarding the decay of beauty particles [23].

3.2.1 Global Layout

The LHC particle accelerator is contained in a circular tunnel with a circumference of 27 km which passes through both France and Switzerland. The tunnel is submerged at a depth ranging from 50 - 179 m underground. The cross-sectional diameter of the tunnel is 3.8 m and is lined with concrete. This tunnel contains two adjacent and parallel rings which each accelerate particle beams in opposite directions. These rings are pictured below with locations of the four experiments shown.

Each of the four main experiments is located at a pit shown by a blue star in figure 3.2. These pits are the locations at which the parallel rings intersect and are the interaction points of the accelerated particles.

3.3 Operation and Luminosity

A run of the detector is a period during which the LHC is operational and any of the four experiments can acquire data. During a run of the LHC, beams of proton bunches [1] are accelerated and guided by magnets up to the centre-of-mass collision energy of that run. These proton bunches then collide at four interaction points along the ring of the LHC. These points each correspond to the centre point of one of the four detectors.

Of all the proton-proton bunches that are accelerated, not all of them will interact through inelastic scattering, to produce measurable processes. The experiments are interested in only the physical processes that result from proton-proton inelastic scattering interactions. The interactions that produce these processes are those that are then measured by one of the detector systems.

The instantaneous luminosity for a colliding bunch pair is a parameter which is a measure of the rate of

[1] There are about 10^{11} protons per bunch [25]

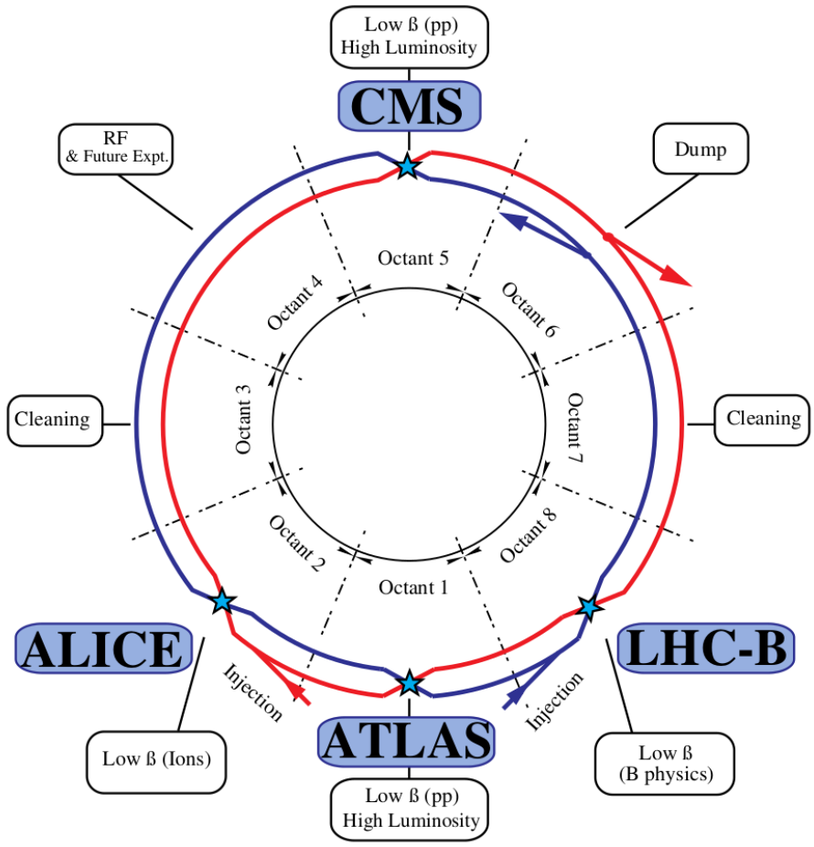


Figure 3.2: Illustration of the LHC ring showing locations of ALICE, ATLAS, CMS and LHC-B experiments [24].

interactions that take place for some physical process per the processes cross section during a period of time contained within the total duration of a run of the detector. It is mathematically defined as:

$$\mathcal{L} = \frac{R_{inel}}{\sigma_{inel}} \tag{3.1}$$

where $\mathcal{L}$ is the instantaneous luminosity, R_{inel} is the interaction rate of the physical process in question, and σ_{inel} is the p-p inelastic cross-section.

A collider operating at some revolution frequency f_r with n_b proton-proton bunches crossing at an interaction point, equation (3.1) may be re-written as

$$\mathcal{L} = \frac{\mu \cdot n_b \cdot f_r}{\sigma_{inel}} \tag{3.2}$$

where μ is the average number of inelastic interactions per bunch crossing [26].

The peak design centre-of-mass collision energy [2] and design luminosity [3] are 14 TeV and 10^{34} cm^{-1} respectively [27]. The peak instantaneous luminosity was 19 x 10^{33} cm^{-1} [28].

Run-1 of the LHC occurred between the years 2010 and 2012. During Run-1 the LHC was operated at peak beam energies of 4 TeV with the corresponding centre-of-mass energy being 8 TeV [27].

Run-2 of the LHC occurred between the years 2015 and 2018. During Run-2 the LHC was operated at peak beam energies 6.5 TeV with the corresponding centre-of-mass energy being 13 TeV [29].

3.3.1 Luminosity Decay

Luminosity decays over the course of a run due to degradation of intensities and emittances of the of circulating beams. The primary source of this degradation while the LHC is operational is the beam loss due to collisions [30]. The initial decay time of the colliding beams is:

$$\tau_{\text{nuc}} = \frac{N_{tot,t_0}}{L \cdot \sigma_{tot} \cdot k} \tag{3.3}$$

where N_{tot,t_0} is the initial beam intensity, L is the initial luminosity, σ_{tot} is the total beam cross-section and k is the number of interaction points of the colliding beams. This initial decay time shows the direct proportionality between the beam intensity and the luminosity of the collider

As functions of time, the beam intensity and luminosity due to only the effects of beam loss due to collisions are given by:

$$\mathcal{N}(\text{t}) = \frac{N_{tot,t_0}}{1 + \frac{t}{\tau_{nuc}}} \tag{3.4}$$

$$\mathcal{L}(\text{t}) = \frac{L_0}{(1 + \frac{t}{\tau_{nuc}})^2} \tag{3.5}$$

where the time required to reach $\frac{1}{e}$ of the initial luminosity is given by:

[2] This is the beam energy that the LHC machine is designed to be able to deliver. The beam energy has not been reached during a run.

[3] This is luminosity that the LHC machine is designed to be able to deliver. The design luminosity has been exceeded by a factor of two

$$\tau_{1/e} = (\sqrt{e} - 1)\tau \tag{3.6}$$

Assuming the peak LHC luminosity, L = 10^{34} cm^{-1} and the peak total colliding beam cross-section at a center-of-mass energy of 14 TeV, σ_{tot} = 10^{25} cm^{-2} equation (3.6) yields:

$$\tau = 29 hours \tag{3.7}$$

In addition to the above, we have other effects that result in beam loss such as Toucheck scattering [4] and slow emittance blow-up [5] due to things such as beam-beam interactions [30].

The total luminosity lifetime may be estimated as a sum due to contributing processes:

$$\frac{1}{\tau_{tot}} = \frac{1}{\tau_{nuc,\frac{1}{e}}} + \frac{1}{\tau_{process1}} + \frac{1}{\tau_{process2}} + ... \tag{3.8}$$

The decay is further approximated as an exponential process so that the contribution of the different processes to the decay may be easily added. This gives us the following [25] :

$$\mathcal{L}(\mathrm{t}) = L_0 exp(-\frac{t}{\tau_{tot}}) \tag{3.9}$$

3.3.2 Integrated Luminosity

While the instantaneous luminosity is important for determining the number of events measured by the detector during a particular period contained within a run, the integrated luminosity is important as it directly correlates to the total number of observed events over the course of a run [25]. We integrate equation (3.9) over the duration of a run to obtain [30]:

$$\mathcal{L}_{\mathrm{int}} = L_0 \tau_{tot}(1 - exp(-\frac{T_{run}}{\tau_{tot}}) \tag{3.10}$$

[4] This is scattering of particles that occurs within the rings of the LHC itself.

[5] Emittance is the average spread of a particle within the collider. The emittance blow-up is the tendency of particles to begin to travel in a more cloudy beam as opposed to a precise and streamlined beam

where T_{run} is the duration of a run.

4

ATLAS

The detector used by the ATLAS experiment is a multipurpose particle detector at the Large Hadron Collider (LHC).

At the nominal interaction point of proton bunches, 10 ^{11} protons will collide such that the collisions, at their peak, are estimated to occur at 14 TeV with a design luminosity of 10^{34} cm^{-2} s^{-1} [2].

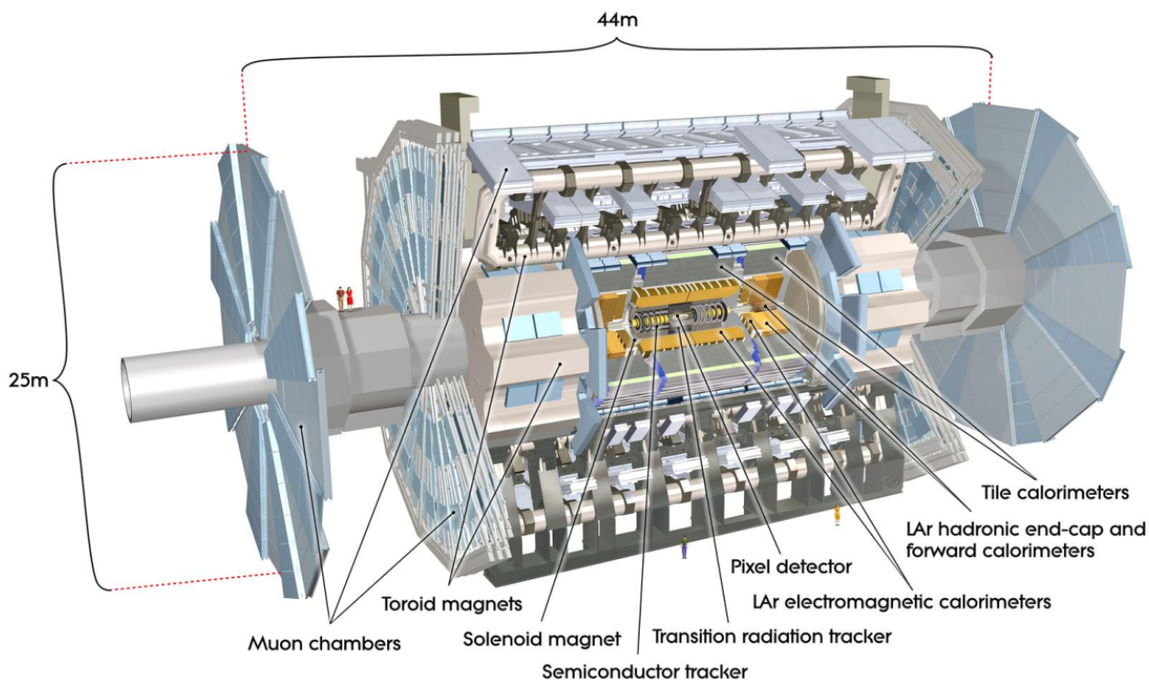


Figure 4.1: The ATLAS detector is shown above. Its total length is 46m, with a total weight of approximately 7000 tons [2].

4.1 Co-ordinate System

The global reference frame of the detector system is a right-handed Cartesian coordinate system with the origin point corresponding to both the nominal interaction point of the proton-proton collisions and the centre of the detector.

The positive x-axis points to the centre of the LHC ring, the positive y-axis points upwards towards both the surface of the earth and the sky, and the positive z-axis points anti-clockwise along the beam direction. The transverse plane which has its basis vectors as the x and y axes is represented with polar coordinates (r, ϕ) with r being the radial distance from the z-axis and ϕ being the azimuthal angle about the z-axis. The pseudo-rapidity (commonly referred to as the eta and shown in figure 4.3) is defined in terms of a polar angle, θ, as η = -ln(tan(θ/2)) where θ is the polar angle between the particle 3-momentum and the z-axis.

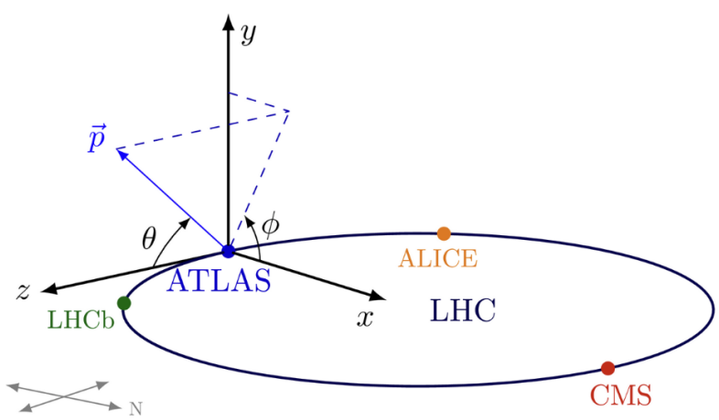


Figure 4.2: The coordinate system for the ATLAS detector is shown [31].

4.2 Sub-Detectors

The detector is comprised of three main sub-detectors that each performs a particular set of tasks for the acquisition of information required to ultimately perform physics analyses. These sub-detectors are the Inner Detector (ID), the Electromagnetic Calorimeter (ECal) and Hadronic Calorimeter (HCal), and the Muon Spectrometer (MS). These are complemented by a magnetic system that consists of four superconducting magnets whose positions in the detector are illustrated in figure 4.1. This magnetic system is comprised of a central solenoid that provides a magnetic field of 2 T for the ID [33], a barrel toroid, and two end-cap toroids that generate 1 T magnetic fields for the MS [34].

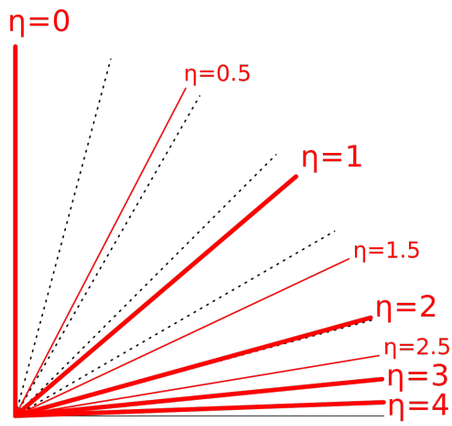


Figure 4.3: A polar plot illustrating the pseudo-rapidity in the ATLAS detector coordinate system is shown [32].

We provide descriptions of all these components but provide a more comprehensive description of the ID as all data used for the content of the analyses in this dissertation contain only ID tracking variables. The sub-detectors and their components descriptions will be mentioned from those closest to the proton-proton interaction point, outwards.

Descriptions of the sub-detector components are given as context for the conditions that are descriptive of the measurement system during operation, and how changes in these conditions may result in a degradation of the quality of the recorded data.

4.2.1 Inner Detector and Track Reconstruction

To reconstruct and identify particles that result from the proton-proton collisions, ATLAS uses the ID. The ID provides essential information for the reconstruction of physics objects such as electrons, muons, τ-leptons, photons and jets. In addition, it provides information for the identification of jets containing b-hadrons and for event-level quantities calculated using charged particle tracks [4].

The design of the ID aims to provide as efficient, precise and robust position and momentum measurements, and identification of traversing charged particles as possible. These charged particles induce electrical charge signals in the IDs components which are then read-out such that we may determine the positions of the hits based on our understanding of the spatial configuration of the ID components. These electrical charge signal measurements (these can also be referred to as hits) are then used to reconstruct their trajectories (these are the tracks) using the positions of the signals and to estimate the associated tracking variables [35]. Combining information about the electrical charges and tracks of these particles incident to the ID components, we can calculate the particle momenta.

The precision of the tracking variables is determined by factors including the resolution of measurement devices, the magnetic field, distribution of material in the ID, and the orientation of the detectors elements. Detector alignment is thus important to optimise this precision.

The ID is designed to have complete azimuthal coverage, and to have pseudo-rapidity coverage in the region $|\eta| < 2.5$. In addition, it has been designed to provide sufficient transverse momentum resolution [1].

It provides charged particle identification over $|\eta| < 2.0$ and for energies between 0.5 GeV and 150 GeV [2]. This range of energies is due to particles measured closer to the collision point having significantly higher energies than those further out within the ID [36].

The ID operates within a 2 Tesla magnetic field provided by the toroidal magnetic system [37]. The picture shown in figure 4.4 is a cross-sectional image to help illustrate the relative position of the sub-detector components. On the right side of the image, the black line represents an individual track traversing the ID. The green dots represent the intersection of the track with the surface of a sub-detector plane, the red stars indicate the sub-detector measurements, and the blue lines represent the residual of the track with the sub-detector hit[38].

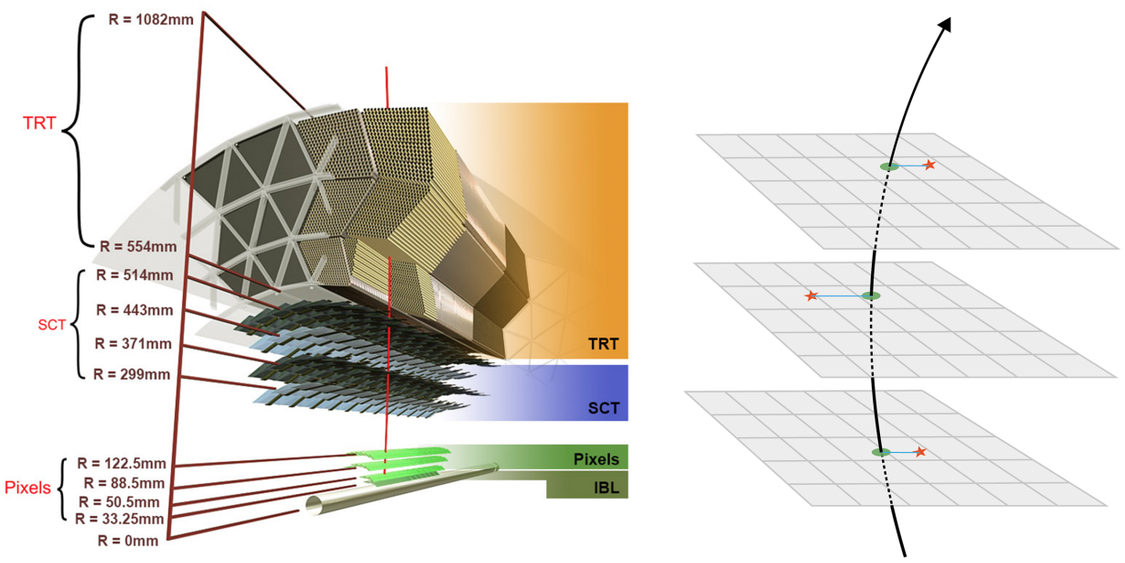


Figure 4.4: Cross-sectional image of the ID sub-detectors, the TRT, SCT and Pixel Detector, from top to bottom [38].

Sub-detectors

The ID detector consists of three sub-detectors, namely the silicon pixel detector, the silicon micro-strip (or SemiConductor) Tracker (SCT), and the straw-tube Transition Radiation Tracker (TRT). Descriptions

[1] In the transverse plane which is perpendicular to the beam axis

of these are as follows, and are all taken from [33]. These are described from the closest to the beam interaction point outwards:

1. The silicon pixel detector provides three measurement points for particles incident from the collision point. The Insertable B-Layer (IBL) is the innermost layer of the pixel detector which is there to cope with the increase in instantaneous luminosity between Run-1 and Run-2 of the LHC. The IBL has approximately 12 million pixels [39]. It then consists of 1744 silicon pixel modules arranged in three concentric barrel layers, and two end-caps of three disks each.

   The local x-coordinates of each module correspond to high-precision position measurements in the global rapidity-azimuthal plane. The local y-coordinates are oriented approximately along the global z-direction (beam axis) in the ID barrel, and along R in the end-caps. Each module is read out by 16 radiation-hard front-end chips bonded to the sensor. The total number of read-out channels is approximately 80.4 million. Pulses are read out if the signal from an incoming particle exceeds some threshold.

2. The SCT provides eight strip measurements (that correspond to four points in space) of particles incident from the collision point. It consists of 4088 modules of silicon-strip sensors arranged in four concentric barrel layers, and two end-caps of nine disks each. These strips are approximately parallel to the magnetic systems solenoid field and beam axis. Most of the modules consist of four silicon strip sensors; two on each side. The strips are, again, read out by radiation-hard front-end readout chips with each chip reading out 128 channels. The total number of read-out channels is approximately 6.3 million

3. The TRT consists of 298 304 drift tubes (straws) read out by 350 848 channels of electronics. The straws in the barrel region are arranged in cylindrical layers and 32 azimuthal sectors. They have split anodes and are read-out on each side. The straws in the end-cap region are radially oriented 80 wheel-like modular structures [40]. It is the TRT that provides the ID with electron identification via transition radiation [2] from polypropylene fibres or foils interleaved between the straws.

Shown below is an image to help illustrate the positions of the sub-detector components and their elements described in the above enumeration.

The ID takes measurements of the trajectories of the decay product particles from the collision event that are not neutrally charged. The ID being the detector component physically situated closest to the proton-proton collision vertex requires the ability to take high precision measurements with sufficiently fine detector granularity due to the higher particle densities close to the interaction region. These high

[2] Transition radiation is emitted electromagnetic radiation from charged particles passing through some heterogeneous media

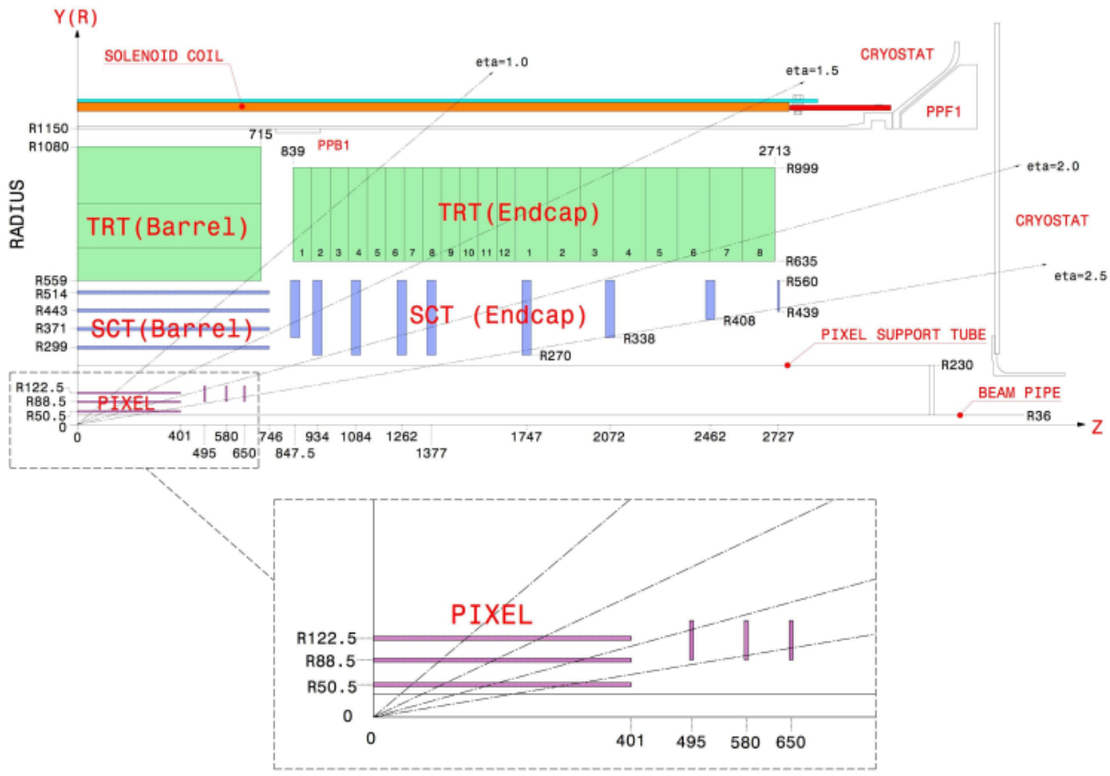


Figure 4.5: Cross-sectional image of the ID detector showing the positions of the sub-detectors elements relative to each other [33].

particle densities are considered to be dense, within this region , relative to the particle densities within regions further out from the beam interaction point in which measurements fall under the cover of the other detector components.

These high particle densities within the ID region strongly correlate with this region being a high radiation environment. This imposes strong conditions on its various components and their service over the sub-detectors 10-year design lifespan [2]. Over this lifespan, the silicon pixel layers must withstand an integrated fluence [3] of approximately $8x10^{14}n_{eq}/cm^2$ while inner parts of the SCT must withstand approximately $2x10^{14}n_{eq}/cm^2$ [2]. These silicon components need to be kept at temperatures between $-5°C$ and $-10°C$ (implying coolant temperatures of approximately $-25°C$) to maintain an adequate noise performance subsequent to radiation damage. The TRT, however, is designed to operate at room temperature.

Due to the above, it is imperative that radiation-tolerant materials are used, and that these materials do not undergo significant structural changes from deformations due to heat following cycles of temperatures between $-20°C$ and $20°C$ [2].

[3] Radiant energy received by some unit surface integrated with the time of irradiation. It is normalised using non-ionising energy loss (NIEL) cross-sections to the expected damage for 1 MeV neutrons.

All information is presented as context for the conditions that are required for nominal operation for the detector. Deviation from these conditions may lead to degradation of the Data Quality (DQ). It is important to continuously assess our understanding of the detector in order to maintain sufficient DQ. This is done by mitigating deleterious effects due to pile-up and by performing calibration checks to ensure the system's integrity.

Pile-up

The number of proton-proton interactions per bunch crossing follows a Poisson distribution. This allows us to interpret the information acquired using a Poisson counting process. The mean value of this distribution is given by μ.

Over the course of a run μ decreases with decreasing beam intensity and increasing emittance, The largest value of μ thus occurs towards the beginning of the run and is μ_{peak} [41].

Extending the logic from (4.1) we rearrange to obtain:

$$\mu = \frac{\mathcal{L} \cdot \sigma_{inel}}{n_b \cdot f_r} \tag{4.1}$$

The uncertainties of μ depend on the uncertainties of the measured $\mathcal{L}$ and σ_{inel}.

In-time pile-up is the impact from additional interactions within the same bunch crossing. Depending on the duration of the read-out window of the ID, signals from neighbouring (in time) bunch crossing can be present when the sub-detector is read out. The impact from these neighbouring bunch crossing is out-of-time pile-up. Due to the time-resolution of the ID, out-of-time pile-up has a smaller impact than in-time pile-up. Out-of-time pile-up slightly increases the track occupancy in the TRT (has a read-out window of 75 ns) in relation to the silicon detectors (have read-out windows of 25 ns). The number of reconstructed tracks and vertices directly corresponds to the amount of in-time pile-up on an event-by-event basis [41].

Track and Vertex Reconstruction

The ID is sensitive to an increase in particle multiplicity with pile-up. As seen by the ID components this is quantified by an increase in the number of channels that need to be read out (track occupancy) from each of its components. This is challenging for the DAQ of the events. The pixel detector naturally has the highest particle flux but has the lowest track occupancy due to the higher granularity of the modules. The track occupancy then increases. and granularity decreases going outwards to the TRT. The increase

in track occupancy makes track reconstruction challenging.

A single track from a particle would typically have 3 pixel clusters, 8 SCT clusters and 30 TRT hits.

Tracks for primary charged particles are constructed using a few algorithms that have different methods of associating tracking information from hits in individual sub-detector modules. A primary charged particle is defined as a particle whose mean lifetime is greater than 3×10^{-11} s and is directly produced by any of the proton-proton interaction, the subsequent decays of such particles, or the interaction of particles with a mean lifetime shorter than 3×10^{-11} s. The transverse momentum for a track to be reconstructed by the inside-out algorithm is required to be $p_T > 400$ MeV.

The inside-out algorithm is the baseline algorithm used to efficiently reconstruct primary charged particles. The algorithm is instantiated using 3-point seeds (obtained from pixel clusters) in space [4] that describe the measurements from the silicon detectors, and add hits moving away from the interaction point using a combinatorial Kalman filter [42]. Using these point seeds the Kalman filter predicts the most likely position which describes the next position in the trajectory of that particular particles' track and updates in that direction. The next prediction is then instantiated by the updated position of the next chosen measurement that aligns with the track according to the filter [43]. After ambiguities in the track candidates in this spatial region are resolved, the tracks are extended to the TRT.

Once the tracks are extended to the TRT, the Kalman filter is used in reverse and adds silicon hits. This is referred to as back-tracking. Back-tracking reconstructs secondary particles. Secondary particles are particles produced by interactions of primary particles. Tracks with a TRT segment but no back-tracked extension back into the silicon detectors are referred to as TRT-standalone tracks [41]. Outlier clusters are then removed along with the rejection of fake tracks. This is achieved using quality selection criteria using limits on things such as the number of holes [5] per track [2].

Issues with track reconstruction lead to degraded track variable resolution due to incorrect signal assignment, decreased efficiency and fake tracks being assigned. Increasing density of collisions with pile-up degrades vertex resolution when a track is included in this vertex, or two vertices are combined into a single reconstructed vertex [41].

Primary vertices are reconstructed by first using a primary vertex finding algorithm that associates reconstructed tracks to vertex candidates, then secondly using a vertex fitting algorithm that reconstructs the vertex position and its corresponding error matrix [44].

[4] These point seeds are 4-vectors

[5] A hole is defined as an active silicon sensor crossed by a track but not generating an associated cluster. By extension, this excludes known disabled sensors

The datasets used in the analyses of this dissertation contain the information of variables that describe this reconstructed tracking information.

Alignment

Alignment is required to provide an accurate description of the geometry of the detector to obtain precise locations and orientations of each of the tracking modules. Alignment of the ID is necessary for reaching sufficient tracking performance. This alignment is done to determine the correct spatial positions of the pixel and SCT silicon modules, and the TRT straw modules. In doing so, 6 degrees of freedom (rotational and translational) are determined for each module treated as rigid bodies in 3-d space. The total number of spatial degrees of freedom needing to be dealt with in alignment is 700 000 [45]. Additionally, it is important to correct for imperfections within each module due to temperature gradients, module bows, and other distortions[37].

The alignment of the detector elements is done using charged tracks, and for the SCT it is supplemented laser interferometric monitoring [6].

4.2.2 Calorimeters

The ATLAS calorimeter system is composed of two main types. These are the Liquid-Argon (LAr) electromagnetic calorimeter (EM), and the hadronic calorimeters. Each of these measures the particle type in its name. The calorimeter system provides coverage of $|\eta| < 4.9$.

The calorimeters are sampling calorimeters. This is that they take measurements from showers of energy produced by particles incident to the bulk of matter contained in the calorimeters using alternating layers of active and absorbing materials. Sampling calorimetry, as opposed to homogeneous calorimetry, is that alternating layers are used as opposed to a single (hence homogeneous) active layer being used.

The calorimeter system as a whole provides closure for electromagnetic and hadronic showers within the region containing its components, thus stopping the propagation of matter within these types of showers before they reach the muon spectrometry system [2].

The LAr EM calorimeter directly surrounds the ID. It is a high granularity LAr sampling calorimeter that uses lead as an absorber. The LAr EM calorimeter is further divided into a barrel module, which covers the region $|\eta| < 1.475$, and two end-cap modules which cover the region $1.375 < |\eta| < 3.2$ on each end

[6] Using monochromatic light split and propagated towards some surface on which the reflected light beams have their phase differences measured, thus providing a measurement of positional drift of components

of the barrel module with some overlap. These barrel and end-cap modules are integrated along with pre-samplers which are mounted in between the cryostat cold-wall and the aforementioned calorimeter modules. The barrel pre-sampler covers $|\eta| < 1.52$, and the end-cap pre-samplers cover $1.5 < |\eta| < 1.8$

The hadronic calorimeters are divided into three regions.

1. The first section is the central-most section which contains the barrel regions. This is further dived into three sub-systems. The central barrel region which covers $|\eta| < 0.8$, and the two extended barrel regions which each cover $0.8 < |\eta| < 1.7$. All of these regions contain scintillator-tile/steel hadronic calorimeters. Each of these barrel regions consists of 64 modules with azimuthal coverages of $\pi/32$ rad.

2. The second section is that which houses the hadronic end-cap calorimeters. This section covers the pseudo-rapidity range $1.5 < |\eta| < 3.2$ and modulates LAr/Cu[7] calorimeter modules.

3. The third section houses two Forward Calorimeters (FCals) at each end of the calorimetry system. These calorimeters modulate both LAr/Cu and LAr/W[8] modules for both electromagnetic and hadronic energy measurements respectively. [37]

Pictured in figure 4.6 is an image illustrating the configuration of the calorimetry system and its labelled components. It is noted that due to the accordion geometry of all the above-mentioned components, azimuthal coverage covers a full revolution in φ, and thus is not specified.

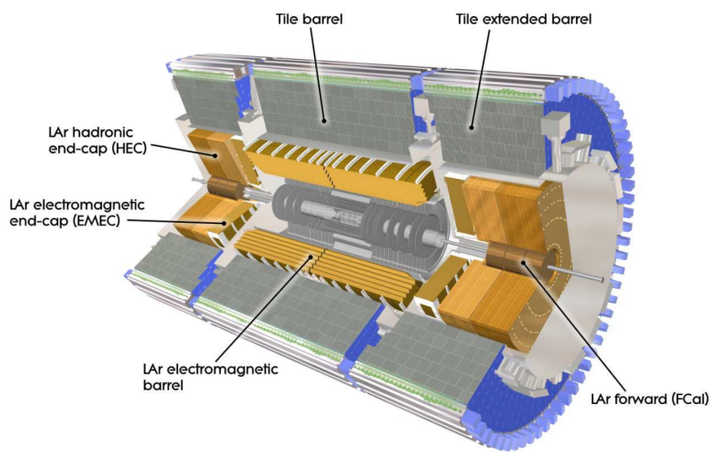


Figure 4.6: Cross-sectional image of the calorimeter detector showing the positions of the sub-detectors elements relative to each other [37].

Measurements are built into topological cell clusters using the spatial distribution of cell signals to allow for the reconstruction of energy and directions of incident particles by combining them with ID tracking information into jets [37].

[7]Liquid-Argon/Copper

[8]Liquid-Argon/Tungsten

Particle jet reconstruction is done by combining tracking information outlined in section 4.2.1 with topological clusters of calorimeter information using particle flow [46].

4.2.3 Muon Spectrometer

The muon spectrometer directly surrounds the calorimetry system. It is designed to take momentum measurements of muons with transverse momentum, $p_T > 3GeV$. The muon momentum is determined by measuring the curvature of muon tracks in a toroidal magnetic field.

These muon tracks fully penetrate (have no deposits in) the calorimeters, and enter the muon chamber. The trajectory of these tracks is normal to the main component of the magnetic field such that the transverse momentum is independent of pseudo-rapidity i.e has no non-zero components that are dependent on the polar angle of incidence.

The magnetic field is provided by three toroidal magnets, one in the central region (barrel), and one for each forward (end-cap)region. The pseudo-rapidity coverage for these regions are $|\eta| < 1.1$ and $1.1 < |\eta| < 2.7$ respectively with a field integral between 2 and 8 T·m [47].

The muon spectrometer is comprised of the following components

1. Monitored Drift Tube chambers (MDTs).
2. Resistive Plate Chambers (RPCs)
3. Thin Gap Chambers (TGCs)
4. Cathode Strip Chambers (CSCs)

MDTs are used for most of the acceptance range. The coordinate perpendicular to the direction of the toroidal field is measured by the MDT and is referred to as the precision (or bending coordinate)[47]. The MDTs are complemented by the RPCs and TGCs which are fast-trigger chambers with good time resolution. The RPCs trigger particles within the barrel region, and the TGCs trigger particles in the end-cap regions.

In a small end-cap region that surrounds the proton-proton beam pipe, the CSCs are used instead of the MDTs due to their ability to cope with higher particle multiplicities [48]. The configuration of these components is shown below in figure 4.7

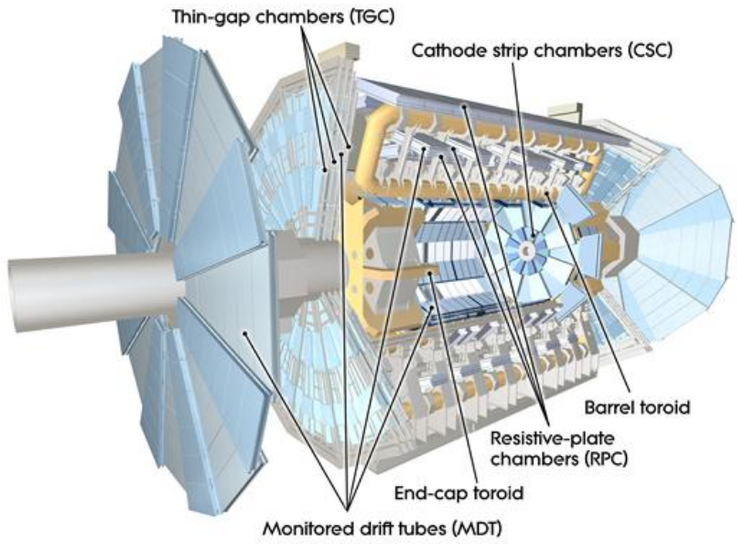


Figure 4.7: Cross-sectional image of the ATLAS detector showing the positions of the muon spectrometer components [37].

4.2.4 LUCID (LUminosity Cherenkov Integrating Detector)

Taking precise measurements of the absolute luminosity is important. In measuring the absolute luminosity, ATLAS assesses and controls systematic uncertainties that affect it. For these measurements to be precise, calibration is required. This calibration relies primarily on LUCID2 (which we also refer to as simply LUCID). LUCID is mentioned in particular as it is the detector that is dedicated to luminosity measurements with a coverage of $5.6 < |\eta| < 5.9$. These luminosity measurements are complemented by bunch-wise measurements from other detectors and are taken online[9].

4.3 Energy resolution

It is important that the jet energy and position resolution are known in order to make as precise and detailed measurements of SM particles that decay to jets as possible. The Jet Energy Resolution (JER) affects the missing transverse momentum (commonly referred to as missing transverse energy or simply MET). The MET is particle energy in the plane tranverse to the beam direction that is not detected by any of the sub-detectors but is expected due to energy and momentum conservation laws [49].

Measurements that involve particles that do not interact via the electromagnetic and strong forces, such as neutrinos and potentially Dark Matter candidate particles, rely quite heavily on well constructed MET. This is because the sub-detector components make use of particle interactions to take measurements.

[9]While the LHC is operational and measurements are being taken by ATLAS during proton-proton collisions

The JER depends on the transverse momentum (p_t) and is parameterized using a functional form of calorimeter based resolutions. There are three independent contributions to the JER which are [49, 50]:

1. N: Term due to the contribution from electronic noise to the signal by front-end electronics, as well as pile-up. This term is expected to dominate in the low p_T region, below 30 GeV
2. S: Term due to statistical fluctuations in the amount of energy deposited in calorimeter cells. The contribution is small for homogeneous calorimeters and larger for sampling calorimeters.
3. C: A constant term due to fluctuations that are a constant proportion of the p_T such as energy deposits in passive material (like solenoid coil), the initial point of the hadronic showers, and non-uniformities of response within the calorimeter. This term is expected to dominate in the high p_T region, above 400 GeV

The general form of the JER in terms of the parameters described above is:

$$\frac{\sigma(p_T)}{p_T} = \frac{N}{p_T} \bigoplus \frac{S}{\sqrt{p_T}} \bigoplus C \tag{4.2}$$

The JER is defined in terms of the p_T of the jet. Calculations done using this equation require that jets must either recoil against some reference object whose momentum can be precisely measured or be balanced with another jet in a dijet system.

In general, the energy resolution of massive objects may be written in a similar form to equation (4.8) as:

$$\frac{\sigma(E)}{E} = \frac{N}{E} \bigoplus \frac{S}{\sqrt{E}} \bigoplus C \tag{4.3}$$

4.3.1 Electromagnetic calorimeter resolution

The design energy resolution for the EM calorimeter is:

$$\frac{\sigma(E)}{E} = \frac{170MeV}{E} \bigoplus \frac{10\%}{\sqrt{E}} \bigoplus 0.7\% \tag{4.4}$$

4.3.2 Hadronic calorimeter resolution

The standalone energy resolution for the hadronic tile calorimeter obtained using test-beams using single pions is:

$$\frac{\sigma(E)}{E} = \frac{52.9\%}{\sqrt{E}} \bigoplus 5.7\% \tag{4.5}$$

The noise term is negligable, and is thus excluded.

4.3.3 Muon spectrometer resolution

The resolution of a signal produced by a charged particle passing through a MDT above 100 GeV becomes important, and above 250 GeV becomes a dominant contribution to the transverse momentum resolution calculation.

The MS design momentum resolution at 100 GeV, and at 1 TeV is given by equations (4.3.3) and (4.3.3) respectively:

$$\frac{\delta(p)}{p} = 3\%$$

(4.6)

$$\frac{\delta(p)}{p} = 10\%$$

(4.7)

At high momentum, the p_T resolution is determined by [51]:

$$\frac{\sigma(p_T)}{p_T} = \sigma_{res} \times p_T \tag{4.8}$$

Where σ_res is the single-hit spatial resolution. The design resolution is $\sigma_{res} = 80\mu m$[52]

4.4 Absolute Luminosity

A detector's particle identification capabilities are important due to the precision with which measurements need to be made. More specifically, in order to improve our ability to identify final state particles, like leptons or photons, or to be able to take measurements of experimental parameters like the missing transverse momentum, it is required that high luminosities be reached and that the detector has sufficient ability to take measurements at these luminosities.

The absolute luminosity is a quantity that represents the geometric overlap of each beam accelerated in the LHC that are visible to the ATLAS detector. It may be computed from parameters of the accelerator, and measurements made by the detector using:

$$\mathcal{L} = \frac{f_{LHC} \cdot \sum_{i=1}^{n_b} N_{i1} \cdot N_{i2}}{4\pi\sigma_x \cdot \sigma_y} \tag{4.9}$$

where f_{LHC} is the revolution frequency of the accelerator (the frequency with which some object will take to complete one revolution of the LCH), n_b is the number of colliding p-p bunches, N_{i1} and N_{i2} are the number of protons per bunch in beams 1 and 2, and σ_x and σ_y are the corresponding beam width and height for these aforementioned beams.

The sub-systems that have the capability for luminosity measurements, along with their respective coverages, are as follows [53, 54, 55, 56]:

1. LUCID - Cherenkov detector dedicated to relative luminosity measurements with $5.6 < |\eta| < 5.9$
2. EMEC - Endcap region of the Liquid-Argon Electromagnetic Calorimeter with $1.375 < |\eta| < 3.2$
3. FCal - Forward Calorimeter with $3.1 < |\eta| < 4.9$
4. TileCal - Scintillator-tile hadronic calorimeter which is divided into three cylinders with the long barrel covering $|\eta| < 1.0$, and two extended barrels covering $0.8 < |\eta| < 1.7$
5. HLT - High Level Trigger (will be discussed in more detail later on in this dissertation) with $|\eta| < 2.5$

Online measurements taken are complemented by offline measurements of reconstructed particles in various bunch crossings, which is in essence counting the tracks of the particles. [53]

4.4.1 Luminosity Blocks

Luminosity blocks (LBs) are the sets of information that describe all measurements taken within the lowest resolution time unit during which ATLAS luminosity data is being recorded. An LB corresponds to approximately 2 minutes of data taking and may be considered the atomic unit of data taking within ATLAS. At the end of a run, there will be many LBs available. This temporally ordered designation of measurements into LBs makes comparisons between these LBs from the sub-detectors within the same run possible so that we may calibrate the models used to assess the validity of the measurements contained within each LB. In addition to this, it allows us to track the behaviour of acquired LB information in a way that allows us to determine how the underlying distributions that describe these data change within a run.

4.5 Commissioning and Calibration

Commissioning and calibration of the ATLAS detector in preparation for the first run required that the following necessary steps were taken [33]:

- Operation of services and controls
- Calibration of the detector
- Synchronisation of all sub-detectors
- Measurement of efficiency and noise occupancy for each sub-detector during operation
- Testing reconstruction software and tracking triggers on data
- Perform alignment of the detector

The data used for alignment preliminary track reconstruction commission were taken from Cosmic-ray events. The events used mostly consisted of a muon traversing the full ATLAS detector. Muons are suitable due to their kinematics working well for specific measurements. These measurements include detector efficiency, tracking resolution, and detector response to ionisation as a function of the momentum and incident angle of particles [33]. This calibration is important as it quantifies our understanding of the detector system so that we may obtain the Data Quality of a system whose properties are known at the time of measurements being taken.

5

Data Acquisition and Quality

The ATLAS experiment makes use of its detector to record large amounts of physics events that are used for analysis. Within a year during a run of the LHC, several petabytes of data are produced with various physics conditions and data formats [57].

5.1 Event selection and triggering

The ability of the ATLAS trigger system to sufficiently process ID tracking information, with the goal of reconstructing particle trajectories in a way that the ID, calorimeter and MS form approximately integrable particle trajectories, is essential for the triggering of physics objects. These objects include but are not limited to electrons, muons and b-jets [58].

The proton beams in the LHC consist of proton bunches that pass by one another at a sufficient rate to allow individual protons within these bunches to collide once every 25 ns (the bunch spacing), where the length of a single bunch is about 1 ns. At a design instantaneous luminosity of 10^{34} $cm^{-2}s^{-1}$ with the aforementioned bunch spacing, the average number of interactions is approximately 23 per bunch-crossing. This is, 10^9 interactions per second [59].

Table 5.1 presents the LHC parameters obtained during runs 1 & 2 as of the end of Run-2 in 2018. The values obtained are from [60, 61].

Unique bunch-crossings and their resultant measured events must be associated to avoid background from collisions corresponding to other bunch-crossings. In theory, for the operation of a trigger satisfying these conditions within this environment, the analysis of event data must be done at a rate of at least 40 MHz to avoid dead time [59].

The increase in pile-up is attributed to the increase in luminosity. Measurements taken in an environment with more pile-up contain larger and more numerous events within a given period, and as a result, take

Parameter	Run-1	Run-2
Center-of-mass energy	7 TeV	13 TeV
Number of bunches	1400	2500
Bunch spacing	50 ns	25 ns
Peak luminosity	$7 \cdot 10^{33}\ cm^{-2}s^{-1}$	$2.1 \cdot 10^{34}\ cm^{-2}s^{-1}$
Peak pile-up	45	60
Event rate	20 MHz	40 MHz

Table 5.1: Table of peak LHC Parameters obtained in runs 1 and 2.

longer to process [62]. Due to this increase in data cardinality, it would be computationally infeasible to store all the data for all of these interactions and thus, selective data storage is required.

As a direct result of the increased data cardinality, recorded events take up more computer memory and require more time to process [63]. This increase in the computational demands on the systems designed to take event measurements results in greater care required to ensure that events are measured with sufficient event rate resolution and sensitivity to identify physics.

To achieve the acquisition of data with sufficient event rate resolution and sensitivity, ATLAS makes use of a multi-level selection process, and a Data Acquisition (DAQ) process with a hierarchical structure. The functional elements of the ATLAS DAQ system were comprised of three levels as deployed in Run-1, and of two levels as deployed in Run-2 [59].

5.1.1 ATLAS DAQ System Information Flow

The three event selection levels defined in Run-1 were [64]:

1. First level trigger (L1). Hardware-based. The event input rate was the required event rate of 20 MHz. The event rate acceptance of the L1 trigger system was a maximum of about 60 - 65 kHz.

2. Second level trigger (L2). Software-based analysis of events within Regions Of Interest (ROIs) marked by L1. The event input rate is the event acceptance rate of the L1 trigger. The maximum event rate acceptance of the L2 trigger system was 5-6 kHz.

3. Event filter. Software-based analysis of the full event. The maximum event rate acceptance of the Event Filter (EF) was about 600 Hz.

In Run-2, the trigger system consisted of two event selection levels. It used the low latency hardware-

based L1 system which is then followed by a higher-level software-based High-Level Trigger (HLT) system for a more detailed event reconstruction. The HLT in Run-2 incorporates the functionality of the L2 and Event filter systems in Run-1 into one sub-system.

These L1 and HLT systems are described as follows:

1. First level trigger (L1). Hardware-based. The event input rate was at the required event rate of 40 MHz. The event rate acceptance of the L1 trigger system was a maximum of about 100 kHz.

2. High-Level Trigger (HLT). Software-based. The event input rate is the event acceptance rate of the L1 trigger. The event rate acceptance of the HLT trigger system was 1.5 kHz.

Regions Of Interest (ROIs) are collections of acquired data within events that contain interesting information that likely does not correspond to background events. These ROIs are areas in the detectors' pseudorapidity-azimuthal space that contain objects that trigger the L1 system. These objects are ID tracking information, higher granularity calorimeter information, and precision measurements from the muon spectrometer. Due to a large amount of total possible acquired data, the detector data readout rate should be reduced to save memory. In doing this the ROIs are identified by the L1 trigger and are precisely the events that are inputted into the HLT for further processing.

L1 accept decisions [1] are distributed by the Timing Trigger and Control (TTC) system. The TTC is a system that provides for the distribution of temporal information of acquired data. This acquired data is recieved either directly by the front-end electronics or indirectly via the Readout Drivers (RODs). The RODs are connected to the DAQ system via the Readout Links (ROLs). Acquired data are pushed from the front-end electronics after classified by the L1 accept decisions into the RODs and are then forwarded via the ROLs [64].

The Central Trigger Processor (CTP) forms the L1 accept decision and distributes it to the HLT, and the TTC [65]. L1 accept signals inputted to the HLT are accurately timed with respect to each beam [64].

The general flow of information is illustrated below in figure 5.1:

The Readout System (ROS), L2 system and Event Builder are connected to two central core switches (the ROS and L2 processing nodes) via a layer of concentrator switches. The second central core switch provides redundancy and additional bandwidth for the first. Events are then built in the Event Building block (see figure fig. 5.1) then transferred via a third core switch to the Event Filter block.

Revisiting and expanding on the description of LBs presented in section 4.4.1, we now discuss LBs from the vantage point of newly presented information. An LB from the perspective of the DAQ system is a

[1] Accept decisions are the output of the particular trigger sub-system

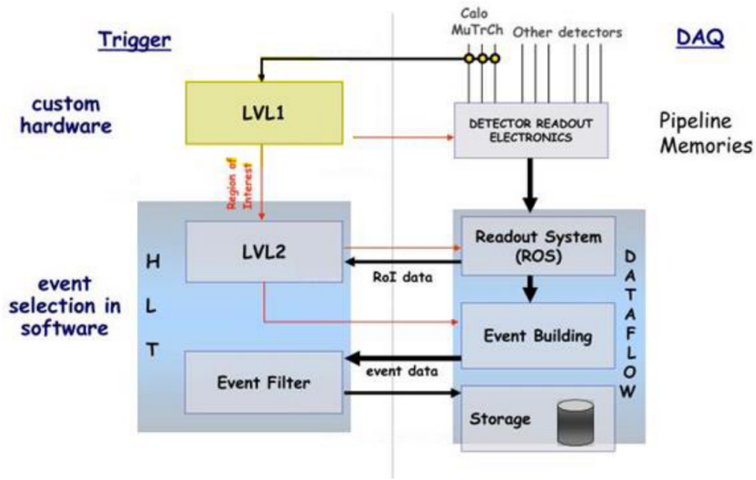


Figure 5.1: Illustration of general information flow of ATLAS Trigger and DAQ systems [66].

collection of events acquired during a short time interval (1-2 min) during which DAQ conditions were stable. Stability here is defined in terms of constant luminosity and no changes in detector operating conditions. The Luminosity Block Number (LBN) is issued with an integer timestamp (incrementing by 1 for each sequential LB) provided by the CTP, and communicated to the HLT. The LBN is stored along with the acquired events in the event data.

5.2 Data Quality

The luminosity delivered by the LHC, the number of events recorded by an experiments detector, and how many of these events are good for physics analyses are related and not equal. The number of events that are good for physics in relation to the total number of events recorded by a detector is the Data Quality (DQ). As shown in figure 5.2 we see the relationship between LHC delivered luminosity, ATLAS recorded luminosity, and recorded luminosity of data good for physics, over the course of Run-2 of the LHC [4].

The DAQ system is designed to be able to handle significant changes in run conditions to minimise the amount of data being lost or marked bad for physics analyses. The evaluation of this design function is essential and is done by monitoring the DQ during DAQ periods to ensure sufficient quality of data for physics analyses. DQ is analysed for individual sub-detector components.

5.2.1 Data Quality Monitoring Framework

Due to the complexity throughout the DAQ process at ATLAS, automatic DQ assessments of incoming data, and visualisation tools for easy identification of any issues are necessary. The Data Quality Monitoring Framework (DQMF) monitors the DQ and operational conditions of the DAQ system. [67].

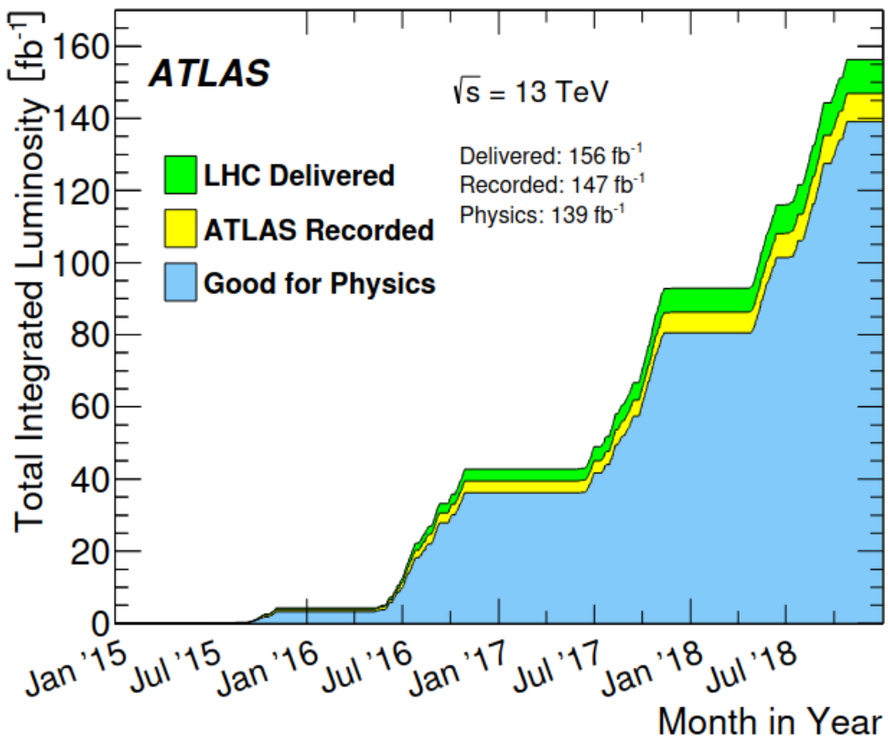


Figure 5.2: Integrated luminosity delivered by the LHC, recorded by ATLAS during stable beam conditions at centre-of-mass energy 13 TeV, and the integrated luminosity of data certified for physics analysis [4]

.

The concept of the DQMF is that of a DQ 'tree'. The leaf nodes of this tree are histogram checks, and interior nodes combine results from their child nodes. The histogram checks that produce DQ results range in complexity from tests that simply identify empty histograms, to ^{2} tests and distribution shape comparisons.

The DQ results propagate up towards the parent node of the tree. The DQ flags implemented are green, yellow, red, and undetermined. Green signals that the automated histogram checks passed the criteria, yellow signals the proposal of a manual review of the histogram in question, and red signals the existence of a potential issue [4]. The 'undetermined flag' is used in cases such as the statistics being too few for sufficient checks to be performed [68].

5.2.2 Data Quality Conditions

The DQ conditions are recorded by setting defects in the defect database. This defect database is contained within the conditions database. All temporal configuration, status, and calibration information are stored in this conditions database.

A defect is an issue recorded within an Interval Of Validity (IOV), with LB-level resolution, during which

the detector conditions are not nominal. Defects are stored for further evaluation. They do not necessarily indicate that data need to be rejected (hence their distinction from flags). A versioning mechanism is in place to ensure the reproducibility of results and allows for the evolution of defects with a gained understanding of detector problems.

There are two types of defects. These are [4]:

1. Primary defects: This type of defect is normally set manually by DQ and subsystem experts as described in section 5.2.1. By default, primary defects are set to 'absent' for an IOV unless the defect is loaded during DQ review. These defects are associated with individual LBs, and are stored.

2. Virtual defects: This type of defect defines the logic used to evaluate the toleration of a primary defect, and whether rejection of data is necessary due to the presence of the primary defect should it not be tolerable. Virtual defects are defined by logical combinations of either primary defects or other virtual defects, and are computed upon access (they are not stored like primary defects). They effectively combine defects into higher-level concepts [69].

An illustration of the logic of primary and virtual defects is given below:

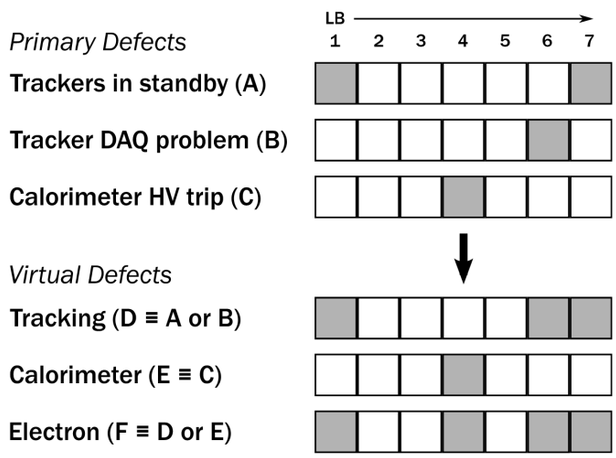


Figure 5.3: Illustration of primary and virtual defects [70]

.

An example of a defect sufficient to reject data for physics analyses is; a LAr calorimeter virtual defect is defined by all LAr calorimeter system defects serious enough for rejection of data. The virtual defect contains information that references primary defects that describe conditions where the relevant data should not be used. Using this virtual defect simplifies querying the defect database when constructing a Good Runs List (GRL).

5.2.3 Online Monitoring

Online, the DQMF allows the avoidance of recording faulty data by automatically performing checks on histograms from the various sub-systems at all stages of the data flow and flagging problems to the shift crew as they occur. The automation of the histogram checks using machine learning is helpful because as more information is gained over the course of the LHC being operational, case-by-case checks by inspection would become tedious. This is particularly the case given that the knowledge and understanding of defects are enhanced over the course of a run, so the training data used to classify faulty data become more than sufficient for automation.

The crew on-shift interacts with the DQMF with the Data Quality Monitoring Display (DQMD) which is a platform used to alert the crew to problems and allow them to debug. As of Run-2 DQMF is designed to be able to monitor electronic channels from all sub-detectors [71]. This monitoring occurs live, and allows for the monitoring of performance and detector status. An example of this online monitoring is if we look at the hit occupancy per unit area in the Pixel, SCT and TRT subsystems in which cold spots are present. The cold spots are as a result of possibly non-operational modules shown in black in figure 5.4. This may point to a calibration issue that would be difficult to identify using individual subsystem monitoring alone. This is an example of a test whose results would be propagated up the DQ tree (as described in section 5.2.1) to give top-level flags of the conditions.

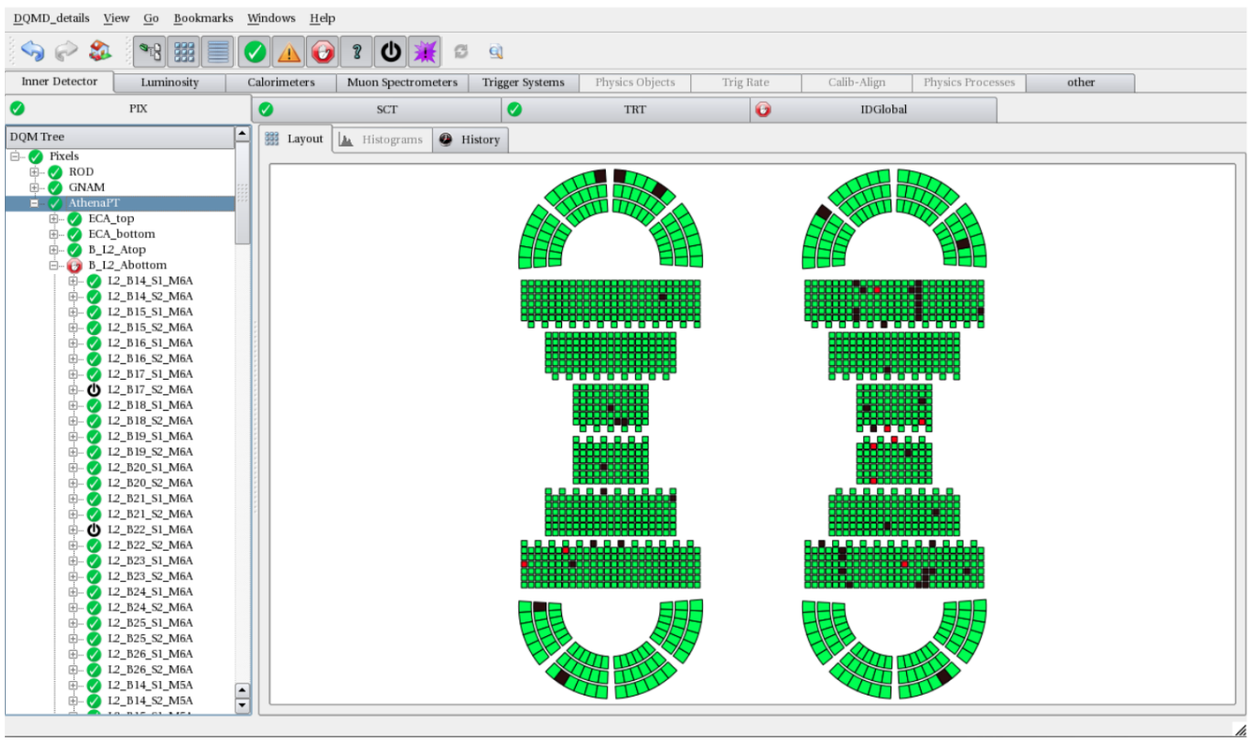


Figure 5.4: Image showing interface of DQMD allowing the monitoring of individual modules [72]

.

Where relevant, data of monitored runs are overlain with data from an assigned reference run. In ad-

dition, there is a mechanism that allows the viewing of the same monitoring output from different runs synchronously. An illustration of this is shown in figure 5.5. The figure shows a monitored online run with a reference run and includes a visual interface that describes the configuration. The configuration specifies input location, the checks and thresholds of the histograms [72]. The cold spots mentioned above shown in terms of the histograms are the white (as opposed to light purple) parts that should be filled in on the left histogram in figure 5.5.

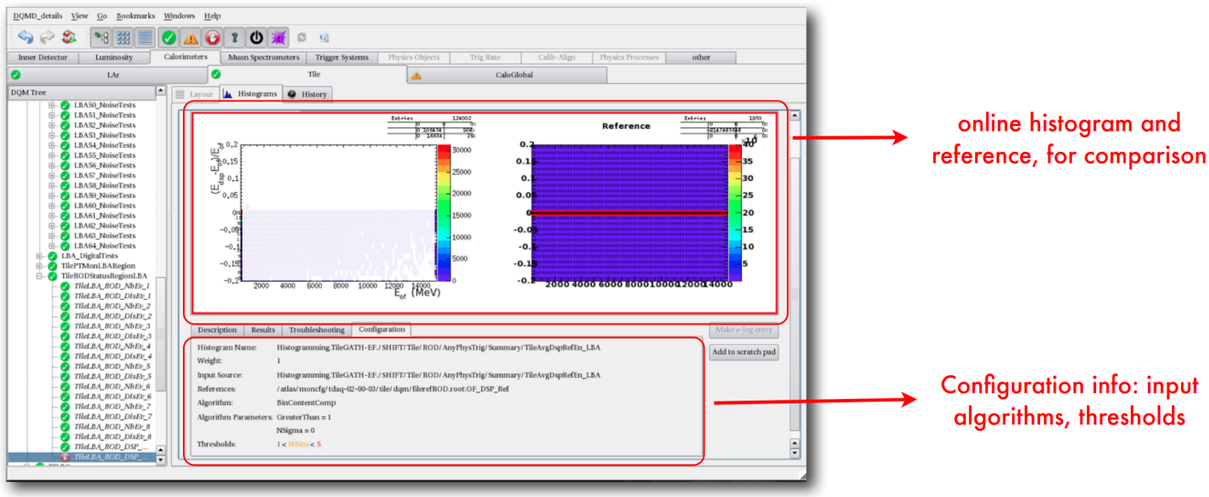


Figure 5.5: Illustration of a histogram for a monitored run (left) shown against a histogram for reference a run (right), with configuration information shown below them [72]

.

5.2.4 Data Quality Evaluation

The ultimate evaluation of online DQ results from the consolidation of the online DQ assessments for each of the sub-detector components. DQ is then monitored offline using fully reconstructed data [67]. This ideally occurs within 24 hours of the data being recorded to verify detector calibration and alignments.

The final product of the evaluation of ATLAS recorded data described above is the GRL for the run in question. The GRLs are lists of all the LB datasets within some runs of a data-taking period of the detector that are good for physics analyses. The GRLs are generated using the offline DQ assessments [71]. The GRLs are stored in a set of XML files that contain lists of LBs that are certified for physics analyses.

Using the information in these files, the integrated luminosity of data recorded by ATLAS that is 'good for physics' is calculated.

5.2.5 Data Quality Performance

All problems that result in the rejection of data are taken into account when calculating the DQ efficiency. The DQ efficiency is calculated as a weighted fraction of the integrated luminosity of the good for physics data with the integrated luminosity of recorded ATLAS data. This weighted fraction is that of the blue histogram in the numerator and the yellow histogram in the denominator of figure 5.2. Only periods during which recorded data are intended for physics analyses are used in calculating the DQ efficiency i.e this excludes commissioning and detector calibration runs.

DQ defects can be associated with physics objects that correspond to a particular sub-detector. As a result, the DQ efficiency may be expressed within the context of each sub-detector.

To illustrate how the results of DQ efficiency may be expressed, the total Run-2 DQ efficiency is shown in figure 5.6, with the efficiencies for each of the sub-detectors given as well:

Run 2 Data Quality Efficiency [%]													
Dataset	Inner Tracker			Calorimeters		Muon Spectrometer				Magnets		Trigger	
	Pixel	SCT	TRT	LAr	Tile	MDT	RPC	CSC	TGC	Solenoid	Toroid	L1	HLT
Standard *pp* @ 13 TeV	99.50	99.85	99.68	99.52	99.65	99.83	99.60	99.96	99.98	99.79	98.84	99.57	99.94

		Data Quality Efficiency [%]	Integrated Luminosity
Standard *pp* @ 13 TeV	Good for Physics	**95.60**	**139.04 fb^{-1}**

Figure 5.6: Table presenting individual and combined DQ efficiency at ATLAS for Run-2 [4].

The high obtained combined DQ efficiency of 95.6% in Run-2 was achieved by consistent detector maintenance, software development and DQ assessments [4].

6

Machine Learning

Machine learning may loosely be considered as the automated detection of meaningful patterns in data. It is important when formulating a problem in machine learning (or in any field) to specify the context in which the problem is being formulated. In doing so, the data which may be useful given a particular problem will take different forms, and will require different procedures to obtain solutions most relevant to the scope.

Machine learning, in the context of electronic computers, means enabling machines with the ability to learn without having to explicitly program the function which maps the learned 'knowledge'. It is generally used in data analysis.

At ATLAS, machine learning is used on several levels along the chain of tasks. This results from the fact that we may use classification algorithms for pattern recognition, and by extension, these algorithms are applicable in duplicate removal, quality selection, outlier detection, and rejection, as well as event selection within some range of acquired data. In general, it is good for use in high-energy physics environments as it aids us with particle identification in searching for rare signals while maximising the suppression of all background information [73]. In the most relevant cases to this dissertation, it may be used for determining separation between acquired data for DQ validation using quality selection, and outlier detection.

6.1 Applications

Current research in machine learning is focused on multiple tasks such as computer vision, variable ranking, collaborative filtering, automatic translation, classification, and named entity and speech recognition.

Of these aforementioned methods, the most relevant to this thesis is classification. Classification is used when some decision-making process between possibly distinguishable datasets is required. This

may take on different forms depending on the particular task and in doing so, several machine learning algorithms and methods may be used [74].

6.2 Methods

In general, there are four machine learning methods. These are:

- Supervised learning
- Unsupervised learning
- Semi-supervised learning
- Reinforcement learning

The lines between supervised and unsupervised learning blur depending on the particular tasks required to be completed, and by extension, the forms of data used in the tasks. More precisely the degree of supervision is given by the proportion of labelled data in relation to unlabelled data within the dataset. Labelled, in this context, means that all the vectors in each dataset have names that correspond to some type of measurement or detector state variable. Full supervision would thus be the consideration of datasets in which all the variables are labelled, as input, and the target output variables are given as well. This is giving the algorithm input information and the desired output information, and training the algorithm to learn the underlying distribution that can map some new input information to its unique output information.

The algorithms used in this project are to determine the separation between datasets and identify which variables in these datasets contribute the most to this separation to identify points of failure within the ATLAS detector. This is that particular variables (or events or tracks within those variables) descriptive of each vector contained in the variable space of each of the datasets corresponding to subsequent luminosity blocks are compared with the corresponding variable in subsequent datasets. However, the two target variables for each dataset comparison in the DQEWS procedure, 'good' or 'bad', are not given as input to training. As such,the DQEWS procedure is unsupervised.

It is noted that the flagging of information gives us the 'good', and 'bad' (if anomalous), target variables for the DQEWS procedure. These target variables are not obtained by training an ML algorithm that takes as input any information and outputs either of these variables. A distinct methodology to this is used in combining the outputs of each individual ML algorithm to obtain the target variables, and is described in section 6.6.

6.2.1 Supervised learning

In the case of supervised learning, the idea is for the parameters of the function outputted by the training procedure to infer a mapping learned from labelled training data. The labelled training data implies known variables used for inference of some output that takes into consideration all these labelled variables (or labelled input vectors). Supervised learning may be said to be performed using some known 'ground truth' given by labelled desired outputs of the supervised algorithm in question.

Analogously, we may consider supervised learning in terms of autonomous agents. Consider some autonomous agent which takes as input some sensory information represented by a sequence of inputs (our training dataset). The agent has embedded in it the desired output of the procedure. The key point here is that the agent can be thought of as learning by example. This, in particular, is effectively equivalent to a machine defined within its own formal system which has a set of inference rules for the transition of its inputs to output. These inference rules are known.

Rigorously, the training set of the machine consists of $n \in \mathbb{N}$ ordered pairs, (x_1, y_1), ..., (x_n, y_n) where each of x_i are elements of some dataset and each of y_i are the corresponding labels. In the case of unsupervised learning, the input data may be labelled, but the primary distinction from supervised learning is that the desired outputs and their labels are not given.

6.2.2 Unsupervised learning

Similarly, we may analogously consider unsupervised learning in terms of the previously mentioned autonomous agents. Consider some autonomous agents which take as input some sensory information represented by a sequence of inputs (our training dataset) possibly but not necessarily labelled. In this case, the autonomous agents do not receive labelled target output datasets. The idea here is that the agent builds a representation of the input datasets such that it may then be used to infer a decision, predictions, communicate inputs to some other automaton, and reconstruction, through learned data representation [75]. The key point here is that the agent does not have a reference point for its output. It is learning by what can effectively be thought of as trial and error.

Rigorously, unsupervised learning may be viewed in terms of learning a probabilistic model of the input dataset. When the neural network trained such that it contains this probabilistic model is inputted with some vector $[x_t, ..., x_n]$ for $n \in \mathbb{N}$ given a set of previous inputs $x_1, ..., x_{t-1}$, the neural network models the probability distribution $P(x_t, ..., x_n \mid x_1, ..., x_{t-1})$ for $t \in \mathbb{R}$.

Such models may be used for anomaly (or outlier) detection or monitoring. As a simple case, consider

x_t, ..., x_n as a set of acquired data from some sub-detector of ATLAS and assume that P(x_t, ..., x_n | x_1, ..., x_{t-1}) is the learnt probabilistic model. This model may then be used to infer some new dataset given the learned underlying distribution. Each variable in this dataset may be probabilistically evaluated, and if this probability is low, the model is either poor, or the detectors' operational conditions are insufficiently calibrated [75]. This is due to this low probability meaning a low ability of the model in question to be able to find patterns in the data obtained from some sub-detector.

6.3 Classification

Classification is a procedure that distinguishes information in distinctly labelled datasets. These datasets contain vectors of the information that is to be classified in relation to other such datasets. The set of datasets used in some classification procedure defines the classification problem-space. Each of these datasets is defined as one of the true classes. A solution in a classification procedure is a quantification of the separation observed between a chosen set of datasets, each represented by a true class-label.

A set of numbers, for $\{z_1,..,z_i\} \in \mathbb{Z}$ is used to represent the true class-labels corresponding to all data within a dataset in a multi-class classification procedure. $Y(\overline{A}) = \overline{B}$ is the function of the classification procedure which maps the input dataset $\overline{A}$ to the output dataset $\overline{B}$. It is this mapping that allows us to quantify the separation between our datasets.

In a binary classification procedure, as is the relevant case for this thesis, the true class-labels are the two integer values in the set $\{z_1, z_2\} \in \mathbb{Z}$. These numbers which are also the lower and upper bounds of the sets that represent background and signal in that order, by convention.

Binary classification may be done using algorithms that construct a model trained to classify vectors within datasets where classification is not the primary function (in the case of neural networks), or using algorithms constructed for classification (in the case of decision trees, K-Nearest Neighbours, and Support Vector Machines). The most common techniques used for classifying events are Bayesian networks, Decision Trees, K-Nearest Neighbours and Support Vector Machines [76]. Relevant to the analyses done within the scope of this dissertation, we expand on Decision Trees.

6.3.1 Decision Trees

A Decision Tree is a directed acyclic graph [77]. This is that the nodes of this graph represent a transformation of the inputted information, and the links connecting the nodes in this graph are uni-directional such that information is transmitted in only one direction. The initial node has no incoming links and is

referred to as the root node [78]. All nodes perform a binary split operation of an inputted event with one child-node for some potential value corresponding to each of these splits. These child nodes then perform this same operation producing their children. An illustration of this procedure is shown in figure 6.1.

The procedure generally halts once no nodes at some layer are in disagreement about a classified event, or when the defined maximum depth of the tree is obtained. However, there are numerous halting criteria discussed below.

In a classification procedure that takes large datasets, it is common to have a collection of trees (or forests). The formal term for a forest is a disconnected directed acyclic graph.

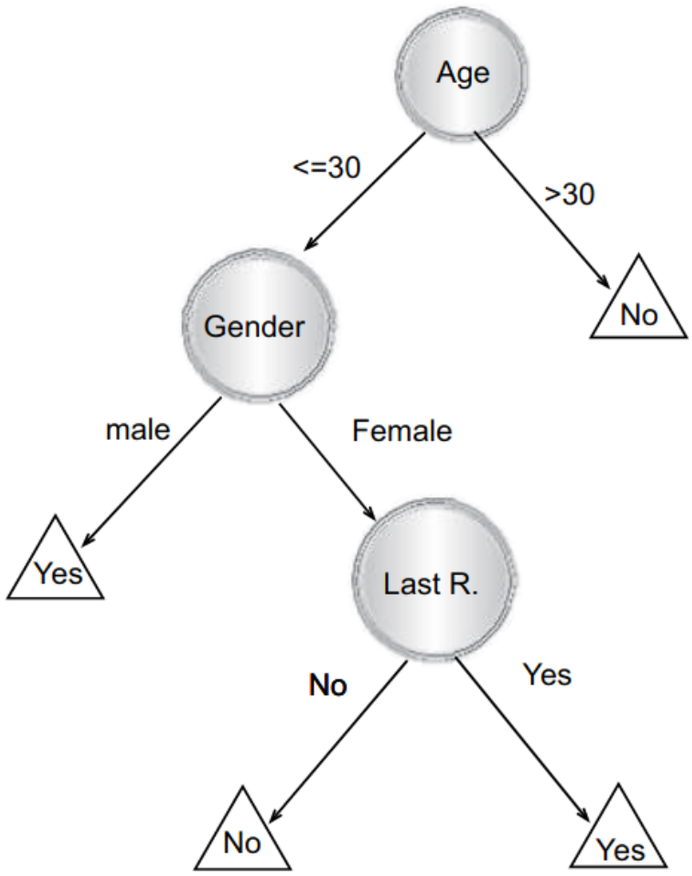


Figure 6.1: Example of decision tree showing process of binary split operations with criteria defined for each node.

6.3.2 Boosted and Gradient Boosted Decision Trees

Boosted Decisions Trees (BDTs) are a class of algorithms used in machine learning for classification problems. They take as input some dataset with variable (or feature) vectors which are descriptive of each dimension of a dataset, and another dataset with equivalent variable labels but possibly distinguishable

values in each of the vectors. BDTs make use of decision trees and apply to them a procedure known as boosting, and gradient descent. Generally, there are two widely used boosting procedures. Adaptive and gradient boosting. In this dissertation we discuss only Gradient Boosted Decision Trees (GBDTs).

Boosting is a meta-heuristic algorithm [1] used primarily for reducing bias and variance in supervised learning approaches. The boosting algorithm allows for a series of decision trees (these are weak learners) that a set of decision trees that have a high error rate to be combined such that a resultant forest (or ensemble) with a lower error rate (this is a strong learner) is attainable [80]. It follows that boosting may be considered as a set of algorithms that convert weak learners to strong ones [81].

Gradient descent is an iterative optimisation algorithm used to find local minima of differentiable functions [82]. The GBDT algorithm is a combination of boosting and gradient descent. The idea is that the GBDT classifies each event in the variable vectors to determine the degree of similarity between the two datasets that it has been trained on. It does so by estimating some function $\tilde{F}$(v') of the true function F(v) that maps v to v' where v, v' $\in \mathbb{R}^{\mathrm{N}}$ are vectors in an N-dimensional variable space, where there are J number of them which correspond to each unique event in the dataset. The model, $\tilde{F}$(v), that describes this mapping is trained in an additive manner such that in every step the a new tree is added to the forest in the optimisation process of the algorithm.

GBDT algorithms determine this model by minimising, through gradient descent, what is called the regularised objective which is defined as [83]:

$$L_t(y) = \sum_{x'}^{N} l(F_i, \tilde{F}_i^{(t-1)} - Y(v_i)) + \Omega(Y_t) \tag{6.1}$$

Here, each operand contained in the regularized objective is:

1. l is a differentiable convex loss function that determines the difference between prediction of the vector label, $\tilde{y}_i^t$ of the i-th instance at the t-th iteration of the boosting procedure.
2. F_i is the true model for input vector v_i
3. $\tilde{F}_i^{t-1}$ is the prediction of the true model at iteration, t-1
4. $Y(v_i)$ is an individual regression tree structure contained in the forest that has some weight, w
5. $\Omega(Y_t)$ is a term used to penalise the complexity of the model by reducing its propensity to over-fit

[1] Heuristic algorithms are search algorithms that are used for finding approximate solutions to problems where exact solutions may be computationally expensive. Decision trees are a heuristic algorithm [79]. Boosting is an algorithm that aims to improve the decision tree algorithm through heuristic methods.

The regularised objective is minimised using a second-order Taylor expansion such that:

$$L_t(y) = \sum_{x'}^{N}(l(F_i, \tilde{F}_i^{(t-1)}) + g_i \cdot Y(v_i)) + \frac{1}{2} \cdot h_i \cdot Y(v_i)^2) + \Omega(Y_t) \tag{6.2}$$

Where the first and second order gradients of the loss function, g_i and h_i are defined as:

$$g_i = \frac{\partial}{\partial \tilde{F}_i^{(t-1)}} l(F_i, \tilde{F}_i^{(t-1)}) \tag{6.3}$$

$$h_i = \frac{\partial^2}{\partial \tilde{F}_i^{(t-1)_2}} l(F_i, \tilde{F}_i^{(t-1)}) \tag{6.4}$$

Halting criteria

The growth of trees and forests by GBDT algorithms continues until a stop criterion is triggered. Common halting criteria are [78] :

1. All instances in the training set belong to a single class
2. The maximum defined depth of nodes has been reached
3. The number of cases in the terminal node is less than the minimum number of cases for the parent notes
4. If a node is split, the minimum number of cases in one or more child nodes is less than the minimum number of required cases for child-nodes
5. The best splitting criteria is not greater than a certain threshold

The GBDT halting criterion which gets triggered generally depends on which hyper-parameters one chooses to use to instantiate the model.

6.3.3 Gradient Boosted Decision Tree Performance

A classification can be made by choosing a classification threshold above and below which the classification probability for a specific event in the subject dataset places it in either the background (or

reference) class or the signal (or subject) class when compared to the reference dataset; with each dataset represented with binary classification true-class labels defined in the third paragraph of section 6.3.

The GBDT then produces the GBDT score with which the quality of agreement between the compared datasets may be assessed where 0.5 means that the compared datasets are statistically equivalent, and 1 means they are statistically distinct.

6.4 Model performance

Ensuring that the performance of a model is sufficient in that it does not over-train is important for the extraction of quantitatively useful results. Over-training of a model commonly occurs when the hyper-parameters of a model are insufficiently chosen for a particular modelling task. Hyper-parameters are numbers inputted into the definition of the model which have a range of effects on the model that they define. In the most relevant case, they may be thought of as the intrinsic characteristics that define the structure of the graph that describes the model, and how information passes through its links.

Hyper-parameter optimisation may be done by inspection i.e by manually tuning the inputted values. It may be done more efficiently by grid-searching.

Grid-searching is a method in whilabelch a models performance is checked for a set of chosen hyper-parameter values. Each of these hyper-parameter values is defined on an n-dimensional search-space that is the grid. Each point on this grid corresponds to a possible best model [84]. All positions on this grid are checked such that all combinations of the chosen hyper-parameters to be grid-searched are tested for the model.

The desired output behaviour of the model may be given. This is that the grid-search procedure may be guided by the desired behaviour of a particular performance metric [84]. For example, in a binary classification procedure with precisely the same two datasets to be classified, it is possible to optimise a grid searching procedure to check for the hyper-parameters which either minimise or maximise the GBDT score over the set of hyper-parameter configurations available in the search-space. Each of these situations is very likely to result in two different hyper-parameter configurations.

Performance metrics

For a binary classification task, results may be presented in a 2×2 Confusion Matrix (CM). The matrix elements take on some number $y \in \mathbb{R}$ in the interval [0, 1].

$$CM = \begin{bmatrix} TP & FN \\ FP & TN \end{bmatrix} \tag{6.5}$$

The CM elements are the True Positive (TP), False Negative (FN), True Negative (TN), and False Positive (FP) predictions of the classification task.

The CM matrix elements are sufficient for determining performance metrics which are used to evaluate the performance of the binary classifier. The following performance metrics are calculable using these matrix elements [85].

Precision: This is a measure of all correctly identified positive cases of a particular multi-class classification procedure from all predicted cases in some set of data.

$$\textbf{Precision} = \frac{TP}{TP+FP} \tag{6.6}$$

Recall: This is a measure of all correctly identified positive cases of a particular multi-class classification procedure from all actual cases in some set of data.

$$\textbf{Recall} = \frac{TP}{TP+FN} \tag{6.7}$$

F1-score: The **F1-score** is obtained by calculating the weighted harmonic average of precision and recall [86]

$$\textbf{F1-score} = \frac{TP}{TP+FN} \tag{6.8}$$

Accuracy: This is a measure of all correctly identified cases in general. The accuracy is commonly interpreted as the GBDT score.

$$\textbf{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{6.9}$$

Error: 1 - **Accuracy**. This is a measure of all incorrectly identified cases.

$$\textbf{Error} = 1 - \frac{TP+TN}{TP+TN+FP+FN} \tag{6.10}$$

Receiver Operating Characteristics: Receiver Operating Characteristic (ROC) analysis is a method borrowed from Signal Processing and has become a standard for binary classifier evaluation. It compares the TP Rate (TPR) and the FP Rate (FPR) [87]. On a ROC curve, the FPR is plotted against the TPR.

The ROC curve illustrates the ability of a binary classifier to discriminate between events in a pair of datasets. It gives the probability that an event is classified correctly at the classification threshold particular to a unique pair of compared datasets. In particular, the area under the ROC curve (ROC AUC) is the measure used to describe the separation between datasets analysed using ROC analysis.

6.5 Anomaly Detection

The ability to detect anomalies (or outliers) within the data descriptive of an environment is as necessary for the observation, and experience, of ideal conditions of some agent interacting with this environment. To the experimental physicist, an ideal environment may include something along the lines of all acquired data being usable and having been acquired with no errors anywhere along the path of the flow of information from the initial acquisition of physical data to the final analysis, possibly leading to verification of theory i.e discovery. This is an ideal situation that does not exist precisely as written, but may be asymptotically approached should the ability to isolate anomalous events in the acquired data be maximised.

Anomaly detection in temporal data, using machine learning, requires methods that allow for the consideration of information allowing the determination of the resultant dynamics that emerge from this information. Time is represented by the association of acquired data that when split into datasets and placed in order, each dataset represents a discrete constituent of temporally ordered information. The set of all these associative datasets contain all information acquired over the course of a DAQ period. We note in particular that while these datasets associate and form discrete elements of time, we do not specify any assumptions of causality from one set to another given that changes to acquired data may occur suddenly due to unexpected changes in the measurement systems configuration. This removes the notion of pure causality between acquired data, although causality exists to some degree given the physics of the particles being measured.

6.6 Concept drift

Concept drift is that the statistical properties of data change with time in the sense that there are changes in the underlying distributions of datasets with time. These changes, when considered between sets of transformations and parameters within some formally defined system, represent conditions sufficient for the definition of concepts, and their situational changes. A concept is defined as the joint probability distribution of a set of input variables, and their labels. This is, a concept defined for a dataset with n variables where $n \in \mathbb{N}$. The concept is the set of ordered pairs, (x_1, y_1), ..., (x_n, y_n) where all of x_n $\in \mathbb{N}^m$ are vectors in an m-dimensional event space which contain the information for the corresponding variable label y_n. The joint-probability distribution of this set may be written as $p(x,y)$ such that:

$$Concept = p(x,y) \tag{6.11}$$

Concept drift is defined as the change of p(x, y) with time. Mathematically this may be expressed as $\exists$ x such that $p_{t_0}(x, y) \neq p_{t_1}(x, y)$ where t_0 and t_1 are different positions in time [88]. One may extend this logic to treat these concepts as able to encompass periods of time. This may be expressed as $\exists\ x$ such that $p_{[t_0,\ t_1]}(x, y) \neq p_{[t_1,\ t_3]}(x, y)$ where t_0 where [t_0, t_1], [t_1, t_3] are distinct periods in time [89].

Concept drift may occur in different ways. These are:

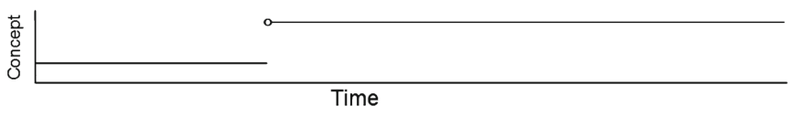
1. The variables underlying distributions may drift while the variable labels underlying distributions remain unchanged.
2. The variables underlying distributions may remain unchanged while the variable labels underlying distribution drifts.
3. A combination of both variables and their labels drifting.

In addition to the particular variable and variable-label configuration, concept drift may occur with different trajectories of the concepts that define a system with time. An illustration of this is shown below:

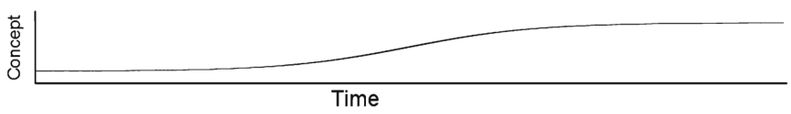
Research of concept drift involves the development of methods and techniques for the detection of drift of datasets over time. In some environment where machine learning is used for inference, the problem of concept drift must be addressed to maximise the accuracy [2] of predictions [91].

The machine learning environment that the DQEWS makes use of is defined, in this dissertation, as

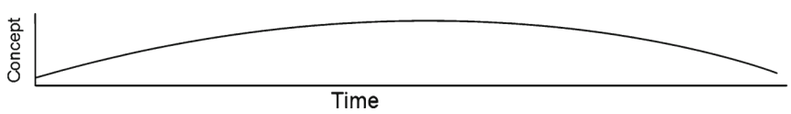
[2] This is a distinct meaning of accuracy from accuracy defined as a binary classification performance metric



(a) Sudden concept drift.



(b) Gradual concept drift.



(c) Reoccurring concept drift.

Figure 6.2: Here we have sudden, gradual and reoccurring types of drift [90].

supervised. We know, from the DAQ process, all variable labels of the training and testing datasets inputted into our model.

Concept drift as considered from the perspective of the DQEWS is therefore described in such a way that drift is to be observed for the underlying distributions, and therefore for the attributions of all variables, x_i, within the ordered pairs that define equation (6.11) as well.

The DAQ system in the context of concept drift is as follows. In ATLAS, the DAQ system as described in section 5.1.1 is additionally a time-continuous process. In particular, this means that over the total time interval (run) during which the detector is acquiring data through measurements taken by the various sub-detectors, these measurements do not stop being taken for the duration of this acquisition process [4].

These measurements are sorted into datasets which contain all the information for measurements taken during a particular sub-interval over the course of this run. These sub-intervals of time within this run are the LBs. This sorting of all acquired data into LBs is, in principle, what allows us to determine the temporally dependent dynamics of the particular sub-system having acquired the data used in the analysis, in such a way that we may use binary classification machines to determine the magnitude of separation between some configuration of LBs over the course of this run. This is a description of the problem of concept drift exhibited by the acquired data at ATLAS.

Within the context of this dissertation, this configuration of LBs that allows us to quantify drift is to be considered in such a way that the data corresponding to the initial and succeeding LBs are contiguously arranged and separable in such a way that the ensemble method [3] that describes the drift procedure is possible. This drift procedure is a streaming data classification problem that makes use of an incremental

[3]Not to be confused with ensemble learning, where several models are trained and combined to produce a single solution from the sum of the constituent models' outputs

learning framework.

6.6.1 Classification Procedure

The classification procedure is that numerous unique GBDT models are incrementally trained in such a way that a baseline input LB is used as reference, and each subsequent LB contained within the set of all available LBs are used as subject. Assessing the validity of taken measurements may be done by determining the separation between these LBs.

The classification procedure is implemented as follows. A LB contains data that may be represented n ordered pairs $(x_1, y_1), ..., (x_n, y_n)$ $n \in \mathbb{N}$ which describe a concept of the dataset such that:

$$\Lambda = p(x, y) \tag{6.12}$$

The set of all available temporally ordered LBs is the incoming data-stream. This data stream is the set of sequential LBs such that:

$$\Lambda set = \{\Lambda_1, ..., \Lambda_n\} \tag{6.13}$$

where Λ_n is the information contained in most recently acquired LB and n is $\mathbf{card}(\Lambda - set)$ which equals the total number of available LBs for training.

The designation of accurate time-stamps for each LB is helpful for the comparisons as it allows us to distinguish between data acquired during particular intervals in time. Following this, we distinguish between the information contained within a LB and a naming structure based on these accurate time-stamps that represent the information contained within a LB.

These time-stamps are issued by the Central Trigger Processor of ATLAS with labels of Luminosity Block Number (LBN) identifier where $N \in \mathbb{N}$, or a Run Number (Run N)[92]. A LB is precisely the information contained in data acquired during the defined period, and the LBN is the label that points to that LB i.e The 14th Luminosity Block (LB 14) is the recorded information that corresponds to a Luminosity Block Number of 14 (LBN of 14).

We represent this set in terms of the LBN identifiers such that:

$$LBNset = \{LBN_1, ..., LBN_n\} \tag{6.14}$$

where LBN_n is the most recently acquired LBs LBN identifier, and n is $\mathbf{card}(LBNset)$ also equalling the total number of available LBs for training. This step is important because using LBNs to point to the information contained in the LBs allows us to identify the particular point in time a particular LB was acquired during a run.

Using equations (6.13) and (6.14) we describe the method in the following way. We train numerous unique GBDT models to describe the ensemble such that a single set of compared reference and subject LBs are concepts written as:

$$\begin{aligned} LB\,comparison &= \{\Lambda_{reference}, \Lambda_{subject}\} \\ &= \{p_{reference}(x,y), p_{subject}(x,y)\} \\ &= \{p_{t_1}(x,y), p_{t_2}(x,y)\} \end{aligned} \tag{6.15}$$

Which we may generalise for all compared LBs as follows:

$$LB\,comparisons = \{\{p_{t_i}(x,y), p_{t_j}(x,y)\} : i \in LBNSet, j \in LBNSet\,and\,i < j\} \tag{6.16}$$

equation (6.16) is the set of pair-wise combinations of LBs such that for some reference LB denoted by index i, we have a corresponding subject LB denoted by index j. The subject LB is ordered in time later than the reference LB by construction.

We may then make use of some distance measure M to determine the magnitude of difference between two concepts separated by some temporal period. Here, M may be any statistical distance measure between two distributions. Outputted from the GBDT training procedure, one may make use of any performance metric that describes the discrepancy between the inputted datasets as the statistical distance measure.

6.6.2 Drift Procedure

We specifically define the drift procedure in the context of this dissertation as a generalisation of the classification procedure. This generalisation is the consolidation of the elements in the set that describe the classification procedure (the LB comparisons) such that we may make inferences on the system as a whole. The drift procedure takes drift into account by making further comparisons of the LB comparisons contained in the set defined by equation (6.16).

An illustration of drift is given in figure 6.3 for some arbitrary run A where all the acquired LB information is good for physics, and for some arbitrary run B where all but one of the acquired LBs are good for physics.

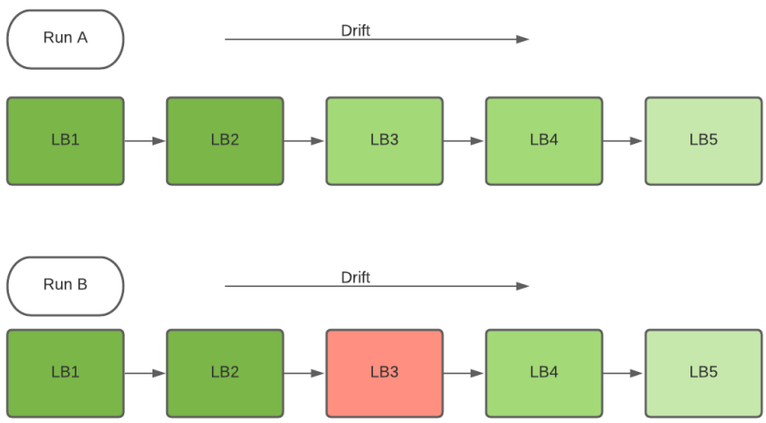


Figure 6.3: Illustration of good drift, and drift containing anomalous information.

For the illustration of concept drift, let us consider M = ΔGBDT score for a particular comparison of reference and subject LBs. In figure 6.3 the magnitude of M directly corresponds to the purity of the green squares. The drift is shown by a whitening of the green squares with increasing ΔLB, and for this arbitrary Run-A, this drift is expected as it is observed for LB information contained in the GRL. For our arbitrary Run-B, the red LB is anomalous in the context of how we expect drift to occur. The goal of the drift procedure is to flag good LBs as 'good' and bad LBs as 'bad'. Using the example of the two arbitrary runs shown in figure 6.3, the idea is that we construct the flagging criteria using only good data (like Run-A) and then evaluate the criteria to see if a bad LB in some other run (like Run-B) will be flagged as 'bad'.

Quantifying drift

The path length, P of drift may be taken into account. This is the cumulative deviation in a performance metric over subsequent LB comparisons with a particular LB used as the reference LB. The total duration

of this cumulative deviation is the region of this path length allowing us to define P_{Region}. Taking the sum of deviations we have.

$$P_{Region} = \sum_{i=1}^{N} \Delta\phi_i \tag{6.17}$$

For example [{1: 'LB1', 2: 'LB2'}, {1: 'LB1', 3: 'LB3'}] (from Run-A in figure 6.3) the LB comparisons of LBs contained in a run are compared, where elements of this list are the LB comparisons, to obtain the deviation:

$$Deviation_{\Delta LB comparison_i} = \Delta\phi_i \tag{6.18}$$

The average drift rate (for a particular performance metric) is then the quotient of this path length with the duration of monotonicity

$$\text{DriftRate}_{\text{avg}} = \frac{P_{Region}}{\Delta LB_{Region}} \tag{6.19}$$

7

Analysis

7.1 Description of used data

The model which describes the flow of information from raw data of physics events that enter the HLT is called the Event Data Model (EDM) [93]. There is an EDM that corresponds to only the ID. The data used for these analyses are n-tuples constructed from full event data from the ID [1].

We refer to a subset of all the information contained in a LB as a LB dataset. The LB datasets used in these analyses are from runs whose data were acquired during Run-2[2] of the ATLAS detector for proton-proton collisions at a centre of mass energy of up to $\sqrt{s} = 13$ TeV. In these analyses the data used does not include data taken in other runs. All of the LB datasets mentioned are from runs that are not necessarily flagged as 'good for physics' according to the DQ group. This is because of the goal of constructing the DQEWS being to be able to identify and flag LB datasets that are not good for physics analyses.

All data used for analysis are LB datasets that are n-tuples that have been constructed to contain only the ID tracking information from each run. Each of the LB datasets used in these analyses contains a subset of information acquired information during a LB time interval at ATLAS. This is because each LB is comprised of several LB datasets that contain the information acquired during that period. The information contained within a sub-interval within a LB is considered to be indistinguishable from information contained within some other sub-interval of that same LB. Within the context of DQEWS, this means that in a binary classification procedure using a GBDT, a comparison of LB datasets that contain subsets of information from data taken during an LB time interval should yield performance metrics that indicate indistinguishable classes. We test for separation between pairwise combinations of the LB datasets available at the time of training. For evaluation of the drift procedure by taking these LB comparisons, we may use either all the available LBs or some subset of them. A used subset is for fixed reference LB

[1] All data used in these analyses was prepared for me by my co-supervisor, J.R.Catmore, University of Oslo

[2] Data taken during the period 2015-2018

i, with all $j <$ **card**$(LBNset)$ in equation (6.16).

7.1.1 Variables

Tracks have associated vertices as mentioned in section 4.2.1. The ID tracking variables contained in the LB datasets are the properties of the tracks.

The tracking variables are defined in such a way that they contain information from both local and global position coordinates, and include general properties about tracks and the quality of their fits.

A single track, T, expressed with respect to the nominal beamline axis may be parametrised at the perigee as:

$$T = (d_0, z_0, \phi, \theta, \frac{q}{p}) \tag{7.1}$$

The parameters in equation (7.1) are defined as follows [94, 95]:

1. d_0: Transverse impact parameter.
2. z_0: Longitudinal impact parameter.
3. ϕ: Momentum azimuthal angle.
4. θ: Momentum polar angle.
5. $\frac{q}{p}$: Charge divided by the momentum.

d_0 and z_0 are parameters that are the distance between the center of the beamline (z-axis) and the track, and the distance between the impact point and the primary vertex along the z-axis respectively.

The general properties of tracks have their own associated tracking variables. These variables are based on the following:

- Number of hits on a module contributing to a tracks fit.
- Number of shared hits on a module. contributing to a tracks fit and another tracks fit.
- Number of holes. This is the number of crossed modules where no hit was found.
- Number of crossed inactive modules that should be counter as a hit.

- χ^2 and number of degrees of freedom values.

The uncertainties in an estimate of T (equation (7.1)) are described in a covariance matrix, C [43]. The covariance tracking variables in the data are the upper triangular elements of this matrix. These covariance matrix elements are shown in table A.1 as 'cov' with the row and column numbers in their names.

The variables in the LB datasets used in the analyses in this dissertation are ID tracking variables only. Each of these variables is represented by a vector whose elements correspond to a single track. The list of variables contained in the LB datasets used for the analysis contained in this dissertation is shown in table A.1.

7.2 Smooth Instantaneous Luminosity Regions

The use of all available LB datasets and taking only the difference in LBNs between our pair-wise LB comparisons as the independent variable does not take into account the amount of acquired information contained within a LB.

The drift procedure requires that the independent variable of the constructed trajectory of separation plots be approximately monotonic, in a similar fashion to the passage of time, to yield useful results representing drift over the course of a run.

Of the LB datasets we have available, while the GRL flags may state that a particular LB is 'good', it is important to ensure that the corresponding instantaneous luminosity is within some interval that displays approximately continuous behaviour in accordance with the monotonicity of the passage of time.

To take this into account, we look at the behaviour of instantaneous luminosity of all contained LBs over the course of a run and check the LBN intervals within the run that correspond to intervals during which the beam conditions were sufficiently stable to exhibit approximately continuous changes in the instantaneous luminosity.

7.2.1 Choosing LBs

The order of the LBs inputted into the classification procedure is based strictly on the order of the LBs in monotonically decreasing intervals of instantaneous luminosity. This order tends to correspond to the order of acquired LBs based on strictly increasing LBNs, but is not the case in general given that there are cases of sudden changes in the instantaneous luminosity in runs. I refer to these sudden changes in

instantaneous luminosity as 'jump discontinuities'. These jump discontinuities seem to correspond to LBs that have the 'Lumi_Rapidly_Changing' tolerable defect which corresponds to reoptimizations of beam conditions in the LHC during a run. At and near these jump discontinuities we observe later LBs suddenly having higher instantaneous luminosities than earlier LBs.

Despite these jump discontinuities, the instantaneous luminosity is approximately monotonically decreasing over the course of a run, and the DQEWS has this condition built-in. This is why we require that the conditions of the ordering of information be explicitly stated and that LB datasets which do not satisfy this condition be excluded from the analyses.

The LB datasets that are inputted into the drift procedure are simply those that are contained within the pseudo-continuous instantaneous luminosity regions that satisfy the above condition. The LB datasets are shown in figures 7.1 and 7.2 where the black dots are all the LBs contained in the runs, and the coloured dots are the used LBs in the respective runs. Each different colour corresponds to an LB that lies within a pseudo-smooth region defined in-between the above-mentioned jump discontinuities. It is these coloured LBNs that are then used in the analyses. Any LBs that are located precisely within the range of LBNs of the jump discontinuities are excluded.

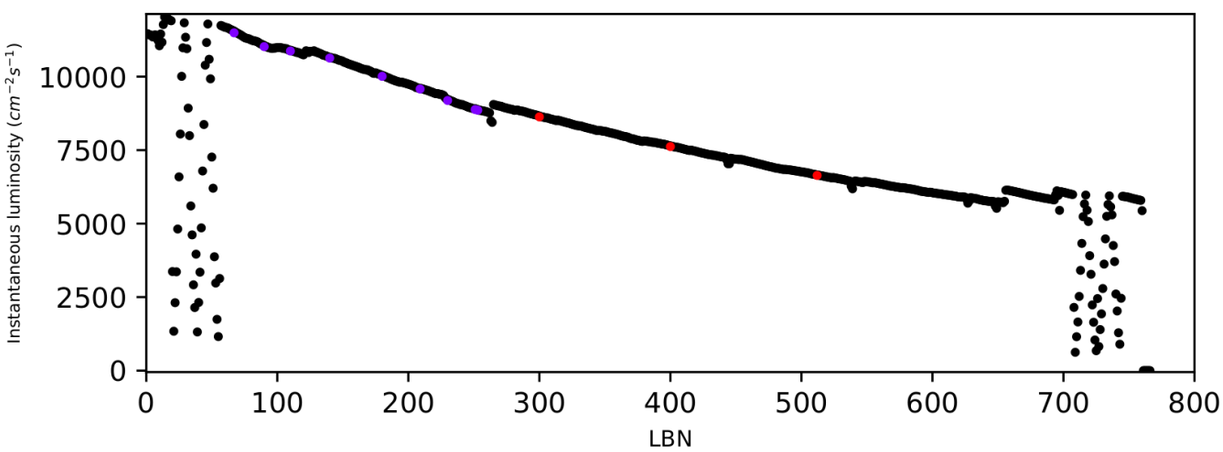


Figure 7.1: LBNs for LB datasets used from Run-349268 that are contained within pseudo-smooth regions of instantaneous luminosity.

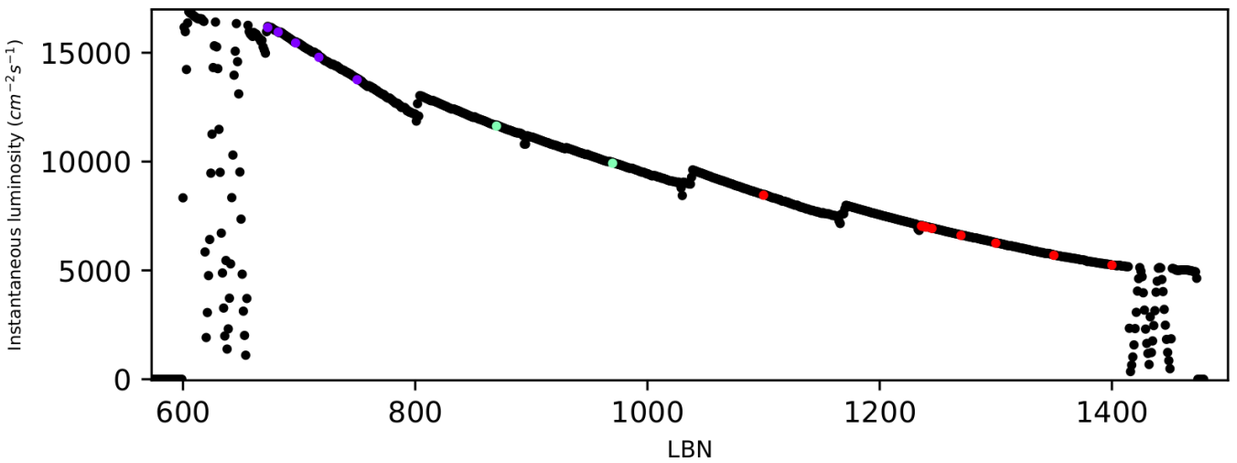


Figure 7.2: LBNs for LB datasets used from Run-357409 that are contained within pseudo-smooth regions of instantaneous luminosity.

The LB datasets (all acquired during Run-2) represented by the coloured dots in figures 7.1 and 7.2 that we use in the construction and implementation of the DQEWS have their LBNs shown in table A.2 and table A.3 in the appendix.

7.2.2 Goodness of LBs

The goodness of the LB data used is first obtained from the GRL and then assessed by the DQEWS to check for equivalence. Given that all LB datasets used in these analyses contain only ID tracking variables the goodness of the selected LB datasets, based on the GRL, have been presented as follows:

1. If the LB dataset is good there is either no ID tracking variables defect, or should there be such a defect it is flagged as a tolerable defect

2. If the LB dataset is bad there is an ID tracking variable defect and it has been flagged as intolerable.

All except for one of the LB datasets that have been used in these analyses have been flagged as good for physics in the GRLs generated by the ATLAS DQ group.

The bad LB dataset is the LB dataset with LBN 252 in Run-349268. The intolerable defects that result in the ID tracking information contained in this LB being flagged as bad are as follows:

1. Pixel barrel in standby.

2. Pixel endcap A in standby.

3. Pixel endcap C in standby .

4. Pixel region in standby.

5. Pixel layer 0 in standby.

7.3 Classification Procedure Implementation

Using the LBs for each run that we have, we have trained multiple classifiers for all LB pairs contained within a run. This has been done because the idea is that we would like to obtain outputs from the classifiers trained that allow us to determine a relationship between these outputs that describe the set of data according to their flag on the GRL for that run.

For this dissertation Gradient Boosted Decision Tree's (GBDT's) [3] are used to classify tracks contained in the LB pairs. The package used for classification was the XGBoost python package [96]. This package makes use of a Gradient Boosted Decision Tree (GBDT) algorithm. The usage of a GBDT was chosen in favour of other algorithms because of a combination of the classification power of decision tree methods with respect to computational price and the ability to determine variable attributions for some trained model. XGBoost also outperforms other tree-based models according to [96] based on the algorithms.

In principle, we have two categories of data that are to be classified by the GBDT and be determined to be distinguishable from one another or not. These categories may take on various names depending on the particular task, but in physics, these are by convention named the signal and background datasets containing some equivalent number of variables (or variables) in some vector $\overline{A}$. In this dissertation, the two categories are any two LBs that are to be tested for degree of separation, where the first category is the reference LB and the second category is the subject LB.

The reference LB is the LB that is first in the list of LBs, and the subject LB is any LB that succeeds the reference LB that is contained in the set of LBs available for the particular run in question. For all LB comparisons done in these analyses, the reference and subject LBs are contained within the same run, and all LBs are those that pass the criteria described in section 7.2.1. The classification procedure defines the set that contains all these individual LB comparisons.

7.3.1 Dataset prepossessing

Prior to the input of the LB datasets into the classification procedure, a few steps are done to ensure consistent results for these analyses. These steps will be enumerated in order of doing.

[3] 'Gradient boosted Decision Tree' (GBDT), 'Gradient Boosting Machine' (GBM) and gradient boosted regression tree' (GBRT) may be used interchangeably

1. We remove all variables that result in rapid drift of the GBDT errors. This is because while we want to test for separation, the procedure is only helpful if there is sufficiently a fine resolution of changes in separation that we can analyse to find a trend. In this step, the rapidly drifting variables are ['trks_vx', 'trks_vy', 'trks_vz', 'trks_pt']. Further studies are needed to find the reason for these variables resulting in rapid drift.

2. The ID tracking variable vectors in all of the LB datasets are then normalised using min-max normalisation. Min-max is used because it scales all vectors such that all their values are between 0 and 1. It does not make their means the same. This is helpful since the features have varying ranges so normalising them to being within the same range allows computation of the outputs to more efficiently occur.

We note that additional variables are removed from the LB datasets because the goal was to find a procedure that can generalise the flagging procedure between runs. Variables not contained in both of Run-1 and Run-2 data are removed. In these studies we have used only Run-2 data and so the removed parameters are ['radiusOfFirstHit', 'pixeldEdx', 'TRTTrackOccupancy', 'TRTdEdx', 'TRTd- EdxUsed-Hits', 'eProbabilityHT', 'eProbabilityComb', 'numberOfIBLOverflows- dEdx', 'numberOfUsedHitsd-Edx']. More data from various runs is needed in order to more closely look at the effects of variables in the LB datasets.

There are two possible variants in the structure of the LB datasets inputted into the GBDT that have been tested. This has been done to test whether the drift procedure necessitates the use of all information in an LB dataset, or whether a subset of information is sufficient to correctly flag information consistently with the flags contained in the GRLs.

1. Using a subset of tracks contained in each LB dataset then training the GBDTs. This variant of LB dataset structure hard-codes an equivalence in the size of the reference and of all the subject LB datasets. A potential flaw with this approach is that situations in which a particular LB datasets contain significantly fewer tracks may lead to operand mismatch errors. An additional flaw is that not all the available information is used so the results may be skewed. For training the classifiers in this case, we input 68 variables $\times$ 300 000 tracks.

2. Using all the tracks contained in each LB dataset then training the GBDTs. There are likely a different number of tracks in every LB dataset. This variant has an advantage over the first in that should an LB dataset with some LBN contain significantly fewer tracks compared to LB datasets with close-by LBNs, this could be an indicator that this LB dataset is bad. However, the computational demands on hardware are increased as compared to the first variant. For training

the classifiers in this case, we input 68 variables $\times$ All tracks. The total number of tracks in each LB dataset varies.

In this dissertation we implement the classification procedure using the second variant. This is done as follows:

1. The hyper-parameters of each GBDT for the LB comparisons are obtained by grid-searching possible values and obtaining those optimal for not distinguishing between the two LB datasets in our control LB comparison. The hyper-parameters may be limited as the LB datasets in the control LB comparison are the first two datasets that we have in the available set of data, but not the first two in the pseudo-smooth instantaneous luminosity regions described in section 7.2. This is due to a legacy iteration of DQEWS testing during which instantaneous luminosity was not considered. Due to the use of full LB datasets, a contingent solution of splitting the first LB dataset in the pseudo-smooth region would not be possible for grid searching on the reference LB dataset in the pseudo-smooth luminosity region. For a more uniquely calculated configuration of hyper-parameters, more data is necessary. However, this does not seem limiting to the results since separation is relative to the reference LB, and the GBDT does not over-train even with these nonunique hyper-parameters.

2. Training of GBDT models, according to equation (6.16) using the reference LB dataset in the control LB comparison as the reference for all LB comparisons with the subject LB datasets for each training instance being the subsequent LBs contained in the set of available LBs. The training procedure is illustrated in figure 7.3

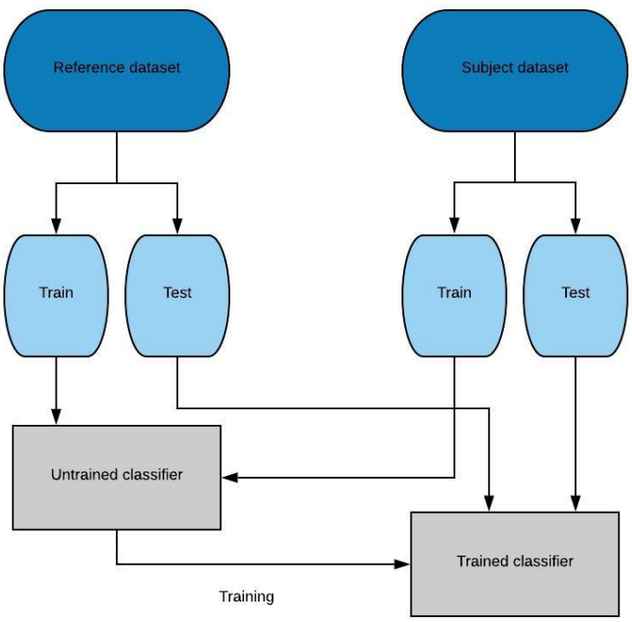


Figure 7.3: Flow of reference and subject LB dataset information in the GBDT training procedure.

7.3.2 Performance evaluations

Over-training

In evaluating the performance of the trained classifiers, it is most important that we implement tests that may output either numerical or visual representations that show signs of over or under-training. Over-training is a direct result of the hyper-parameters not being appropriately chosen for the data inputted into the untrained classifier. These hyper-parameters define the structure and behaviour of the classifier. It is necessary to test for over-training because the validity of results depends on the classifier outputting information that truthfully represents the inputted information by neither removing nor adding information through the introduction of noise.

Equivalent LB datasets should be indistinguishable [4] in that the vectors that span the space of the LB datasets should have approximately equivalent (according to the performance metrics that quantify separation) internal structure, and relationships with the other vectors contained. This equivalence in the distributions of two equivalent LB datasets means that a classifier trained to quantify the separation between them should be unable to distinguish them.

The test for over-training was implemented in the following way. We make use of a control LB comparison to test that the GBDT does not separate between the information in these LB datasets. Each run used in the analyses has a unique control LB comparison which has both the reference and subject LB classes as sub-sets of the same LB. The control LB comparisons make use of the first available LB datasets. The reference and subject LB datasets have LBN = 5 for Run-349268 and LBN = 656 for Run-357409. The reference and subject LB datasets for each run were provided already split (dataset splitting is not included in the methodology of these analyses):

Using the control LB-comparisons, we perform a grid-search on a list of possible hyper-parameters to determine those which define a classifier that does not separate between this control LB information for each run. The idea was that the simplest model structure be used that does not over-train in the control LB comparison whilst also sufficiently distinguishing between separate LB datasets.

The hyper-parameters were chosen by trial and error. This was done by inspecting the GBDT output probability histograms of the control LB comparison. By eye, we found a set of hyper-parameters that seemed to result in the least amount of over-training, shown by histograms for each class that overlapped as much as possible with corresponding performance metrics with values as close to describing indistinguishable binary classes as possible. We then made use of the grid-searching to optimise the values of

[4] They are splits of data from a LB with the same LB number identifier

these hyper-parameters to then maximise the indistinguishability between the LB datasets in the control LB comparison. This was done by setting the grid-search to obtain a GBDT error as close to 0.5 as possible. This is useful because the default classification threshold that defines the boundary at which a binary classifier is unable to distinguish compared datasets is 0.5 [97].

After grid-searching we obtained the hyper-parameters shown in table 7.1:

Hyper-parameter	Run-349268	Run-357409
Max-depth	5	3
N-estimators	120	120
Learning rate	1.2	1
Min-child-weight	2	3
Gamma	2	3

Table 7.1: Table presenting the optimal hyper-parameters obtained for each run using the grid-searching procedure described.

The descriptions of the hyper-parameters chosen are as follows (taken from XGBoost documentation [83]):

- **Max-depth**: The maximum depth of the tree. Larger max_depth results in a more complex model that is more likely to over-fit in training.
  Range: $[0, \infty)$
- **N-estimators**: Number of individual gradient boosted trees. Equivalent to the number of boosting rounds.
  Range: $[0, \infty)$
- **Learning rate**: Step size shrinkage used to prevent over-fitting. It does so as follows. After each boosting step, we can calculate the weights of features. The learning rate shrinks these weights. A larger value defines a more conservative algorithm.
  Range: [0, 1]
- **Min_child_weight**: The minimum sum of instance weight needed in a child node. If a tree partition step results in a leaf node with the sum of instance weight less than the defined value, the process halts. A larger value makes a more conservative algorithm.
  Range: $[0, \infty)$
- **Gamma**: Minimum loss reduction required to make a further partition on a leaf node of an individual tree. A larger value makes a more conservative algorithm.
  Range: $[0, \infty)$

7.3.3 Performance Metric Calculations

To quantify separation between LB information we have used the GBDT error. The GBDT error calculated using the GBDT output-probability arrays with values defined in equation (6.5) then using equation (6.10). This performance metric has been chosen because for increasing separation its value is decreasing. This allows us to implement the drift procedure in a way that makes obtaining flagging criteria for LB datasets that tighten over the course of a run easier due to convergence to an asymptote at 0 rather than some other number.

7.4 Drift Procedure Implementation

The drift procedure is implemented in such a way that the main idea is to consolidate the outputs from each LB comparison in the classification procedure.

After training our classifiers for each LB comparison, we calculate the performance metrics. We have done so explicitly using the GBDT error. These show a general trend of increasing degree of separation measured for each subsequent LB comparison over the course of a run. The instantaneous luminosity of the LB comparisons subject LB is used as the independent variable. Using this, we show that drift occurs over the course of a run as a function of the instantaneous luminosity.

A lower GBDT error corresponds to a higher degree of separation between compared LBs. This is consistently shown for our available LB datasets.

7.4.1 Data-Quality flagging algorithm

Prior to describing the DQ flagging algorithm, we must distinguish between the flagging of information on a run-by-run basis and a LB-by-LB basis. In the former case, the flagging requires that LB information used to construct the flagging criteria, as well as the testing LB information for an entire run are acquired and present at the time of analyses. In the latter case, only the LB information used to construct the flagging criteria is required prior to analyses where testing LB information may be inputted into the DQ flagging algorithm live. The DQ flagging algorithm described in this dissertation makes use of the latter method.

The main idea behind constructing the DQ flagging algorithm is that:

1. We make use of the sequence of the performance metrics calculated from the drift procedure to

determine the trajectory of the performance metrics. We use the trajectory to identify a general trend of separation from some reference LB over the course of the optimising runs available LBs to construct the flagging criteria. This trend is described by what is called the optimising curve.

2. We construct our flagging criteria using the optimising curve. Our flagging criteria are constructed to flag LB datasets as either 'bad' or 'good'. The LB datasets used to construct the flagging criteria are called the optimising sets.

3. We test the flagging criteria using the performance metrics calculated using LB information from a second run. The LB datasets used to test against the flagging criteria are called the evaluation sets.

Determining a trend

The averages instantaneous luminosity of each LB generally decreases as a run progresses while the GBDT errors of the LB comparisons generally increase. The instantaneous luminosity is used instead of the ΔLB because by using it we have more physics information about the ID tracking vectors contained in each LB dataset.

The idea is to determine a trend that approximates the trajectory of the drift of the acquired LB data over the course of a run. We plot points of GBDT error vs instantaneous luminosity. We then determine the trend-line from these points using the curve fitting function native to Scipy [98]. This function uses non-linear least squares minimisation to fit a user-defined function to the input data. In doing this curve fitting, we incorporate into all further analyses from this point onwards that our performance metrics are a function of the data at some instantaneous luminosity. We refer to this fitted curve as the optimising curve.

We would ideally determine the goodness of each of these fits to our data by performing a standard χ^2 goodness-of-fit hypothesis test in the following way. The hypotheses of our test, with a significance level of, $\alpha = 0.05$ would be as follows:

- H_0: Metrics are distributed in a way that the optimising curve sufficiently describes their relationship.
- H_a: Metrics are not distributed in a way that the optimising curve sufficiently describes their relationship.

The accept-reject conditions for our hypotheses would be as follows:

- Accept H_0 if p-value $< \alpha$ or $\chi^2 < \chi^2_{crit}$.
- Reject H_0 if the accept condition is not met.

However, due to our inability to obtain estimations of the uncertainties for the GBDT error points, we estimate the uncertainties (σ_i's) as the residuals between the data points and expectation from the best fit of the function in fig. 7.4, fig. 7.5, fig. 7.6, and fig. 7.7. The χ^2's are calculated using the estimated σ_i's, after the fact, based on the expected value of the fit in relation to the actual GBDT error points using $\chi^2 = \Sigma r_i^2/\sigma_i^2$, where r_i are the residuals for each point, and $r_i = \sigma_i$. This leads to estimations of χ_i^2 for all points being equal to 1 so that our χ^2/DoF = number of points/DoF. This was done because we do not have calculated uncertainties for the LB comparison GBDT models themselves. More work is needed to obtain these uncertainties. With proper uncertainty estimates, χ^2/DoF should be close to 1.

The above χ^2/DoF values are shown although given the construction of the σ's, the χ^2 values are not useful for goodness of fit hypothesis testing.

The fits for Run-349268 and Run-357409 are shown in that order. Within this order, the linear fits, and then the exponential fits, are shown.

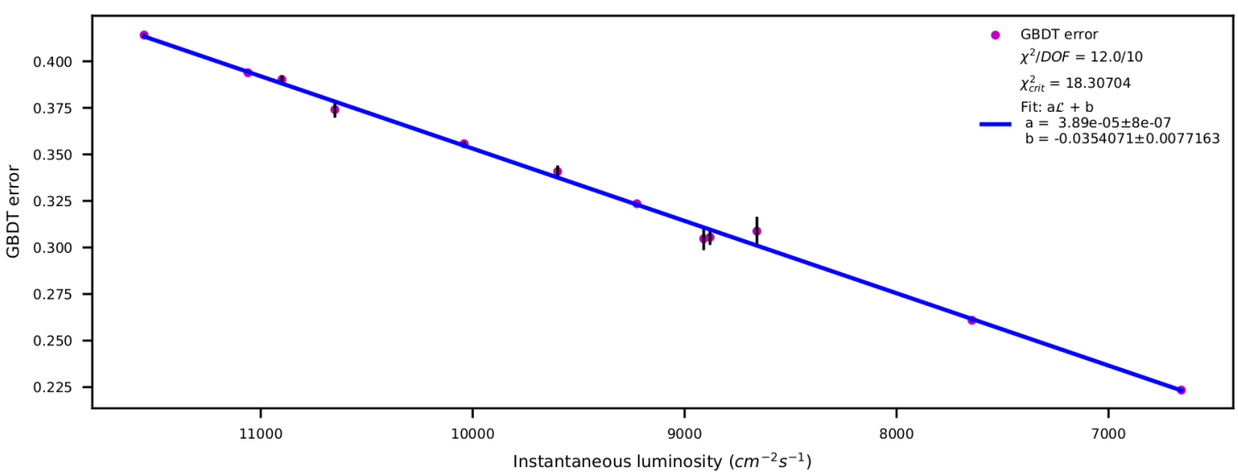


Figure 7.4: GBDT error plotted with linear function for each available LB in the pseudo-smooth instantaneous luminosity LBN domain in Run-349268.

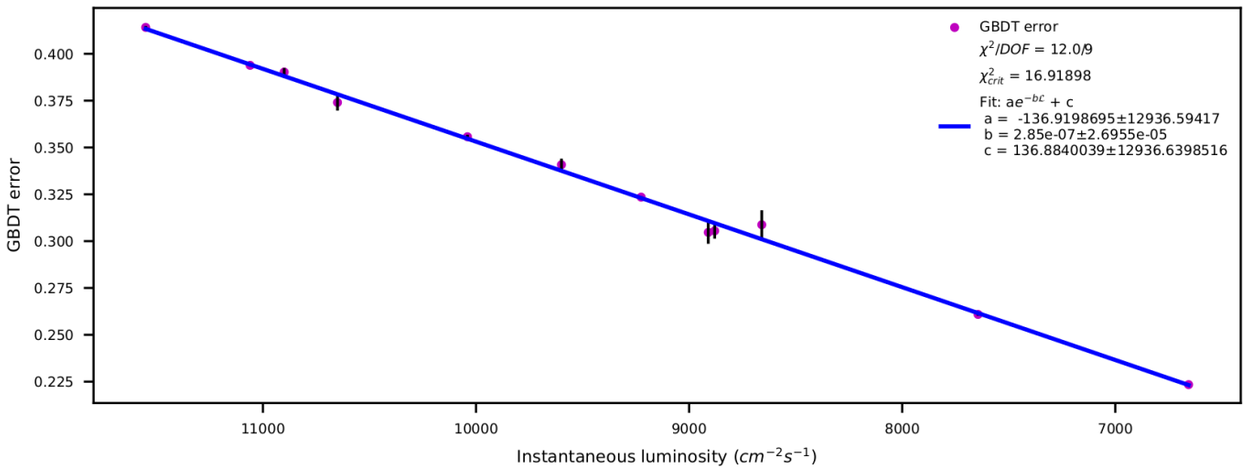


Figure 7.5: GBDT error plotted with exponential function for each available LB in the pseudo-smooth instantaneous luminosity LBN domain in Run-349268.

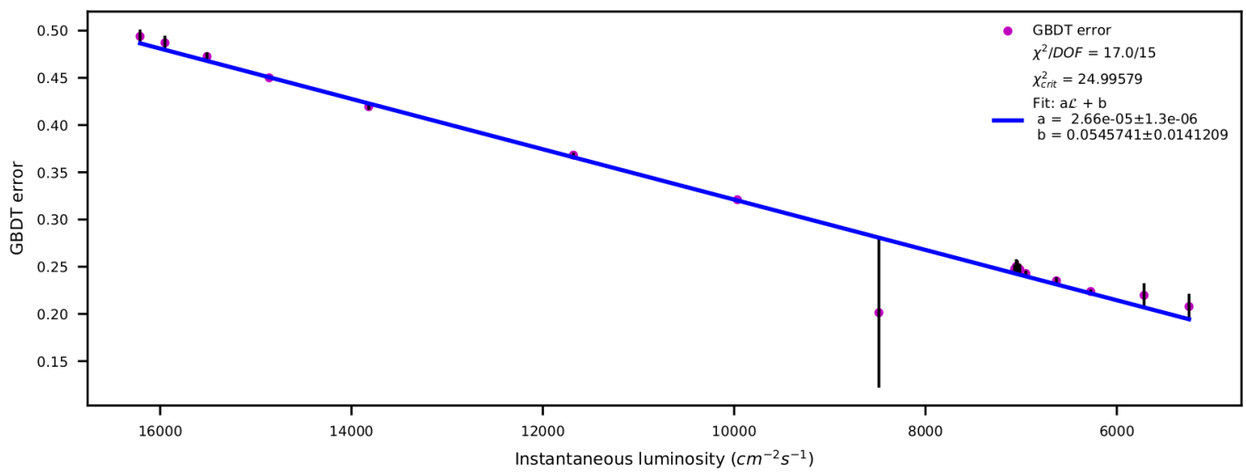


Figure 7.6: GBDT error plotted with linear function for each available LB in the pseudo-smooth instantaneous luminosity LBN domain in Run-357409.

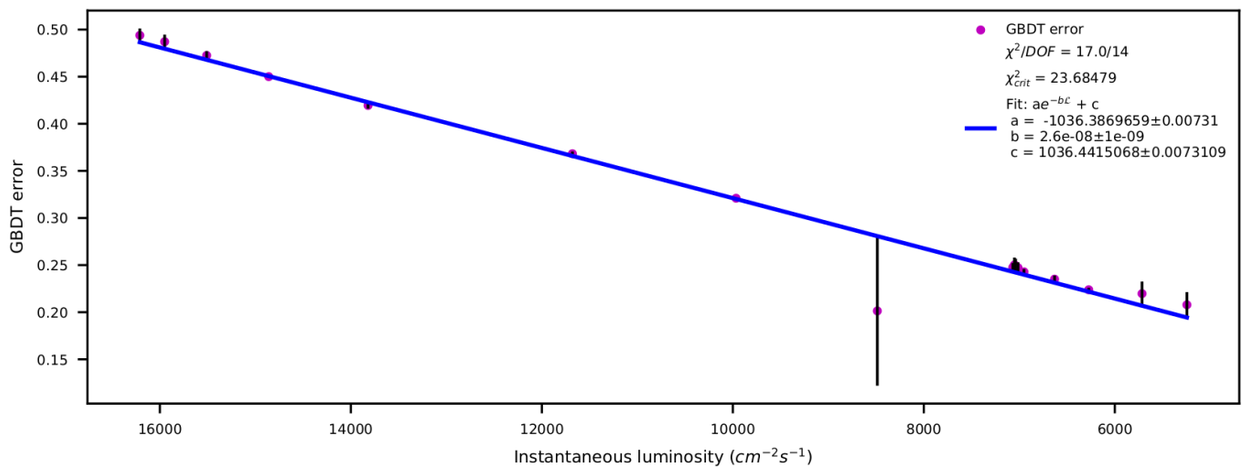


Figure 7.7: GBDT error plotted with exponential function for each available LB in the pseudo-smooth instantaneous luminosity LBN domain in Run-357409.

The Run-357409 χ^2/DoF for both the linear and exponential fits are lower than the Run-349268 χ^2/DoF, driven entirely by the difference between the number of points, since the number of free coefficients for

the fitted curves remains the same between runs.

Since b$\mathcal{L}$ is small (in both fig. 7.5 and fig. 7.7), by a first-order Taylor expansion, the exponential fit is not functionally different to the linear fit. We, therefore, opt to use the linear fit as the optimising curve for each run as it has the fewest number of coefficients that describe its trajectory and to simplify, slightly, the implementation of the DQEWS.

Constructing the flagging criteria

The flagging criteria are constructed using the trend determined by the performance metrics of the optimising runs' LB comparisons. This procedure may be applied for any chosen performance metric, but for these analyses we have used the GBDT error due to its relationship with the GBDT score and the fact that its value decreases as LB comparisons become more separated. This is helpful because dealing with functions that converge to 0 is generally easiest.

We refer to the set of GBDT errors for all the available LB comparisons used to determine the trend as the 'optimising data'. This trend is referred to as the 'optimising curve' for a run.

Obtaining the tightest possible flagging criteria using the information from a particular optimising run set may be done through Heuristic optimisation. In doing so, one may obtain a configuration of flagging criteria that bind the notion of 'goodness' to the behaviour exhibited by a particular known 'good' run. The idea behind constructing the flagging criteria borrows some principles from statistics to create a crude Heuristic method that flags our known good LBs as 'good'. The LB that is flagged is the subject LB in each LB comparison.

The general form of the equation the optimising curve, Ω, for the GBDT error with instantaneous luminosity as the independent variable that we have decided to use for simplicity of the DQEWS is:

$$\Omega = a \cdot \mathscr{L} + b \tag{7.2}$$

Ω_i denotes the optimising curve at an LB that has a particular instantaneous luminosity , $\mathscr{L}_i$ at LB comparison, i. The coefficients, 'a' and 'b' are decorrelated.

Once the optimising curve is determined, we use the function of this curve to obtain an approximate value of the given GBDT error at some instantaneous luminosity. We then obtain a distance from each of approximated GBDT error values that should flag good LB datasets as good. This is determined by upper and lower thresholds that are based on upper and lower curves obtained by appropriately adding

or subtracting 2σ errors of the optimising curve fits' coefficients. The sign of the operations for each coefficient is determined by the signs of the correlation coefficients contained in the covariance matrix for the curve fit coefficients. The distance between the upper and lower curves defines the bandwidth for acceptable data.

The upper curve is:

$$Upper = (a + 2\sigma_a) \cdot \mathscr{L} + (b + 2\sigma_b) \tag{7.3}$$

The lower curve is:

$$Lower = (a - 2\sigma_a) \cdot \mathscr{L} + (b - 2\sigma_b) \tag{7.4}$$

The optimising curves calculated in the analyses are monotonically increasing for the GBDT error for increasing instantaneous luminosity.

We additionally note that the flagging thresholds are two-sided. This means that there are thresholds that are calculated using the optimising curve that are both above and below the curve. This allows for the flagging of LB comparisons that deviate from the expected trend in both situations of being too separated, or not separated enough when compared to the surrounding LB comparisons. This is represented by a two-sided inequality. GBDT errors of the LB comparisons for a run that are contained within the extrema of this two sided inequality are flagged as 'good', and those which are not are flagged as 'bad'. The flags are placed in a list called the DQEWSGoodnessList.

Pseudo-code of the algorithm used to flag the chosen LB datasets is algorithm 1. The colours in this block of pseudo-code correspond to the colours of objects in the figures in section 7.4.1 and section 7.4.1 to come.

How close a given GBDT error for an LB comparison is to its corresponding flagging threshold gives the tightness of the flag. The tightness may be thought of as the degree of goodness of a particular LB. The tightness of the flag is defined as $Tightness_{\Delta LBcomparison_i}$ and is calculated with:

$$Tightness_{\Delta LBcomparison_i} = \frac{GBDTerror - \Omega_i}{(\frac{Upper_i - Lower_i}{2})} \tag{7.5}$$

Algorithm 1 DQEWS flagging algorithm

```
1: procedure DQEWSFLAGGING(p)          ▷ p is list of GBDT errors for LB comparisons in a run
2:     if Run = RunN then                                  ▷ Run to test is defined by user
3:         for i in GBDT error list do
4:             if lower <= GBDTerror <= upper  then
5:                 DQEWSGoodnessList.append('good')
6:             else
7:                 DQEWSGoodnessList.append('bad')
8:
```

The distance between a particular LB comparisons GBDT GBDT error and the point on the optimising curve at that LB comparisons instantaneous luminosity is in the numerator. Half of the band-with, defined by the distance between the upper or lower curves is in the denominator. LB comparisons flagged as 'good' have an absolute value of tightness below one. LB comparisons flagged as 'bad' have an absolute value of tightness above one. LB comparisons whose GBDT error lie above the optimising curve have a negative value of tightness, and LB comparisons whose GBDT error lie below the optimising curve have a positive value of tightness.

The tightness is a measure of how close a particular LB comparison is to being flagged as either 'good' or 'bad'. A tighter flag has an absolute value of tightness value closer to one, and a looser flag has a tightness value further from one.

The tightness is a helpful metric for the interpretation of the flags. This is because if a flag is good (or bad) but the magnitude of tightness for that LB comparison is clearly different to others near it, and is near the boundary between good and bad flags, that LB comparison may be assessed individually.

Flagging criteria have been constructed for each run. The flagging criteria are illustrated by showing the curve that describes the criteria and testing the criteria on the same run to illustrate the containment of the LB GBDT errors within the band of acceptable values for the GBDT errors to correspond to values that flag the LB comparison as 'good'.

Self-evaluating the flagging criteria on same run

The flagging criteria are tested by applying them to evaluation data. The evaluation data is structured in the same way as optimising data but is used as input to the algorithm rather than as internally loaded information in the algorithm's definition. There are three levels of flagging information. These are enumerated as:

1. It is important that we test that the flagging criteria constructed using an optimising runs' LB datasets flag these datasets contained within this run as good when using this same information as the testing run. This is that all good LB datasets from the LHC Run 2 data-taking period and run that the optimising curve is constructed with are flagged as good. This is a self-consistency check. This check is to determine whether the flagging criteria work when tested on precisely the same information. Criteria cannot fail this level of testing.

2. The criteria constructed using good LB datasets from the same LHC data-taking period (but not necessarily the same run) flag these LB datasets as good.

3. Using our criteria that pass at least one of the above conditions, we insert the bad LB datasets into the flagging Run set. Should criteria fail this level of testing, the criteria may simply be too tight and may be readjusted; or this may also indicate problems with the data either used to optimise or to evaluate the criteria.

To ensure self-consistency of the flagging criteria constructed using each runs data, we make use of two equally sized (same number of tracks) LB dataset splits from the same LBN. With the set of all the first LB dataset splits from equation (6.14), we construct the flagging criteria for this test. With the set of all the second LB dataset splits, we test these flagging criteria. By doing this, we create a situation in which which the flagging criteria constructed using a particular set of tracks are not tested on precisely the same set of tracks because the result of a test in this manner test would be trivially consistent, and this is not helpful. This is done to ensure that the flagging criteria work when testing them on equivalent but independent information.

The self-consistency checks for each of the runs are shown as follows:

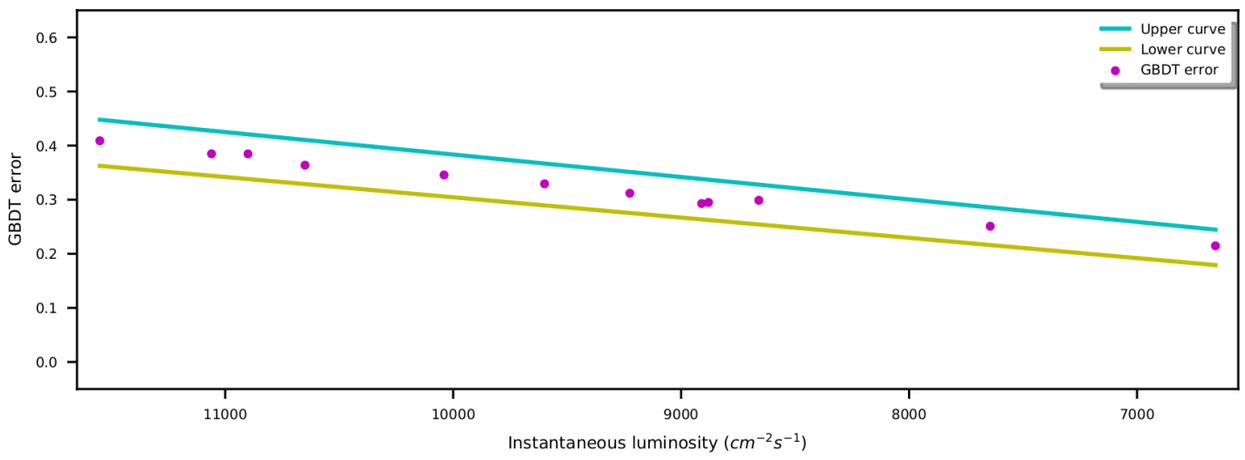


Figure 7.8: Plot for self-consistency check for Run-349268.

We see that for the self-consistency check of Run-357409 information the LB dataset with LBN = 1100 is flagged as bad. The fit of the optimising curve used to construct the flagging criteria used in this

LB comparison	Reference LB: Subject LB	Subject LB duration (s)	GBDT error	InstLumi	Flag	Tightness
1	lb0057 : lb0067	60	0.4089	11550	Good	0.08749706779263429
2	lb0057 : lb0090	60	0.3848	11060	Good	-0.024015369836695482
3	lb0057 : lb0110	60	0.3847	10900	Good	0.12463697967086156
4	lb0057 : lb0140	60	0.3638	10650	Good	-0.14332965821389196
5	lb0057 : lb0180	60	0.3457	10040	Good	0.004547176960970065
6	lb0057 : lb0209	60	0.3293	9599	Good	0.030753327303269157
7	lb0057 : lb0230	60	0.3119	9225	Good	-0.03769109119662625
8	lb0057 : lb0251	60	0.2929	8910	Good	-0.2147165259348613
9	lb0057 : lb0253	53	0.2949	8880	Good	-0.12860786682776212
10	lb0057 : lb0300	60	0.2989	8659	Good	0.21440217391304348
11	lb0057 : lb0400	60	0.2509	7644	Good	0.002302158273381295
12	lb0057 : lb0512	60	0.2147	6656	Good	0.08700961685238895

Table 7.2: Table for self-consistency check for Run-349268.

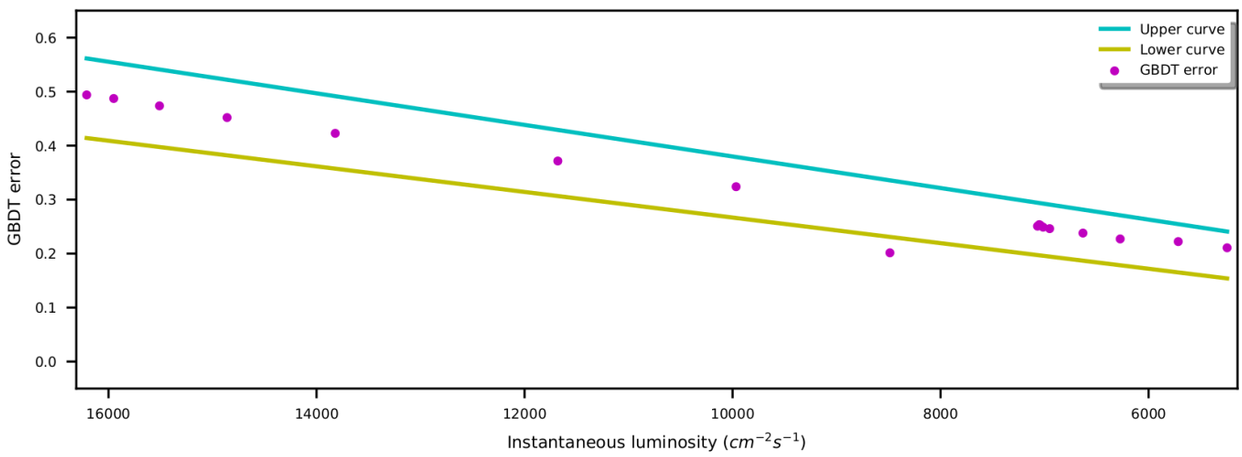


Figure 7.9: Plot for self-consistency check for Run-357409.

LB comparison	Reference LB: Subject LB	Subject LB duration (s)	GBDT error	InstLumi	Flag	Tightness
1	lb0672 : lb0673	60	0.4939	16210	Good	0.08464460119370593
2	lb0672 : lb0682	60	0.4872	15950	Good	0.08807615916718033
3	lb0672 : lb0697	60	0.4736	15510	Good	0.062404234573060303
4	lb0672 : lb0717	60	0.4519	14860	Good	-0.0002857346953353811
5	lb0672 : lb0750	60	0.4227	13820	Good	-0.025327771156138257
6	lb0672 : lb0870	60	0.3715	11680	Good	0.06110112726678647
7	lb0672 : lb0970	60	0.3239	9966	Good	0.026383355467020808
8	lb0672 : lb1100	10	0.2014	8487	Bad	-1.5617484252719986
9	lb0672 : lb1236	60	0.2508	7068	Good	0.1060449762739839
10	lb0672 : lb1237	60	0.2536	7051	Good	0.17387712958182758
11	lb0672 : lb1238	60	0.2521	7039	Good	0.14961768960529034
12	lb0672 : lb1240	60	0.2489	7014	Good	0.09706125827814568
13	lb0672 : lb1245	60	0.2462	6951	Good	0.07538940809968847
14	lb0672 : lb1270	60	0.238	6631	Good	0.08166719559928065
15	lb0672 : lb1300	60	0.2271	6273	Good	0.05380875202593193
16	lb0672 : lb1350	60	0.2222	5715	Good	0.27607019112551695
17	lb0672 : lb1400	60	0.2107	5245	Good	0.30712558996201217

Table 7.3: Table for self-consistency check for Run-357409.

evaluation includes this LB. The reason for its inclusion is that prior to implementing the classification procedure there are no obvious differences in the relationship between the LBN and the instantaneous luminosity of this LB in relation to the other available LBs. In addition to this, there are no intolerable defects that this LB has that the others do not.

When looking at the top 5 ranked variable attributions [5] for LB with LBN 1100 , and the next closest available LB datasets prior and posterior, we have the values shown in tables 7.4, 7.5 and 7.6.

[5]Calculated using the gain method native to the XGBoost software package

Rank	Relative importance (%)	Variable
1	11.68	numberOfSCTSharedHits
2	10.67	numberOfTRTTubeHits
3	8.80	cov02
4	8.15	numberDoF
5	5.29	numberOfSCTHoles

Table 7.4: Showing the 5 most important variables for differentiating LB 970 (7 244 633 tracks) from the reference LB.

Rank	Relative importance (%)	Variable
1	11.27	numberOfTRTTubeHits
2	11.19	numberDoF
3	9.76	numberOfSCTHoles
4	9.07	numberOfSCTSharedHits
5	7.85	numberOfPixelSharedHits

Table 7.5: Showing the 5 most important variables for differentiating LB 1100 (2 835 968 tracks) from the reference LB.

Rank	Relative importance (%)	Variable
1	12.46	numberOfTRTTubeHits
2	12.19	numberDoF
3	9.01	numberOfSCTHoles
4	8.87	cov04
5	6.99	numberOfSCTSharedHits

Table 7.6: Showing the 5 most important variables for differentiating LB 1236 (6 170 499 tracks) from the reference LB.

There are no indications of attribution information being a strong factor for the higher lower GBDT error computed for the LB dataset with LBN 1100

However, we see that the duration of LBN 1100, in relation to all the others used in the Run-357409 self-evaluation, is 10 seconds as opposed to 60 seconds. This is shown in 7.9. In addition to this, the total number of tracks contained in the LB dataset with LBN 1100 is 2 835 968. The LB dataset that contains the next fewest number of tracks has 5 713 268 tracks and has LBN 1300.

It is desirable to closely scrutinise the variable importances of LB 1100 to determine whether there are indeed any actual differences in the tracking variable relative importance values calculated by the GBDT. To this end, we look at density plots of the distributions of each of the top 5 variables attributed in the LB comparison flagged as 'bad', and compare them with density plots of the same variables in the adjacent LB comparisons.

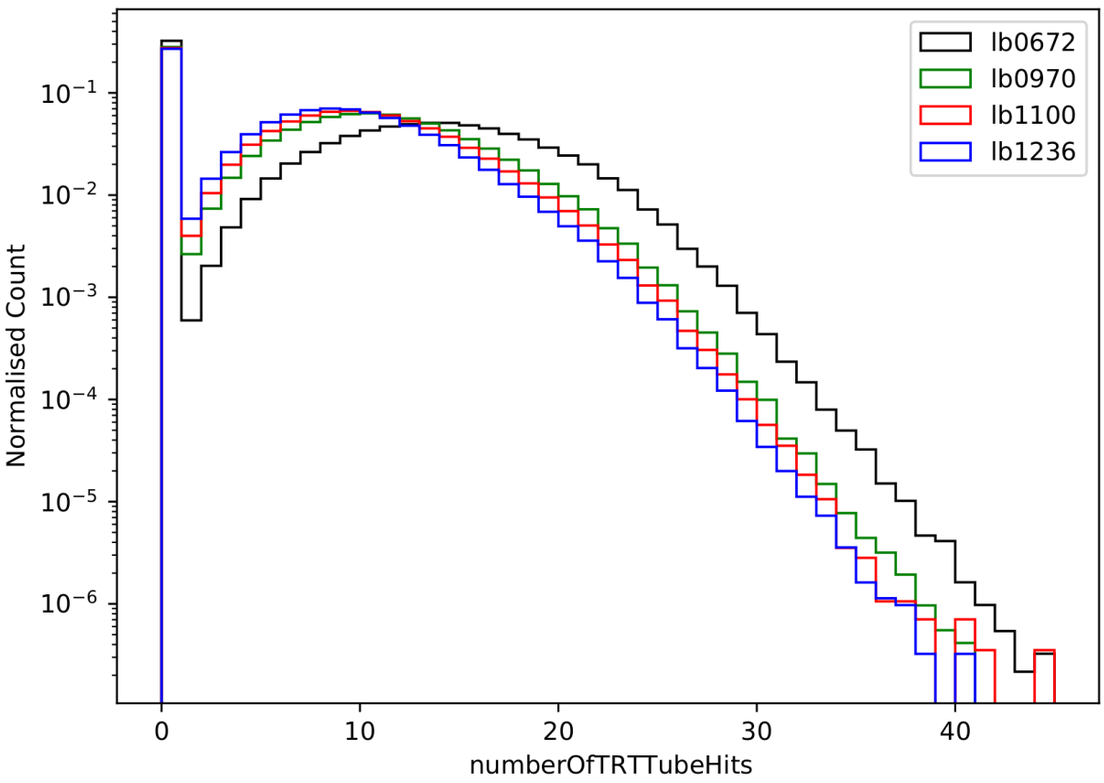


Figure 7.10: Rank-1 variable by relative importance of bad LB comparison calculated using 'gain' method density plot comparison. Compared variable is numberOfTRTTubeHits in LB 970, LB 1100 and LB 1236. Reference LB 672 is shown for comparison.

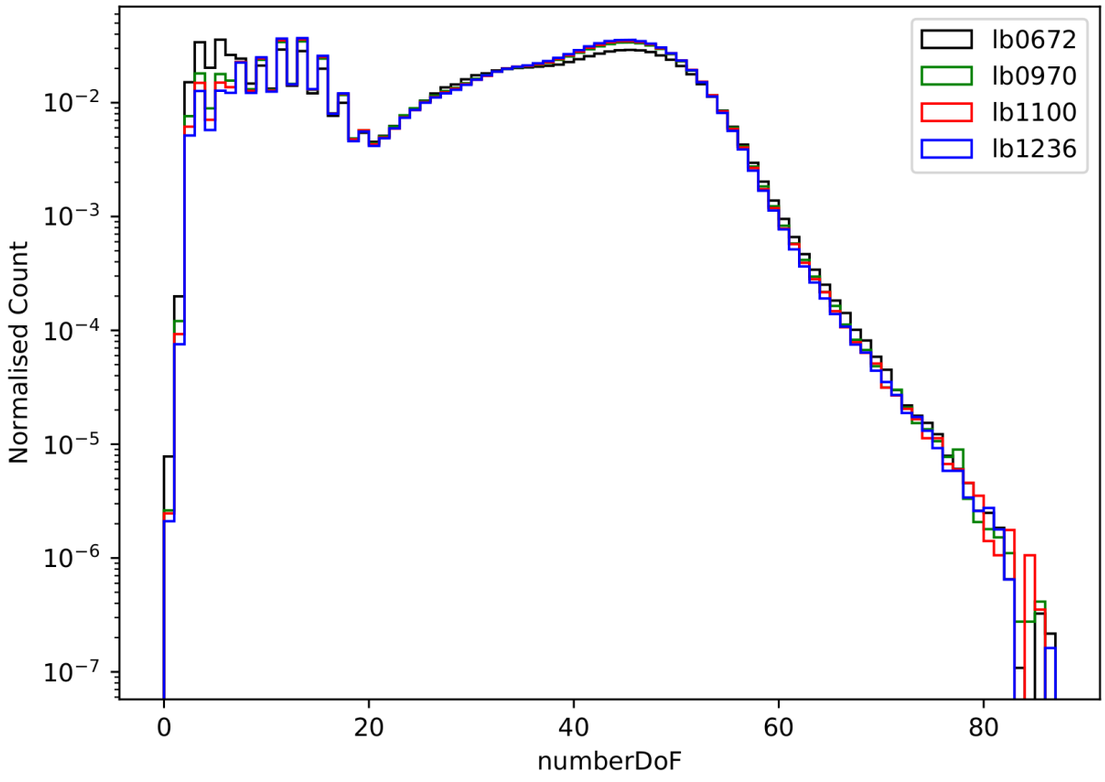


Figure 7.11: Rank-2 variable by relative importance of bad LB comparison calculated using 'gain' method density plot comparison. Compared variable is numberDoF in LB 970, LB 1100 and LB 1236. Reference LB 672 is shown for comparison.

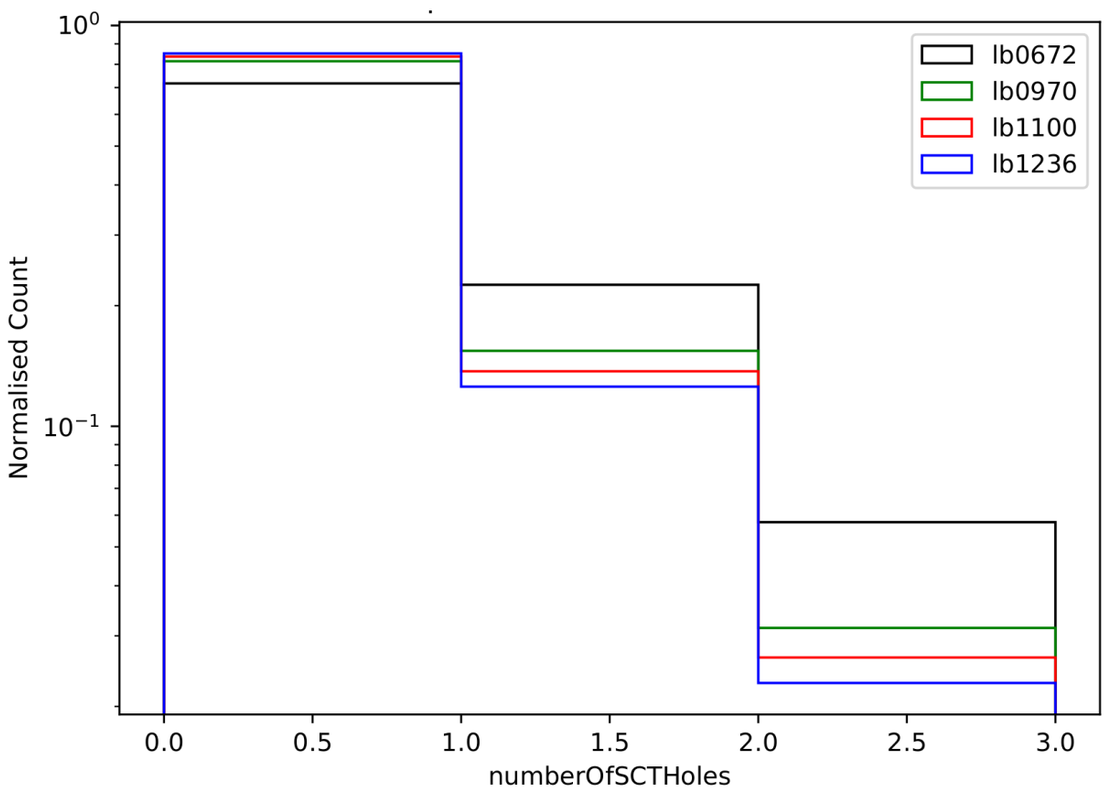


Figure 7.12: Rank-3 variable by relative importance of bad LB comparison calculated using 'gain' method density plot comparison. Compared variable is numberOfSCTHoles in LB 970, LB 1100 and LB 1236. Reference LB 672 is shown for comparison.

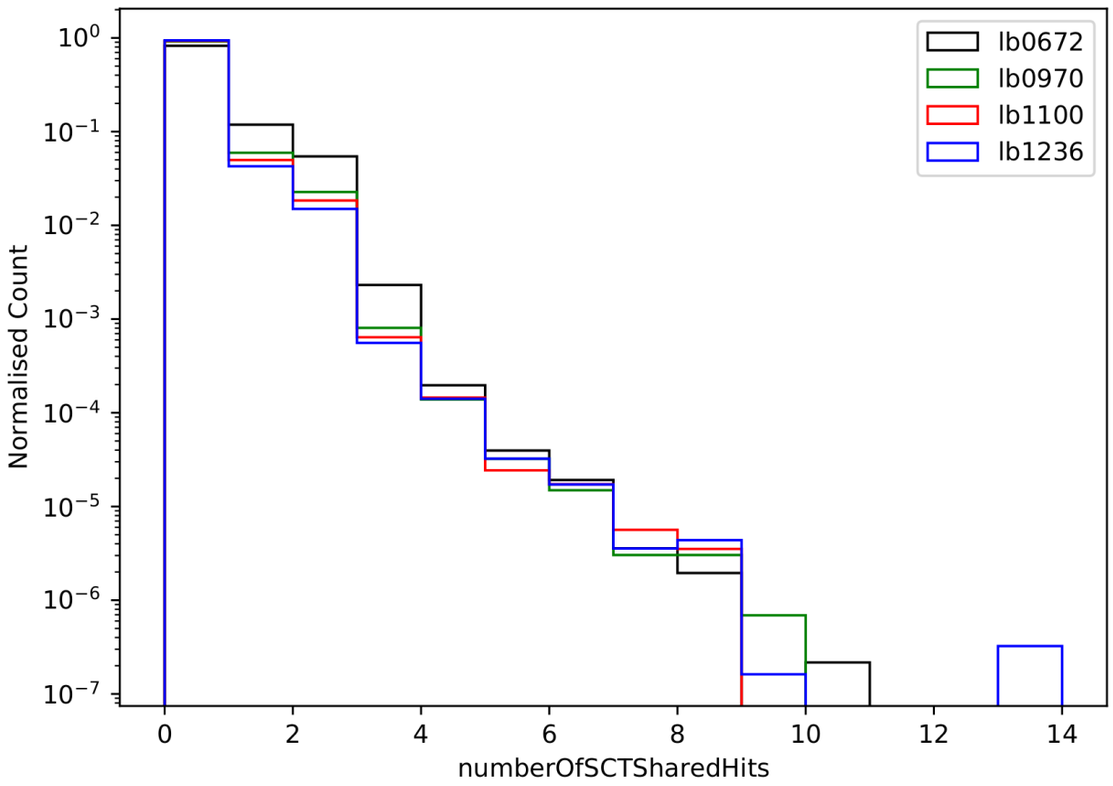


Figure 7.13: Rank-4 variable by relative importance of bad LB comparison calculated using 'gain' method density plot comparison. Compared variable is numberOfSCTSharedHits in LB 970, LB 1100 and LB 1236. Reference LB 672 is shown for comparison.

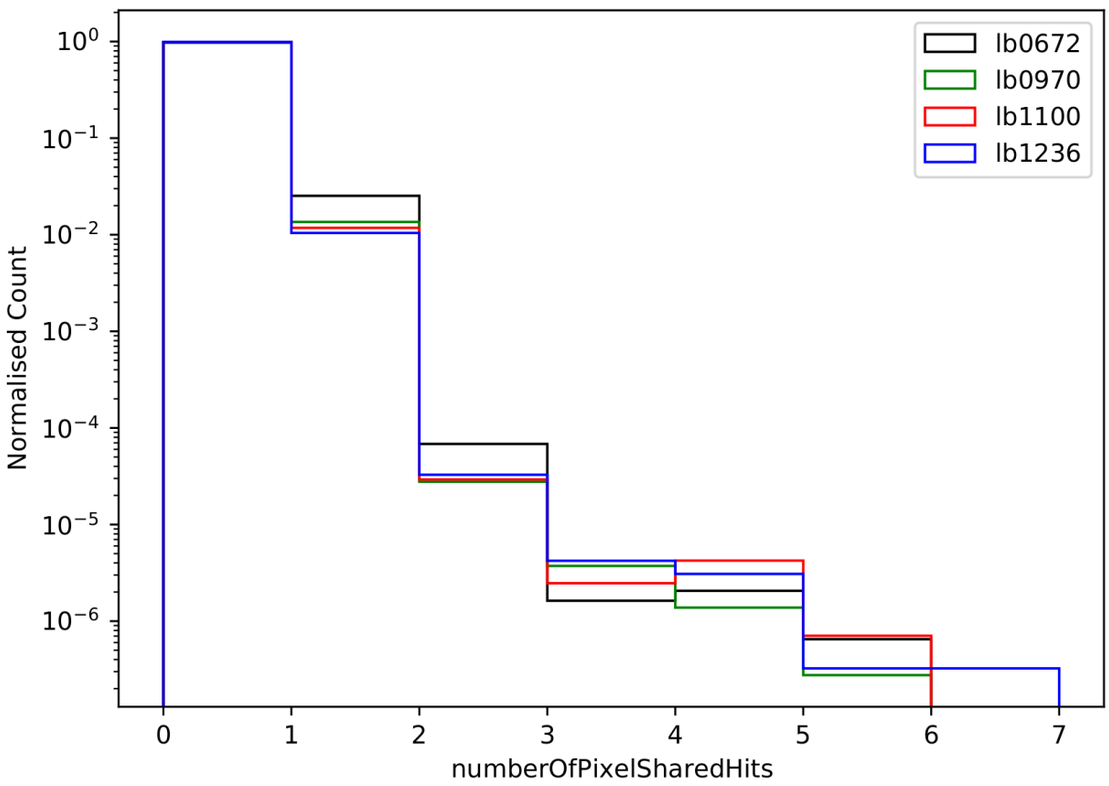


Figure 7.14: Rank-5 variable by relative importance of bad LB comparison calculated using 'gain' method density plot comparison. Compared variable is numberOfPixelSharedHits in LB 970, LB 1100 and LB 1236. Reference LB 672 is shown for comparison.

Looking at the density plots of the top 5 ranked variables of the 'bad' LB dataset and the closest available 'good' LB datasets prior and posterior to it (figures 7.10, 7.11, 7.12, 7.13 and 7.14), we do not see any significant differences in the plots that could indicate reasons for the 'bad' flag based on the shape alone.

When looking at the GBDT output probability distribution of the bad LB comparison, we see a slight difference in the shape of this distribution in relation to what we expect based on the shapes of the prior and posterior LB comparisons of the nearest available LBs. We show the histograms to illustrate this in figure 7.15, figure 7.16 and figure 7.17.

The shift in the GBDT output probability distribution as seen in 7.16 reflects the vastly differed sizes of training data for the reference and subject LBs. This is that the tree learns that it is "always more likely" for a track to be from the reference LB.

Figure 7.15: GBDT output probability histogram as a density plot (area under the curve normalised to 1) for LB 970 (7 244 633 tracks).

The sensitivity of the algorithm with respect to the choice of hyper-parameters for the self-evaluations is insignificant in these self evaluation tests. Results supporting this statement are shown in B.1

Cross-evaluating flagging criteria between runs

We then perform tests on LB datasets between runs. For Run-2 we test the cases where the good LB datasets available from Run-349268 or Run-357409 are used as the optimising data, and then the LB datasets from the other run, including any bad LB datasets, are used as the evaluation data.

The only LB dataset that we have that is flagged as bad by the GRL with the issues being with ID tracking variables is the LB dataset in Run-349268 with LBN 252. The results for the cross-evaluation

Figure 7.16: GBDT output probability histogram as a density plot (area under the curve normalised to 1) for LB 1100 (2 835 968 tracks).

Figure 7.17: GBDT output probability histogram as a density plot (area under the curve normalised to 1) for LB 1236 (6 170 499 tracks).

where available Run-349268 LB datasets (excluding the bad LB dataset) are used as the optimising data and the available Run-357409 LB datasets are used as evaluating data are shown in figure 7.18:

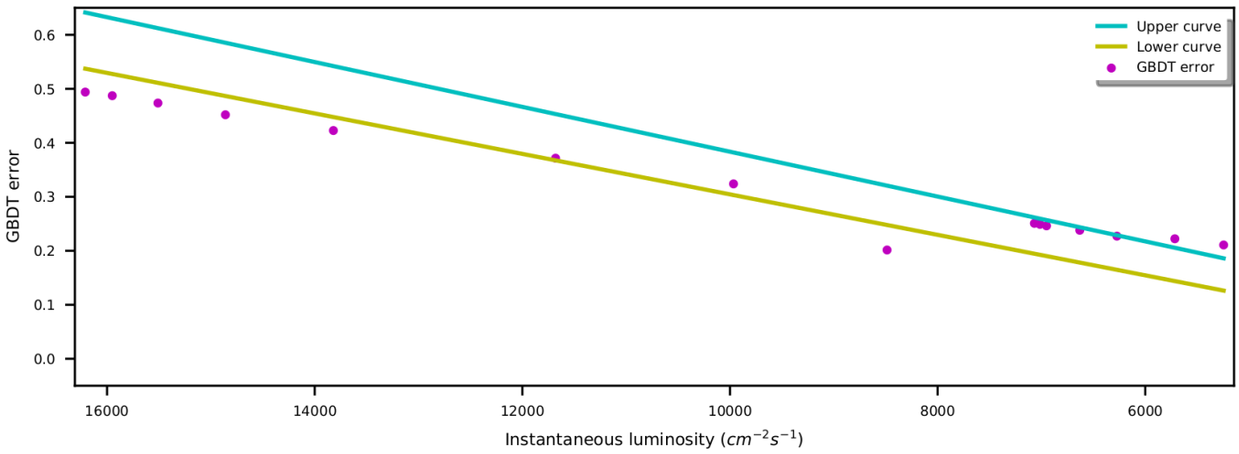


Figure 7.18: Plot showing flagging criteria constructed using Run-349268 optimising LB datasets and tested against Run-357409 LB evaluating datasets.

LB comparison	Reference LB: Subject LB	Subject LB duration (s)	GBDT error	InstLumi	Flag	Tightness
1	lb0672 : lb0673	60	0.4939	16210	Bad	-1.8327887757063233
2	lb0672 : lb0682	60	0.4872	15950	Bad	-1.781962916221726
3	lb0672 : lb0697	60	0.4736	15510	Bad	-1.7384433030422757
4	lb0672 : lb0717	60	0.4519	14860	Bad	-1.7040868066119055
5	lb0672 : lb0750	60	0.4227	13820	Bad	-1.5282779072230461
6	lb0672 : lb0870	60	0.3715	11680	Good	-0.9051060853345769
7	lb0672 : lb0970	60	0.3239	9966	Good	-0.4747052111068848
8	lb0672 : lb1100	10	0.2014	8487	Bad	-2.2705075445816183
9	lb0672 : lb1236	60	0.2508	7068	Good	0.6757963679666568
10	lb0672 : lb1237	60	0.2536	7051	Good	0.7808076292653852
11	lb0672 : lb1238	60	0.2521	7039	Good	0.7506710408589324
12	lb0672 : lb1240	60	0.2489	7014	Good	0.6854838709677419
13	lb0672 : lb1245	60	0.2462	6951	Good	0.6808574426622696
14	lb0672 : lb1270	60	0.238	6631	Good	0.8289252407888702
15	lb0672 : lb1300	60	0.2271	6273	Good	0.950758167891199
16	lb0672 : lb1350	60	0.2222	5715	Bad	1.54050550874919
17	lb0672 : lb1400	60	0.2107	5245	Bad	1.8271481109996657

Table 7.7: Table for flags for Run-349268 optimising LB datasets used to construct the flagging criteria tested against Run-357409 LB evaluating datasets.

For the results shown in figure 7.18 we note that the instantaneous luminosities of the Run-349268 optimising set lie have the extrema $\{6656\ ,\ 11550\}cm^{-2}\cdot s^{-1}$. The instantaneous luminosities of the Run-357409 evaluating set have the extrema, $\{5245\ ,\ 16210\}cm^{-2}\cdot s^{-1}$.

The LB comparisons that are flagged correctly, with the exception of LBN 1100, in this cross-evaluation are those that lie within the instantaneous luminosity range of $[6273\ ,\ 11680]cm^{-2}\cdot s^{-1}$. The extrema of the subset of good flags in the evaluating set are numerically very similar to the extrema of the optimising set. The infimum of the subset of good flags in the evaluating sets' instantaneous luminosity range is contained in the optimising sets' instantaneous luminosity range. All bad flags, with the exception of LBN 1100, lie outside of the optimising sets' instantaneous luminosity range.

It seems that when evaluating the flagging criteria, extrapolation outside of the instantaneous luminosity range of the optimising sets data does not work with the LB data-sets used for analyses. It is possible that with a larger range of LBs available, the optimising curve fits may have been better such that extrapolation may have been more successful.

The results for the cross-evaluation where available Run-357409 LB datasets are used as the optimising data and the available Run-349268 LB datasets (including the bad LB dataset) are used as evaluating data are shown in figure 7.19:

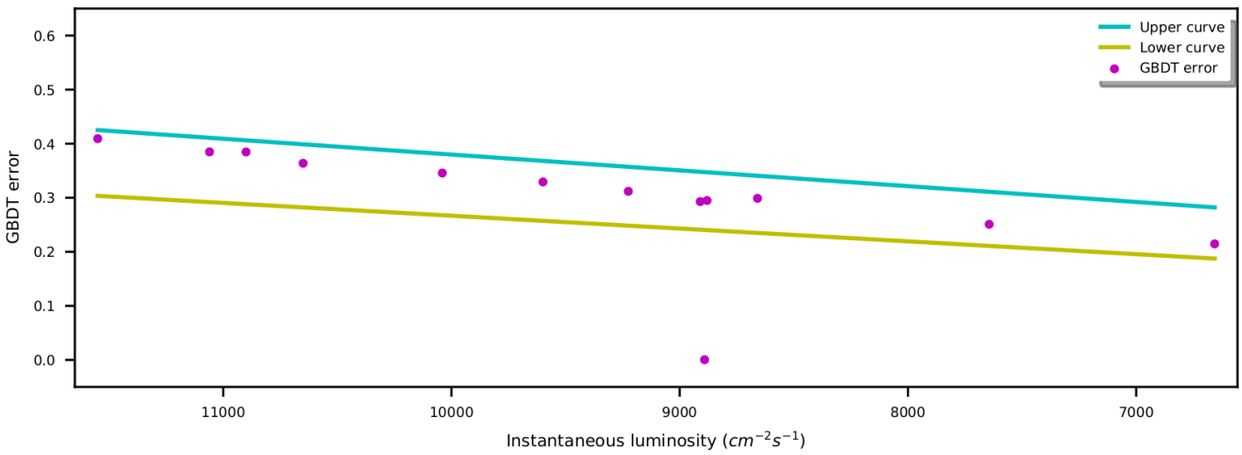


Figure 7.19: Plot showing flagging criteria constructed using Run-357409 optimising LB datasets and tested against Run-349268 LB evaluating datasets.

LB comparison	Reference LB: Subject LB	Subject LB duration (s)	GBDT error	InstLumi	Flag	Tightness
1	lb0057 : lb0067	60	0.4094	11550	Good	0.7401807723911257
2	lb0057 : lb0090	60	0.3848	11060	Good	0.5625682830489958
3	lb0057 : lb0110	60	0.3847	10900	Good	0.6356786046905426
4	lb0057 : lb0140	60	0.3638	10650	Good	0.39886918529940885
5	lb0057 : lb0180	60	0.3457	10040	Good	0.3768525052928723
6	lb0057 : lb0209	60	0.3293	9599	Good	0.29967544175982685
7	lb0057 : lb0230	60	0.3119	9225	Good	0.16773837957009002
8	lb0057 : lb0251	60	0.2929	8910	Good	-0.029126213592233007
9	lb0057 : lb0252	16	0.0004	8891	Bad	-5.486590038314176
10	lb0057 : lb0253	53	0.2949	8880	Good	0.02374941561477326
11	lb0057 : lb0300	60	0.2989	8659	Good	0.2094013052113875
12	lb0057 : lb0400	60	0.2509	7644	Good	-0.19916100679184978
13	lb0057 : lb0512	60	0.2147	6656	Good	-0.4242104151262279

Table 7.8: Table for flags for Run-357409 optimising LB datasets used to construct the flagging criteria tested against Run-349268 LB evaluating datasets.

For the results shown in figure 7.19 we note that the instantaneous luminosities of the Run-357409 optimising set have the extrema, $\{5245\ ,\ 16210\} cm^{-2} \cdot s^{-1}$. The instantaneous luminosities of the Run-349268 evaluating set lie have the extrema $\{6656\ ,\ 11550\} cm^{-2} \cdot s^{-1}$.

All the LB comparisons are flagged correctly in this cross-evaluation. The extrema of the instantaneous luminosities of our evaluating set are all contained in the set of instantaneous luminosities bounded by the extrema of the optimising set. Thus, in this cross-evaluation, rather than extrapolating our flagging criteria for instantaneous luminosity values smaller and larger than the extrema of the optimising set described above, we are interpolating them to values between. This cross-evaluation supports our hypothesis that extrapolation outside of the optimising set instantaneous luminosity range does not work, while interpolation does. More data and tests are needed to more completely test this hypothesis.

The LB comparison that is flagged as bad has the only available subject LB dataset in the evaluating set that has intolerable defects. It has the following intolerable defects (all indicate failures in pixel modules):

1. Pixel_Barrel_Standby.
2. Pixel_EndcapA_Standby.
3. Pixel_EndcapC_Standby.
4. Pixel_IBL_Standby.
5. Pixel_Layer0_Standby.

Taking a look at the variable attributions for the bad LB and the prior, and posterior LBs in a similar fashion to the bad LB identified in section 7.4.1 we have:

Rank	Relative importance (%)	Variable
1	18.74	numberOfSCTSharedHits
2	7.32	numberOfTRTSharedHits
3	5.81	numberDoF
4	5.01	numberOfPixelSharedHits
5	4.67	numberOfSCTHoles

Table 7.9: Showing the 5 most important variables for differentiating LB 251 ()7 231 602 tracks) from the reference LB.

Rank	Relative importance (%)	Variable
1	3.64	numberOfPixelSharedHits
2	3.61	z0
3	3.14	numberOfTRTSharedHits
4	2.60	numberOfSCTHits
5	2.37	expectNextToInnermostPixelLayerHit

Table 7.10: Showing the 5 most important variables for differentiating LB 252 (2 971 tracks) from the reference LB.

The top-ranked variable attributed in the bad LB comparison is numberOfPixelSharedHits. This is somewhat consistent with the intolerable defects of this same LB. However, the relative importance of this top variable is only 3.64 % and is not much different to the next four. This suggests that the separation is being computed using all the variables. This reduces our confidence in the top-ranked variables' attribution value. This is because its' similarity in value to the remaining top-ranked variables means that

Rank	Relative importance (%)	Variable
1	13.94	numberOfSCTSharedHits
2	7.41	numberOfTRTSharedHits
3	6.24	numberDoF
4	5.45	numberOfSCTHits
5	4.77	numberOfPixelSharedHits

Table 7.11: Showing the 5 most important variables for differentiating LB 253 (7 751 098 tracks) from the reference LB.

this variable does not more strongly contribute to the DQEWS flag than any of the others, but is one of two that suggest failures in pixel modules. The second that suggests a failure in pixel modules is expectNextToInnermostPixelLayerHit with a relative importance of 2.37 % at rank 5. With this said, it seems as though all of the variables become less important with the key point being that all the variables are driving the separation quantified between the LBs.

We look at the density plots in a similar fashion to those shown in section 7.4.1.

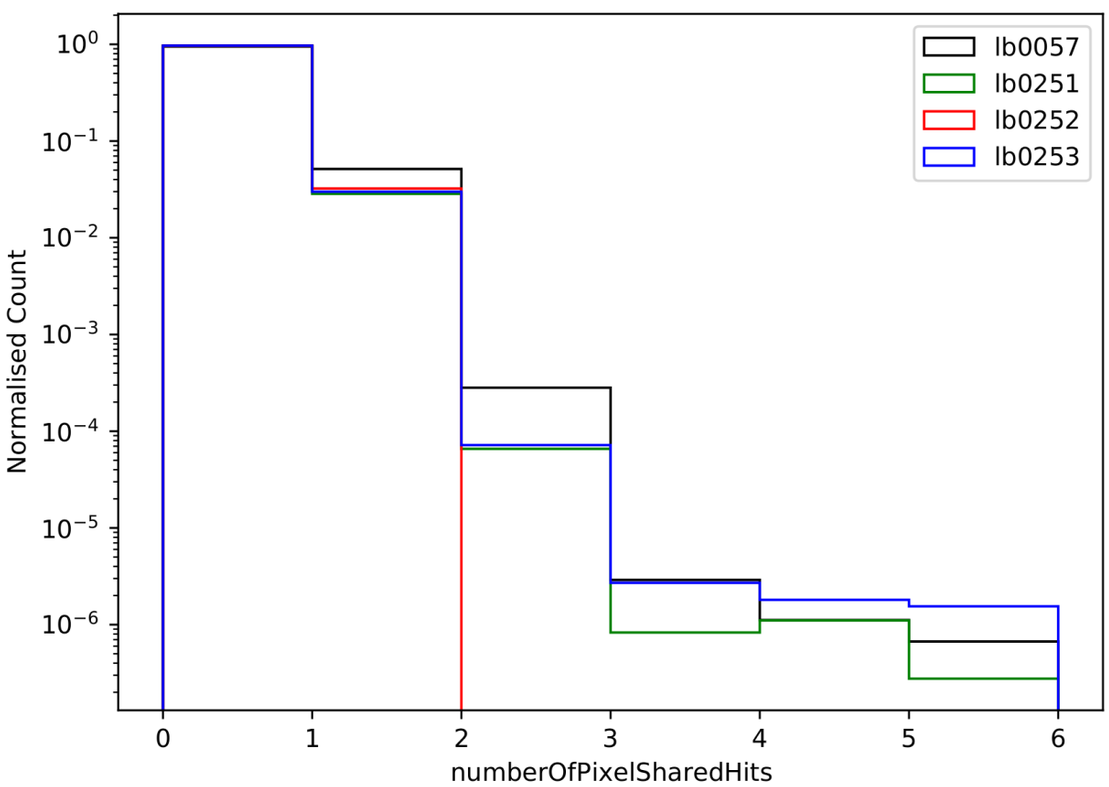


Figure 7.20: Rank-1 variable by relative importance of bad LB comparison calculated using 'gain' method density plot comparison. Compared variable is numberOfPixelSharedHits in LB 251, LB 252 and LB 253. Reference LB 57 is shown for comparison. Plot has broken axes to remove areas in plot that do not give any information.

Looking at the shapes of density plots of the top 5 ranked variable distributions in figures 7.20, 7.21,

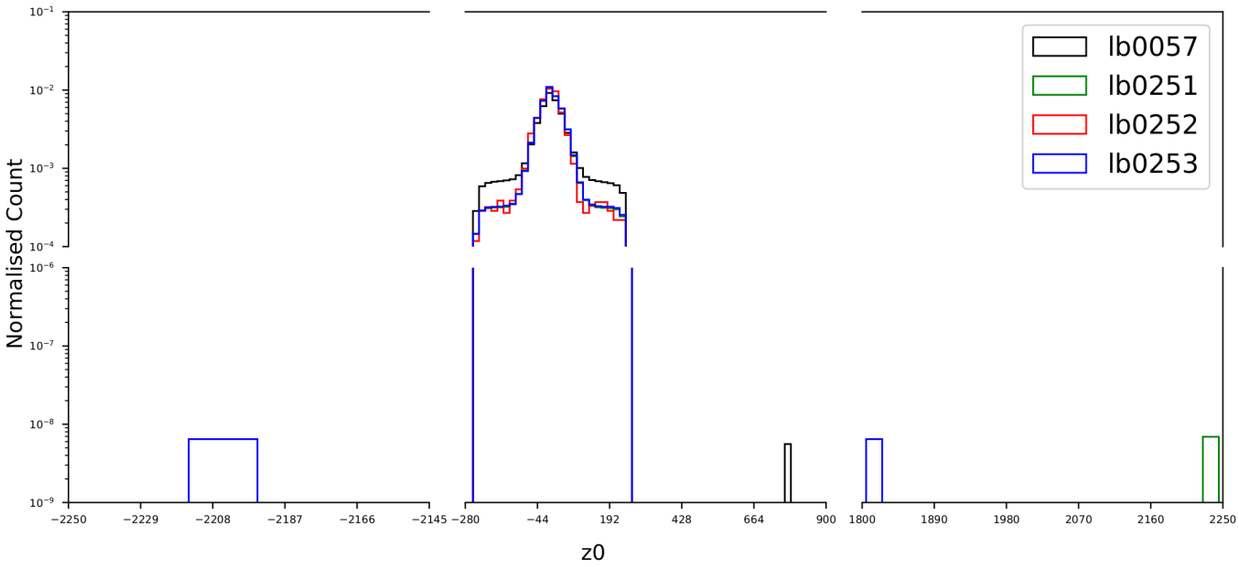


Figure 7.21: Rank-2 variable by relative importance of bad LB comparison calculated using 'gain' method density plot comparison. Compared variable is z0 in LB 251, LB 252 and LB 253. Reference LB 57 is shown for comparison.

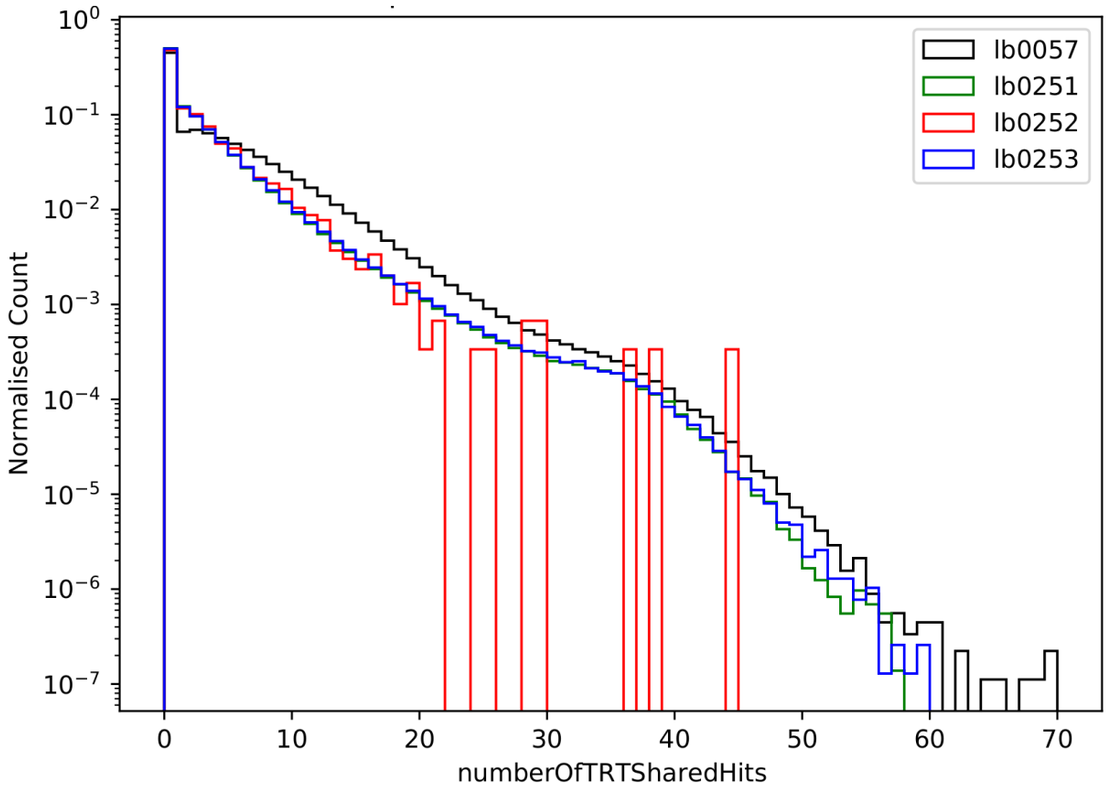


Figure 7.22: Rank-3 variable by relative importance of bad LB comparison calculated using 'gain' method density plot comparison. Compared variable is numberOfTRTSharedHits in LB 251, LB 252 and LB 253. Reference LB 57 is shown for comparison.

7.22, 7.23 and 7.24 the following is observed:

1. The variable that has the most similar distributions between the three adjacent LBs is that of expectNextToInnermostPixelLayerHit (rank 5). The density plots for this variable seem to be precisely

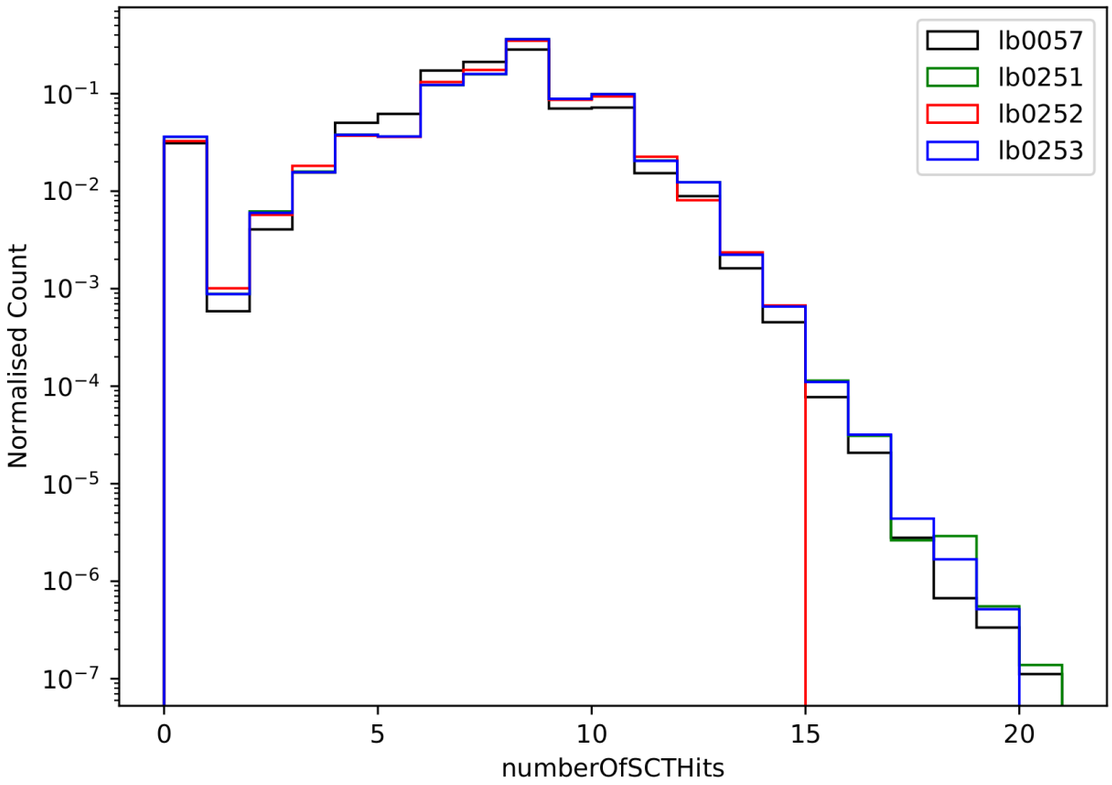


Figure 7.23: Rank-4 variable by relative importance of bad LB comparison calculated using 'gain' method density plot comparison. Compared variable is numberOfSCTHits in LB 251, LB 252 and LB 253. Reference LB 57 is shown for comparison.

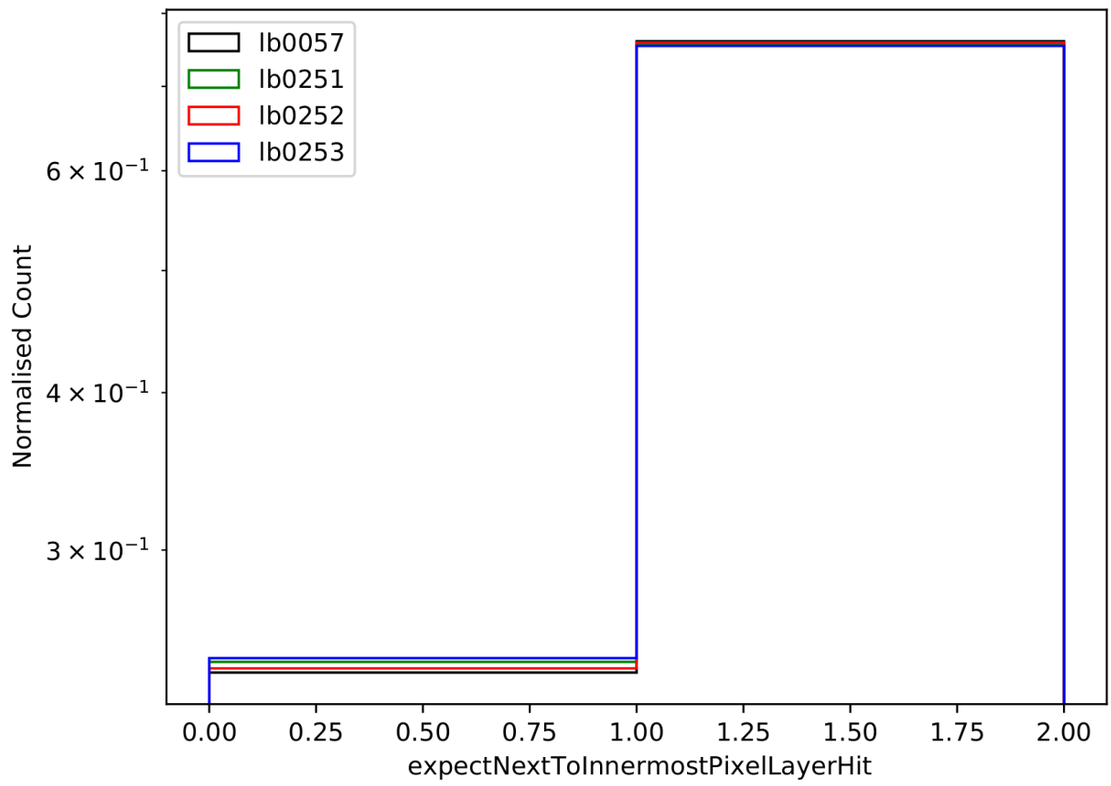


Figure 7.24: Rank-5 variable by relative importance of bad LB comparison calculated using 'gain' method density plot comparison. Compared variable is expectNextToInnermostPixelLayerHit in LB 251, LB 252 and LB 253. Reference LB 57 is shown for comparison.

the same. This is shown in figure 7.24.

2. The variables that have the next most similar distributions are numberOfPixelSharedHits (rank 1), z0 (rank 2), and numberOfSCTHits (rank 4). This is shown in figure 7.20, figure 7.21 and figure 7.23.

3. The variable that has the most dissimilar distribution of Lb 252 in relation to LB 251 and LB 253 is that of numberOfTRTSharedHits (rank 3). This is shown in figure 7.22. This figure shows that after a value of 20 for numberOfTRTSharedHits, the distribution for LB 252 differs significantly from that of LB 251 and LB 253.

By observing these distributions, a shifter should be able to identify this bad LB based on the differences in the shapes of the ranked variables shown when compared to the adjacent good LBs. We note however, while there are differences that could point to problems as described in the above enumeration (particularly in points 2 and 3), it is unclear whether these differences are due to actual differences in the behaviour of the detector components at this time, or whether they are a consequence of the severely limited number of tracks, or LB duration, in LB 252 as compared to LB 251 and LB 253. The severely limited number of tracks in LB 252 opens the possibility that the shape of the histograms of the variable distributions could smooth out and converge to the shapes of the histograms for LB 252 and 253 in the event that statistics were increased. This suggests that future attempts at early DQ monitoring must take into account the length of the LB under study, an easily implemented fix to reduce this type of false positive.

The reasons for the differences in the shapes of the histograms of the variable distributions are unclear. Despite the reasons being unclear for the top 5 variables using the 'gain' attribution method, Lb 252 is flagged as 'bad' correctly. This suggests that studies on the effects of dataset sizes (and by extension, lengths) and the methods used to obtain attribution values may be helpful. The goal is thus to obtain an unambiguous procedure such that, in addition to correctly flagging LBs, the DQEWS is then able to correctly attribute tracking variables that point to actual locations of failure in the detector while explicitly taking into account the number of tracks in an LB.

The flagging criteria directly compare the GBDT error, and the difference between good and bad flags is based only on the value of the GBDT error. The bad LB is more separated from the reference LB than the other LBs within the evaluation run. When looking at the GBDT output probability distribution of the bad LB comparison, we do however see a stand-out difference in the shape of this distribution in relation to what we expect based on the shapes of the prior and posterior LB comparisons of adjacent LBs. We show the histograms to illustrate this in figure 7.25, figure 7.26 and figure 7.27.

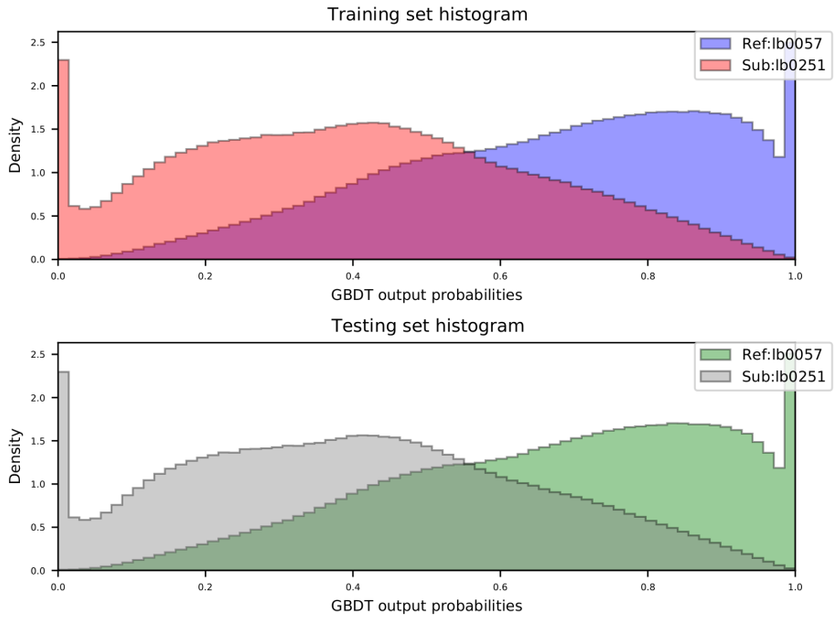


Figure 7.25: GBDT output probability histogram as a density plot (area under the curve normalised to 1) for LB 251 (7 231 602 tracks).

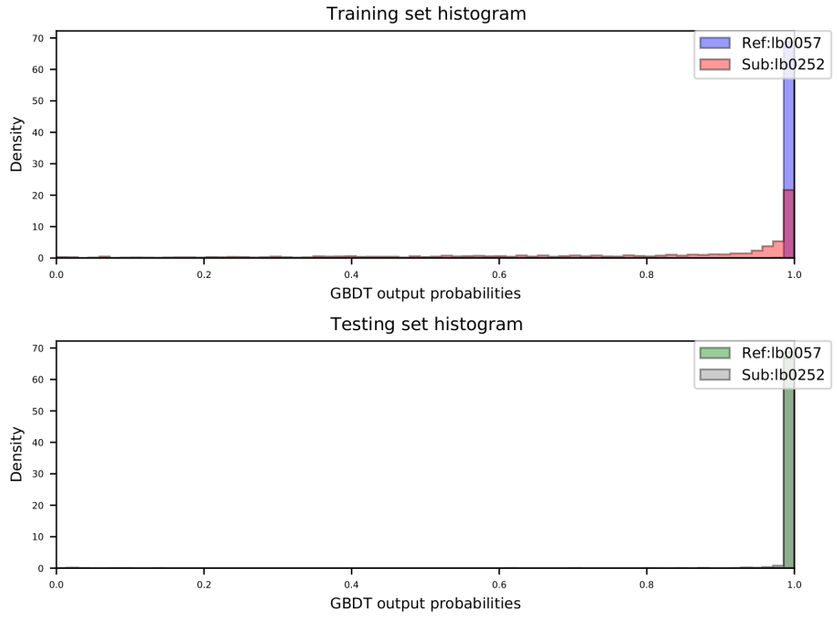


Figure 7.26: GBDT output probability histogram as a density plot (area under the curve normalised to 1) for LB 252 (2 971 tracks).

The sensitivity of the DQEWS algorithm with respect to the choice of hyper-parameters for the cross-evaluations is insignificant. All LB datasets contained within the instantaneous luminosity range of the optimising LB datasets are flagged correctly. This is shown in appendix B.2. This supports the notion that extrapolation of the flagging criteria outside of this range does not work.

This opens up new questions about the method of the DQEWS and the best ways to construct flagging criteria that identify real reasons of failures in the ATLAS detector system consistently with the particular method used for variable attribution. Introducing usage of both an effective attribution method to

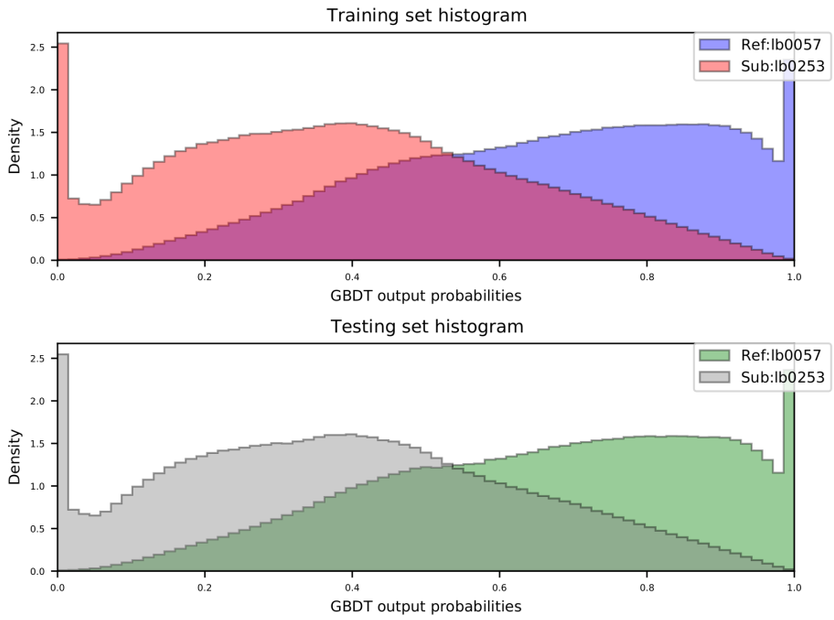


Figure 7.27: GBDT output probability histogram as a density plot (area under the curve normalised to 1) for LB 253 (7 751 098 tracks).

improve model explainability, and GBDT output probability distribution shape analyses in addition to the performance metric analyses may be helpful to begin understanding the full extent of the behaviour that results in a particular flag. Fitting continuous distributions to the histograms for the GBDT output probability distributions and observing the temporal behaviour of either one, or some combination of the distribution variables may be a method that may lead to a more comprehensive and correct flagging procedure for the DQEWS.

8

Conclusion & Future Work

Flagging criteria have been constructed using the GBDT outputs of LB comparisons that test for separation between LB datasets from Run-349268 and Run-357409 of the ATLAS detector that have been flagged as good in the GRLs, by the DQ group, that represent a subset of ID tracking information contained within a LB. The set of LB datasets inputted to the LB comparisons used to construct the criteria has been termed the optimising set. These flagging criteria have then been evaluated using GBDT outputs of LB comparisons of LB datasets also from Run-349268 and Run-357409. The set of LB datasets inputted to the LB comparisons used to construct the criteria has been termed the evaluating set. We note that the reference LB datasets in the LB comparisons computed by the DQEWS are not at the beginning of the runs used.

Self-evaluations [1], and cross-evaluations[2] have been performed. The flagging criteria have been constructed as a function of the instantaneous luminosity. The instantaneous luminosity of the LBs contained in each run do not necessarily have the same magnitudes. This results in constructed flagging criteria that do not trivially generalise towards being able to flag all information correctly. In attempting to find a sufficient solution to this, the construction and evaluation of the flagging criteria have been done in the following ways:

1. Both the optimising set and the evaluating set contained precisely the same LB datasets from the same run. As such, the LBNs for each LB dataset in both the optimising and evaluating sets has a one-to-one mapping. From this, it logically extends that the instantaneous luminosities for each LB dataset in both the optimising and evaluating sets have a one-to-one mapping. In this case, an extension of the optimising curve to construct flagging criteria for LB datasets outside of the instantaneous luminosity range of LB datasets in the optimising data does not happen. This is a description of the self-evaluation described in section 7.4.1.

2. The optimising set and the evaluating set each contained different LB datasets from different runs.

[1] Optimising and evaluating sets contain same LB datasets from same run

[2] Optimising and evaluating sets contain different LB datasets from different runs

The implication of this is that it is not necessarily the case that there is a one-to-one mapping of the LBNs of the LB datasets contained in the optimising and evaluating sets. Here, it is necessary to extend the optimising curve either backwards [3] or forwards [4], or both. With these extensions, we effectively extrapolate our information to obtain approximate flags for LB comparisons that contain LB datasets in the evaluating set outside of the instantaneous luminosity range of the optimising sets LB datasets. In the case where the LB datasets used in the evaluation set have instantaneous luminosities that lie within the range of the instantaneous luminosity extrema of LB datasets in the optimising set, there is an interpolation of information, and thus the evaluations are more robust. This is a description of the cross-evaluations described in section 7.4.1.

The DQEWS has been shown to sufficiently flag good LB datasets as good in the following cases.

- Optimising and evaluation sets are from the same LHC-Run and Run (our self-evaluations shown in section 7.4.1), under the condition that all LB datasets contained are of sufficiently expected duration and size.
- Optimising and evaluation LB datasets are from the same LHC-Run (our cross-evaluations shown in section 7.4.1), under the condition that the range of instantaneous luminosity for the LB datasets in evaluating set is similar to, or within, the instantaneous luminosity range of the LB datasets in the optimising set.

Further studies need to be done to determine the precise behaviour of constructed flagging criteria evaluated via interpolation versus extrapolation of optimising set instantaneous luminosity information. The DQEWS fails when attempting to extrapolate outside of the optimising sets instantaneous luminosity range. However, it is noted that there is potential in improving the extrapolation ability should more data be used in an optimising run. In addition to this, more detailed studies on the relationship between the variables contained in the input LB datasets and the instantaneous luminosity may be helpful.

An issue with the DQEWS that has not been accounted for and has not been tested is in potential situations in which bad LB datasets contained in a run are not identifiable as bad based on the GBDT error alone. Useful additions to the DQEWS in future may be along the lines of analyses of fitted continuous distributions by incorporating tests for the drift in the distributions shape parameters in addition to tests of separation.

The explainability of the GBDT models can be significantly improved, allowing for robust and consistent variable attribution of LB comparisons. In doing so, the identification of areas in the detector based on

[3] if the available optimising data begins at a later LBN than the available evaluating data

[4] if the available optimising data begins at an earlier LBN than the available evaluating data

these attributed variables can be made accurate. This would improve the diagnostic ability of the DQEWS as currently defined. A suggested method of improvement in model explainability is the Shapley Additive Explanation [100].

Extending these analyses by testing the ability of the DQEWS to analyse and flag information acquired by all the ATLAS detector sub-systems is a further reaching goal.

The ideal case would be that the DQEWS as currently defined is robust when using LB datasets within similar instantaneous luminosity regions to construct and evaluate the flagging criteria. Robust, in this case, is that an extension of the variable-space of the LB datasets to include information from all detector sub-systems will be sufficient to construct flagging criteria that correctly identify anomalous information with the extended LB datasets.

Should the above suggestion fail, the DQEWS may need to define separate flagging criteria for each detector sub-system and incorporate those results into a consolidated output.

Appendix A

Further detail of datasets used

The tables in this appendix are supplementary to the content in section 7.2.1.

A.1 Variables in LB datasets

These tables show the variables in the LB datasets used in the analyses.

Run-2 DQEWS testing variables	
chiSquared	cov00
cov01	cov02
cov03	cov04
cov11	cov12
cov13	cov14
cov22	cov23
cov24	cov33
cov34	cov44
d0	numberDoF
phi	qOverP
theta	z0
expectInnermostPixelLayerHit	expectNextToInnermostPixelLayerHit
numberOfContribPixelLayers	numberOfDBMHits
numberOfGangedFlaggedFakes	numberOfGangedPixels
numberOfInnermostPixelLayerHits	numberOfInnermostPixelLayerOutliers
numberOfInnermostPixelLayerSharedHits	numberOfInnermostPixelLayerSplitHits
numberOfNextToInnermostPixelLayerHits	numberOfNextToInnermostPixelLayerOutliers
numberOfNextToInnermostPixelLayerSharedHits	numberOfNextToInnermostPixelLayerSplitHits
numberOfOutliersOnTrack	numberOfPhiHoleLayers
numberOfPhiLayers	numberOfPixelDeadSensors
numberOfPixelHits	numberOfPixelHoles
numberOfPixelOutliers	numberOfPixelSharedHits
numberOfPixelSplitHits	numberOfPixelSpoiltHits
numberOfPrecisionHoleLayers	numberOfPrecisionLayers
numberOfSCTDeadSensors	numberOfSCTDoubleHoles
numberOfSCTHits	numberOfSCTHoles
numberOfSCTOutliers	numberOfSCTSharedHits
numberOfSCTSpoiltHits	numberOfTRTDeadStraws
numberOfTRTHighThresholdHits	numberOfTRTHighThresholdHitsTotal
numberOfTRTHighThresholdOutliers	numberOfTRTHits
numberOfTRTHoles	numberOfTRTOutliers
numberOfTRTSharedHits	numberOfTRTTubeHits
numberOfTRTXenonHits	numberOfTriggerEtaHoleLayers
numberOfTriggerEtaLayers	standardDeviationOfChi2OS

Table A.1: Table listing the variables contained in the LB datasets.

A.2 Available and used LBNs

The LBNs that are used in the analyses are highlighted in green (for LBs flagged as good in the GRLs), and red (for the LB flagged as bad in the GRLs) on these tables. The rest that are not used are left in black text. This is determined by the procedure described in section 7.2.1.

Run-349268 LBN	GRL Flag
5	Good
7	Good
11	Good
17	Good
19	Good
57	Good
67	Good
90	Good
110	Good
140	Good
180	Good
209	Good
230	Good
251	Good
252	Bad
253	Good
300	Good
400	Good
512	Good
745	Good
754	Good
760	Good

Table A.2: The LBNs for the LB datasets available for these analyses with their corresponding flag in the GRLs for Run-349268

Run-357409 LBN	GRL Flag
656	Good
657	Good
660	Good
665	Good
671	Good
672	Good
673	Good
682	Good
697	Good
717	Good
750	Good
800	Good
870	Good
970	Good
1100	Good
1236	Good
1237	Good
1238	Good
1240	Good
1245	Good
1270	Good
1300	Good
1344	Good
1400	Good
1472	Good
1473	Good

Table A.3: The LBNs for the LB datasets available for these analyses with their corresponding flag in the GRLs for Run-357409

Appendix B

Results of analyses on hyper-parameter swapping

This appendix contains the results for the flagging criteria (upper and lower curves) constructed using GBDTs trained using swapped hyper-parameters for each of Run-349268 .

B.1 Self-evaluations with swapped hyper-parameters

The effects of swapping the hyper-parameters optimised using the methodology outlined in section 7.3.2 for each run have also been checked. This is using hyper-parameters optimised for GBDTs trained on Run-349268 datasets to train GBDTs using Run-357409 datasets. For the self-consistency checks the following is noted:

1. The bandwidth (distance between lower and upper curves) changes very slightly. There is a slight reduction in the bandwidth. This is because the curve fit parameters and their errors do not change significantly. This implies that there is a small sensitivity on the algorithm as a result of a change in hyper-parameters. More tests needed to determine a more general understanding of the implications of hyper-parameter changes on the DQEWS for a larger set of LHC runs.
2. The flagged LB comparisons do not change. This is shown in figure B.1 and figure B.2 below:
3. The tightness of the LB comparisons differ slightly. Regarding the tightness of the flags for the self-consistency checks we note the following:
    - For the Run-349268 self-consistency check shown in figure B.1: The tightness of all of the LB comparisons increases slightly (all values are numerically closer to one in the swapped hyper-parameter evaluation) when using Run-357409 hyper-parameters to train GBDTs with the Run-349268 data. Results showing this are in table B.1 below.

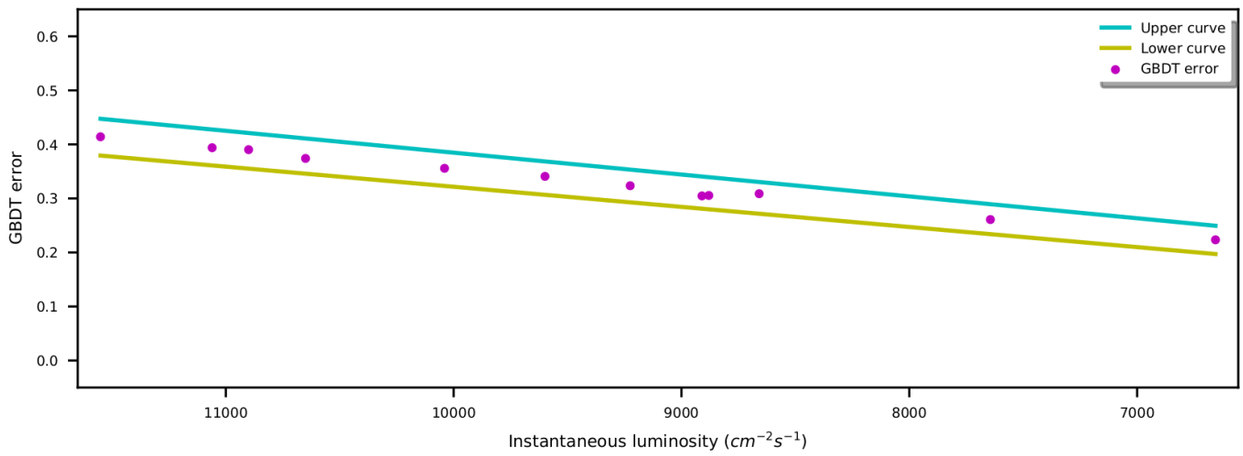


Figure B.1: Plot for self-consistency check using hyper-parameters for Run-357409 to train GBDTs using Run-349268 datasets.

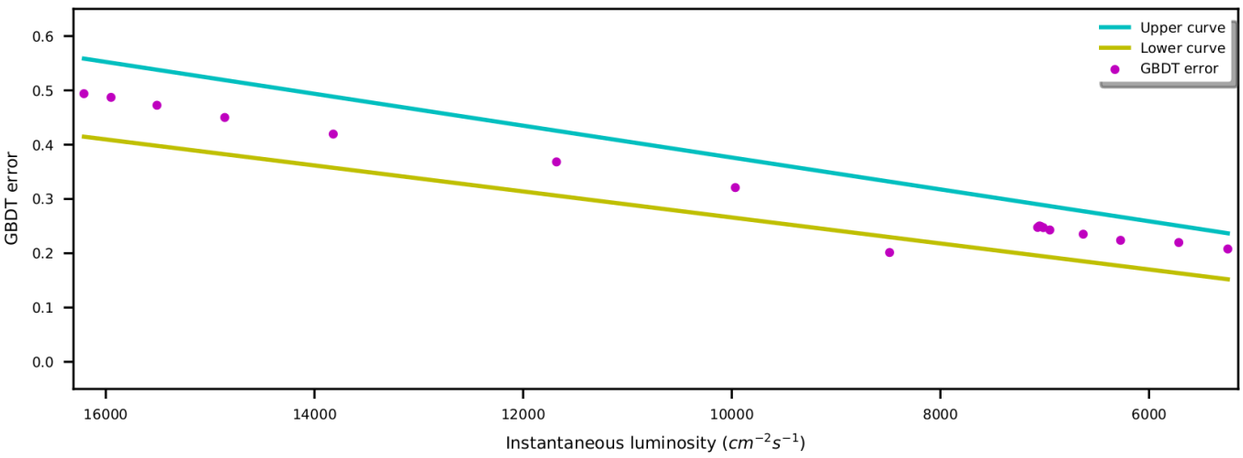


Figure B.2: Plot for self-consistency check using hyper-parameters for Run-349268 to train GBDTs using Run-357409 datasets.

LB comparison	Reference LB: Subject LB	Subject LB duration (s)	GBDT error	InstLumi	Flag	Tightness
1	lb0057 : lb0067	60	0.4141	11550	Good	0.022623769648890846
2	lb0057 : lb0090	60	0.3939	11060	Good	-0.013535870055647466
3	lb0057 : lb0110	60	0.3903	10900	Good	0.06578747915719266
4	lb0057 : lb0140	60	0.374	10650	Good	-0.13319011815252416
5	lb0057 : lb0180	60	0.3557	10040	Good	0.03354430379746835
6	lb0057 : lb0209	60	0.3408	9599	Good	0.10553577209452897
7	lb0057 : lb0230	60	0.3235	9225	Good	0.015846814130075933
8	lb0057 : lb0251	60	0.3047	8910	Good	-0.20449966420416385
9	lb0057 : lb0253	53	0.3055	8880	Good	-0.13855725575920633
10	lb0057 : lb0300	60	0.3088	8659	Good	0.26348936170212767
11	lb0057 : lb0400	60	0.2609	7644	Good	-0.024869345828077132
12	lb0057 : lb0512	60	0.2234	6656	Good	0.00841300191204589

Table B.1: Table for self-consistency check using hyper-parameters for Run-357409 to train GBDTs using Run-349268 datasets.

- For the Run-357409 self-consistency check shown in figure B.2: The tightness of all of the LB comparisons increases (all values are numerically closer to one in the swapped hyper-parameter evaluation) when using Run-349268 hyper-parameters to train GBDTs with the

Run-357409 data. Results showing this are in table B.2 below.

LB comparison	Reference LB: Subject LB	Subject LB duration (s)	GBDT error	InstLumi	Flag	Tightness
1	lb0672 : lb0673	60	0.4939	16210	Good	0.10177975528364851
2	lb0672 : lb0682	60	0.4871	15950	Good	0.10530749789385004
3	lb0672 : lb0697	60	0.4725	15510	Good	0.06639537374170057
4	lb0672 : lb0717	60	0.45	14860	Good	-0.007908032510800323
5	lb0672 : lb0750	60	0.4195	13820	Good	-0.05192425167990226
6	lb0672 : lb0870	60	0.3682	11680	Good	0.040023444695637615
7	lb0672 : lb0970	60	0.3209	9966	Good	0.014338869225882567
8	lb0672 : lb1100	10	0.2014	8487	Bad	-1.553316376442966
9	lb0672 : lb1236	60	0.2476	7068	Good	0.09918578830495928
10	lb0672 : lb1237	60	0.2502	7051	Good	0.16384419983065202
11	lb0672 : lb1238	60	0.2494	7039	Good	0.15420461766574878
12	lb0672 : lb1240	60	0.2472	7014	Good	0.12176495545184557
13	lb0672 : lb1245	60	0.2428	6951	Good	0.06387055567383437
14	lb0672 : lb1270	60	0.2351	6631	Good	0.08262849707221862
15	lb0672 : lb1300	60	0.2238	6273	Good	0.04585225384870971
16	lb0672 : lb1350	60	0.2197	5715	Good	0.2944546287809349
17	lb0672 : lb1400	60	0.2079	5245	Good	0.3188200589970501

Table B.2: Table for self-consistency check using hyper-parameters for Run-349268 to train GBDTs using Run-357409 datasets.

So, while the flagged LB comparisons do not change when swapping hyper-parameters, the tightness of these flagged LB comparisons decreases.

B.2 Cross-evaluations with swapped hyper-parameters

The effects of swapping the hyper-parameters for the optimising and evaluating runs are checked in the same way as described in section 7.4.1. For the cross-evaluation tests we note the following:

1. The flagged LB comparisons change in one of the two cross-evaluations. This is again shown by looking at the GBDT error points that lie between between the upper and lower curves. The results for each of the swapped hyper-parameter cross-evaluations are as follows:

    - The swapped hyper-parameter cross-evaluation in which the optimising datasets are from Run-349268 and the evaluating datasets are from Run-357409 has one LB comparison that is flagged differently to the normal test. This is shown graphically in figure B.3. LB comparison 6 at instantaneous luminosity 11680 $cm^{-2}s^{-1}$ is flagged bad instead of good.

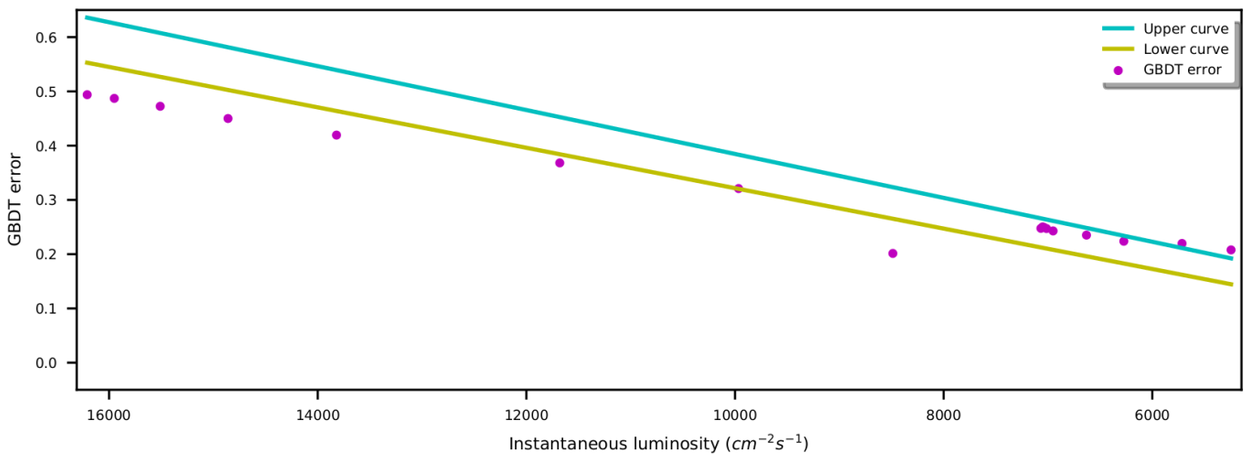


Figure B.3: Plot showing flagging criteria constructed using Run-349268 optimising LB datasets and tested against Run-357409 LB evaluating datasets. Hyper-parameters for each of optimising and evaluating set GBDTs are swapped.

    - The swapped hyper-parameter cross-evaluation in which the optimising datasets are from Run-357409 and the evaluating datasets are from Run-349268 shows no changes. The figure and results showing this are:

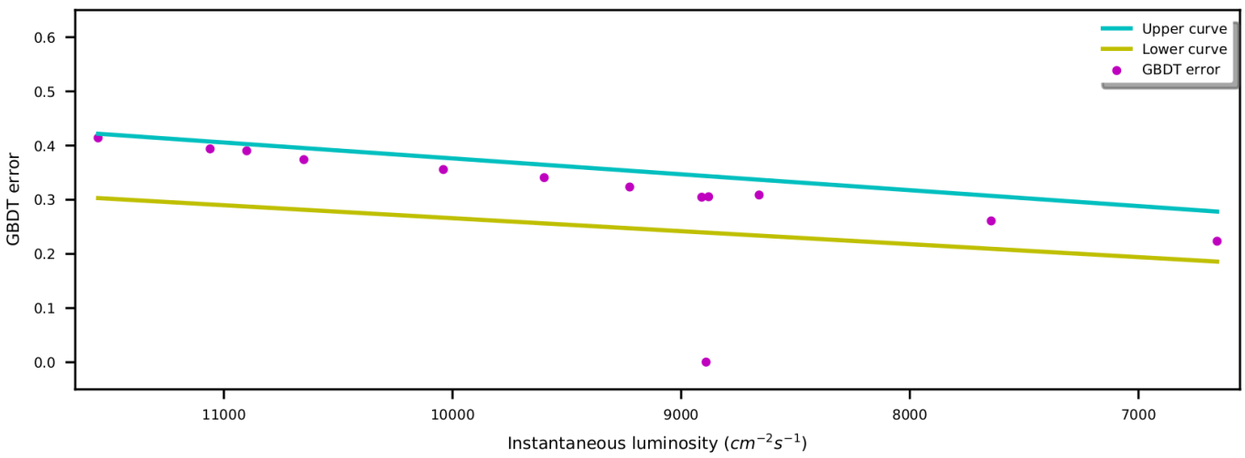


Figure B.4: Plot showing flagging criteria constructed using Run-357409 optimising LB datasets and tested against Run-349268 LB evaluating datasets. Hyper-parameters for each of optimising and evaluating sets GBDTs are swapped.

2. The tightness of the LB comparisons differ. Regarding the tightness of the flags for the cross-evaluations we note the following:

   - For the cross-evaluation in which the optimising datasets are from Run-349268 and the evaluating datasets are from Run-357409: The tightness of the LB comparisons changes very slightly using Run-357409 hyper-parameters to train GBDTs with the Run-349268 data. Numerical values for the tightness which correspond to figure B.3 are shown in table B.3 below:

LB comparison	Reference LB: Subject LB	Subject LB duration (s)	GBDT error	InstLumi	Flag	Tightness
1	lb0672 : lb0673	60	0.4939	16210	Bad	-2.419836302359172
2	lb0672 : lb0682	60	0.4871	15950	Bad	-2.363083657587549
3	lb0672 : lb0697	60	0.4725	15510	Bad	-2.342242019302153
4	lb0672 : lb0717	60	0.45	14860	Bad	-2.33481519115966
5	lb0672 : lb0750	60	0.4195	13820	Bad	-2.177765985672592
6	lb0672 : lb0870	60	0.3682	11680	Bad	-1.4650313914440065
7	lb0672 : lb0970	60	0.3209	9966	Good	-0.9799872935196949
8	lb0672 : lb1100	10	0.2014	8487	Bad	-3.195189003436426
9	lb0672 : lb1236	60	0.2476	7068	Good	0.3136304307290696
10	lb0672 : lb1237	60	0.2502	7051	Good	0.43561030235162373
11	lb0672 : lb1238	60	0.2494	7039	Good	0.4243556219648861
12	lb0672 : lb1240	60	0.2472	7014	Good	0.3786008230452675
13	lb0672 : lb1245	60	0.2428	6951	Good	0.3061032863849765
14	lb0672 : lb1270	60	0.2351	6631	Good	0.4925315970892378
15	lb0672 : lb1300	60	0.2238	6273	Good	0.6066183669473273
16	lb0672 : lb1350	60	0.2197	5715	Bad	1.3432108788309316
17	lb0672 : lb1400	60	0.2079	5245	Bad	1.6532663316582916

Table B.3: Table for flags for Run-357409 optimising LB datasets used to construct the flagging criteria tested against Run-349268 LB evaluating datasets. Hyper-parameters for each of optimising and evaluating set GBDTs are swapped.

   - For the cross-evaluation in which the optimising datasets are from Run-357409 and the evaluating datasets are from Run-349268: The tightness of all of the LB comparisons increases (all values are numerically closer in magnitude to one in the swapped hyper-parameter evalu-

ation) when using Run-349268 hyper-parameters to train GBDTs with the Run-357409 data. The flagged LBs do not change when swapping the hyper-parameters. Numerical values for the tightness which correspond to figure B.4 are shown in table B.4 below:

LB comparison	Reference LB: Subject LB	Subject LB duration (s)	GBDT error	InstLumi	Flag	Tightness
1	lb0057 : lb0067	60	0.4141	11550	Good	0.8715573149161964
2	lb0057 : lb0090	60	0.3939	11060	Good	0.7673356878284091
3	lb0057 : lb0110	60	0.3903	10900	Good	0.7849331713244229
4	lb0057 : lb0140	60	0.374	10650	Good	0.6260976466455919
5	lb0057 : lb0180	60	0.3557	10040	Good	0.6078307261054345
6	lb0057 : lb0209	60	0.3408	9599	Good	0.5622400887163848
7	lb0057 : lb0230	60	0.3235	9225	Good	0.43483992467043314
8	lb0057 : lb0251	60	0.3047	8910	Good	0.24229665071770334
9	lb0057 : lb0252	16	0.0003	8891	Bad	-5.578160919540229
10	lb0057 : lb0253	53	0.3055	8880	Good	0.2735288479969331
11	lb0057 : lb0300	60	0.3088	8659	Good	0.4542898691226369
12	lb0057 : lb0400	60	0.2609	7644	Good	0.05343980343980344
13	lb0057 : lb0512	60	0.2234	6656	Good	-0.18494856524093126

Table B.4: Table for flags for Run-357409 optimising LB datasets used to construct the flagging criteria tested against Run-349268 LB evaluating datasets. Hyper-parameters for each of optimising and evaluating set GBDTs are swapped.

Bibliography

[1] The ATLAS Collaboration. "A combination of measurements of Higgs boson production and decay using up to 139 fb1 of proton–proton collision data at $\sqrt{s}$ = 13 TeV collected with the ATLAS experiment". In: *Physical Review D* 101 (2020), pp. 1–81. DOI: 10.1103/PhysRevD.101.012002.

[2] The ATLAS Collaboration. "The ATLAS experiment at the CERN Large Hadron Collider". In: *Journal of Instrumentation 3* 3 (2008), pp. 13–14. DOI: 10.1140/epjc/s10052-017-5004-5.

[3] Martin zur Nedden. "The LHC Run 2 ATLAS trigger system: design, performance and plans". In: *Journal of Instrumentation* 12 (2017), pp. 1–81. DOI: 10.1088/1748-0221/12/03/C03024.

[4] G. Aad et al. "ATLAS data quality operations and performance for 2015–2018 data-taking". In: *Journal of Instrumentation* 15.04 (Apr. 2020), P04003–P04003. DOI: 10.1088/1748-0221/15/04/p04003. URL: https://doi.org/10.1088/1748-0221/15/04/p04003.

[5] Stefan Mattig. "The Online Luminosity Calculator of ATLAS". In: *Journal of Physics* 331 (2011). DOI: 10.1088/1742-6596/331/2/022035.

[6] James Zhang, Ilija Vukotic, and Robert Gardner. "Anomaly detection in wide area network mesh using two machine learning anomaly detection algorithms". In: *Future Generation Computer Systems* 93 (2018). DOI: 10.1016/j.future.2018.07.023.

[7] M. Asorey. "A concise introduction to quantum field theory". In: *International Journal of Geometric Methods in Modern Physics* 16 (2018), p. 1. DOI: 10.1142/S021988781940005X.

[8] Wikipedia. *Standard Model of Elementary Particles*. URL: https://en.wikipedia.org/wiki/File:Standard_Model_of_Elementary_Particles.svg. (accessed: 01.07.2021).

[9] M. Robinson et al. *A Simple Introduction to Particle Physics*. 2008. DOI: 10.48550/ARXIV.0810.3328. URL: https://arxiv.org/abs/0810.3328.

[10] F Gianotti and S Virdee. "The discovery and measurements of a Higgs boson". In: *Mathematical, physical, and engineering sciences* 373.2032 (2015), pp. 1–25. DOI: http://dx.doi.org/10

.1098/rsta.2014.0384. URL: https://royalsocietypublishing.org/doi/pdf/10.1098/rsta.2014.0384.

[11] Tara Shears. "The Standard Model". In: *Mathematical, physical, and engineering sciences* 373.2032 (2012), pp. 805–817. DOI: https://doi.org/10.1098/rsta.2011.0314. URL: https://royalsocietypublishing.org/doi/epdf/10.1098/rsta.2011.0314.

[12] B. C. Allanchach. "Beyond the Standard Model". In: *CERN Yellow Rep. School Proc.* 6 (2019). Ed. by M. Mulders and C. Duhr, pp. 113–144. DOI: 10.23730/CYRSP-2019-006.113.

[13] Stephen Wolfram. "The Computational Structure of Programs and the Universe". In: *23rd International Symposium on Principles and Practice of Declarative Programming*. PPDP 2021. Tallinn, Estonia: Association for Computing Machinery, 2021. ISBN: 9781450386890. DOI: 10.1145/3479394.3479397. URL: https://doi.org/10.1145/3479394.3479397.

[14] Ben Gripaios. *Lectures on Physics Beyond the Standard Model*. 2015. arXiv: 1503.02636 [hep-ph].

[15] Rolf-Dieter Heuer. "CERN and 60 years of science for peace". In: *AIP Conference Proceedings* 1645.430 (2015), pp. 14–15. DOI: 10.1063/1.4909616.

[16] *Open Access Policy for CERN Publications*. Tech. rep. Geneva: CERN, May 2021. URL: https://cds.cern.ch/record/1955574.

[17] Ewa Lopienska. "The CERN accelerator complex - January 2022. Complexe des accélérateurs du CERN - Janvier 2022". In: (Feb. 2022). General Photo. URL: https://cds.cern.ch/record/2800984.

[18] "LHC Guide". Mar. 2017. URL: https://cds.cern.ch/record/2255762.

[19] Christopher J. Rhodes. "The ATLAS experiment at the CERN Large Hadron Collider". In: *Science Progress* 96 (2013), pp. 95–96. DOI: 10.3184/003685013X13623370524107.

[20] Vasiliki A Mitsou. "Overview of searches for dark matter at the LHC". In: *Journal of Physics: Conference Series* 651 (Nov. 2015), p. 012023. DOI: 10.1088/1742-6596/651/1/012023. URL: https://doi.org/10.1088/1742-6596/651/1/012023.

[21] David Barney. "Presentation for public - Introduction to CMS for CERN guides". In: (2013). URL: https://cds.cern.ch/record/2629323.

[22] Christiane Lefevre. "ATLAS Brochure (English version). Brochure d'ATLAS (version anglaise)". 2011. URL: https://cds.cern.ch/record/1426673.

[23] Eberhard Keil. *The Large Hadron Collider LHC*. CERN, 2013, pp. 95–96.

[24] Eberhard Keil. "The Large Hadron Collider LHC". In: (Oct. 1996). revised version submitted on 2004-08-19 11:10:07, 5 p. URL: https://cds.cern.ch/record/316527.

[25] Werner Herr and B Muratori. "Concept of luminosity". In: (2006). DOI: 10.5170/CERN-2006-002.361. URL: https://cds.cern.ch/record/941318.

[26] The Collaboration et al. "Luminosity determination in pp collisions at TeV using the ATLAS detector at the LHC". In: *European Physical Journal C* 71 (Jan. 2011), pp. 1–37. DOI: 10.1140/epjc/s10052-011-1630-5.

[27] Mike Lamont. "The First Years of LHC Operation for Luminosity Production". In: (2013), MOYAB101. 5 p. URL: https://cds.cern.ch/record/2010134.

[28] *Luminosity determination in pp collisions at* $\sqrt{s} = 13$ *TeV using the ATLAS detector at the LHC*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS CONF-2019-021. Geneva: CERN, 2019. URL: http://cds.cern.ch/record/2677054.

[29] Jorg Wenninger. "Operation and Configuration of the LHC in Run 2". In: (Mar. 2019). URL: https://cds.cern.ch/record/2668326.

[30] Lyndon Evans and Philip Bryant. "LHC Machine". In: *Journal of Instrumentation* 3.08 (Aug. 2008), S08001–S08001. DOI: 10.1088/1748-0221/3/08/s08001. URL: https://doi.org/10.1088/1748-0221/3/08/s08001.

[31] Till Tantau. *The TikZ and PGF Packages. Manual for version 3.0.0*. Dec. 20, 2013. URL: http://sourceforge.net/projects/pgf/.

[32] File:Pseudorapidity plot.svg. *Different values of pseudorapidity shown against a polar grid*. 2012. URL: https://commons.wikimedia.org/wiki/File:Pseudorapidity_plot.svg (visited on 07/24/2012).

[33] The Collaboration et al. "The ATLAS Inner Detector commissioning and calibration". In: *The European Physical Journal C* 70 (Dec. 2010), pp. 787–821. DOI: 10.1140/epjc/s10052-010-1366-7.

[34] Herman Kate. "ATLAS Superconducting Toroids and Solenoid". In: *Applied Superconductivity, IEEE Transactions on* 15 (July 2005), pp. 1267–1270. DOI: 10.1109/TASC.2005.849560.

[35] Georges Aad et al. “Alignment of the ATLAS Inner Detector in Run-2”. In: *Eur. Phys. J. C* 80 (July 2020). 61 pages in total, author list starting page 45, 26 figures, 4 tables, published in EPJC. All figures including auxiliary figures are available at http://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PAPERS/I 2019-05, 1194. 41 p. DOI: 10.1140/epjc/s10052-020-08700-6. arXiv: 2007.07624. URL: https://cds.cern.ch/record/2724037.

[36] Morad Aaboud et al. “Electron reconstruction and identification in the ATLAS experiment using the 2015 and 2016 LHC proton-proton collision data at $\sqrt{s} = 13$ TeV”. In: *Eur. Phys. J. C* 79 (Aug. 2019). 63 pages in total, author list starting page 47, 16 figures, 4 tables, final version published in EPJC. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHY 2017-01, 639. 40 p. DOI: 10.1140/epjc/s10052-019-7140-6. arXiv: 1902.04655. URL: https://cds.cern.ch/record/2657964.

[37] G. Aad et al. “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. URL: http://dx.doi.org/10.1140/epjc/s10052-017-5004-5.

[38] ATLAS Collaboration. “Experiment Briefing: Keeping the ATLAS Inner Detector in perfect alignment”. General Photo. July 2020. URL: http://cds.cern.ch/record/2723878.

[39] Yosuke Takubo. *ATLAS IBL operational experience*. Tech. rep. Geneva: CERN, 2017. DOI: 10.22323/1.287.0004. URL: https://cds.cern.ch/record/2235541.

[40] The ATLAS TRT collaboration et al. “The ATLAS TRT end-cap detectors”. In: *Journal of Instrumentation* 3.10 (Oct. 2008), P10003–P10003. DOI: 10.1088/1748-0221/3/10/p10003. URL: https://doi.org/10.1088/1748-0221/3/10/p10003.

[41] *Performance of the ATLAS Inner Detector Track and Vertex Reconstruction in the High Pile-Up LHC Environment*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/G CONF-2012-042. Geneva: CERN, Mar. 2012. URL: http://cds.cern.ch/record/1435196.

[42] Greg Welch and Gary Bishop. *An Introduction to the Kalman Filter*. Tech. rep. 95-041. Chapel Hill, NC, USA: University of North Carolina at Chapel Hill, 1995. URL: http://www.cs.unc.edu/~welch/kalman/kalmanIntro.html.

[43] Håvard Gjersdal, Are Strandlie, and Ole Røhne. *Straight line track reconstruction for the ATLAS IBL testbeam with the EUDET telescope.* Tech. rep. Geneva: CERN, June 2014. URL: https://cds.cern.ch/record/1708349.

[44] *Performance of primary vertex reconstruction in proton-proton collisions at $\sqrt{s}$ =7 TeV in the ATLAS experiment*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GR CONF-2010-069. Geneva: CERN, July 2010. URL: http://cds.cern.ch/record/1281344.

[45] V Lacuesta. "Track and vertex reconstruction in the ATLAS experiment". In: *Journal of Instrumentation* 8.02 (Feb. 2013), pp. C02035–C02035. DOI: 10.1088/1748-0221/8/02/c02035. URL: https://doi.org/10.1088/1748-0221/8/02/c02035.

[46] M. Aaboud et al. "Jet reconstruction and performance using particle flow with the ATLAS Detector". In: *The European Physical Journal C* 77.7 (July 2017). ISSN: 1434-6052. DOI: 10.1140/epjc/s10052-017-5031-2. URL: http://dx.doi.org/10.1140/epjc/s10052-017-5031-2.

[47] G. Aad et al. "Commissioning of the ATLAS Muon Spectrometer with Cosmic Rays". In: (Jan. 2010).

[48] C. Amelung. "The alignment system of the ATLAS muon spectrometer". In: *European Physical Journal C* 33 (July 2004), s999–s1001. DOI: 10.1140/epjcd/s2004-03-1794-6.

[49] Georges Aad et al. "Jet energy scale and resolution measured in proton–proton collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector". In: *Eur. Phys. J. C* 81.8 (2021), p. 689. DOI: 10.1140/epjc/s10052-021-09402-3. arXiv: 2007.02645 [hep-ex].

[50] Francesca Cavallari. "Performance of calorimeters at the LHC". In: *Journal of Physics: Conference Series* 293 (Apr. 2011), p. 012001. DOI: 10.1088/1742-6596/293/1/012001. URL: https://doi.org/10.1088/1742-6596/293/1/012001.

[51] ATLAS Collaboration and Irinel Caprini. "Muon reconstruction efficiency and momentum resolution of the ATLAS experiment in proton-proton collisions at $\sqrt{s}$=7 TeV in 2010". In: *The European Physical Journal C* 74 (Apr. 2014). DOI: 10.1140/epjc/s10052-014-3034-9.

[52] The ATLAS collaboration. "Resolution of the ATLAS muon spectrometer monitored drift tubes in LHC Run 2". In: *Journal of Instrumentation* 14.09 (Sept. 2019), P09011–P09011. DOI: 10.1088/1748-0221/14/09/p09011. URL: https://doi.org/10.1088%2F1748-0221%2F14%2F09%2Fp09011.

[53] ATLAS Collaboration. *Luminosity determination in pp collisions at $\sqrt{s} = 13$ TeV using the ATLAS detector at the LHC*. 2022. arXiv: 2212.09379 [hep-ex].

[54] Merve Nazlim Agaras. *The ATLAS Tile Calorimeter performance and its upgrade towards the High-Luminosity LHC*. 2021. arXiv: 2105.09099 [physics.ins-det].

[55] D.J. Mahon. "ATLAS LAr calorimeter performance in LHC Run 2". In: *Journal of Instrumentation* 15.06 (June 2020), p. C06045. DOI: 10.1088/1748-0221/15/06/C06045. URL: https://dx.doi.org/10.1088/1748-0221/15/06/C06045.

[56] L. Fabbri. "Forward Detectors in ATLAS: LUCID, ZDC and ALFA". In: *17th International Workshop on Deep-Inelastic Scattering and Related Subjects*. Berlin, Germany: Science Wise Publ., 2009, p. 166.

[57] D Barberis et al. "The ATLAS EventIndex: data flow and inclusion of other metadata". In: *Journal of Physics: Conference Series* 762 (Oct. 2016), p. 012028. DOI: 10.1088/1742-6596/762/1/012028.

[58] ATLAS Collaboration. *The ATLAS Inner Detector Trigger performance in pp collisions at 13 TeV during LHC Run 2*. 2021. arXiv: 2107.02485 [hep-ex].

[59] "The ATLAS Data Acquisition and High Level Trigger system". In: *Journal of Instrumentation* 11.06 (June 2016), P06008–P06008. DOI: 10.1088/1748-0221/11/06/p06008. URL: https://doi.org/10.1088/1748-0221/11/06/p06008.

[60] J. T. Boyd. *LHC Run-2 and Future Prospects*. 2020. arXiv: 2001.04370 [hep-ex].

[61] ATLAS Collaboration. *The ATLAS Inner Detector Trigger performance in pp collisions at 13 TeV during LHC Run 2*. 2021. arXiv: 2107.02485 [hep-ex].

[62] William Panduro Vazquez and ATLAS Collaboration. *The ATLAS Data Acquisition system in LHC Run 2*. Tech. rep. Geneva: CERN, Feb. 2017. DOI: 10.1088/1742-6596/898/3/032017. URL: https://cds.cern.ch/record/2244345.

[63] William Panduro Vazquez and. "The ATLAS Data Acquisition System in LHC Run 2". In: *Journal of Physics: Conference Series* 898 (Oct. 2017), p. 032017. DOI: 10.1088/1742-6596/898/3/032017. URL: https://doi.org/10.1088/1742-6596/898/3/032017.

[64] "The ATLAS Data Acquisition and High Level Trigger system". In: *Journal of Instrumentation* 11.06 (June 2016), P06008–P06008. DOI: 10.1088/1748-0221/11/06/p06008. URL: https://doi.org/10.1088/1748-0221/11/06/p06008.

[65] S Ask et al. “The ATLAS central level-1 trigger logic and TTC system”. In: *Journal of Instrumentation* 3.08 (Aug. 2008), P08002–P08002. DOI: 10.1088/1748-0221/3/08/p08002. URL: https://doi.org/10.1088/1748-0221/3/08/p08002.

[66] Mikhail Mineev, Fedor Prokoshin, and Alexander Iakovlev. “Trigger information data flow for the ATLAS EventIndex”. In: (Sept. 2018). URL: https://cds.cern.ch/record/2637563.

[67] A Corso-Radu et al. “Data Quality Monitoring Framework for the ATLAS experiment: Performance achieved with colliding beams at the LHC”. In: *Journal of Physics: Conference Series* 331.2 (Dec. 2011), p. 022027. DOI: 10.1088/1742-6596/331/2/022027. URL: https://doi.org/10.1088/1742-6596/331/2/022027.

[68] J Adelman et al. “ATLAS offline data quality monitoring”. In: *Journal of Physics: Conference Series* 219.4 (Apr. 2010), p. 042018. DOI: 10.1088/1742-6596/219/4/042018. URL: https://doi.org/10.1088/1742-6596/219/4/042018.

[69] T. Golling et al. “The ATLAS Data Quality Defect Database System”. In: *Eur. Phys. J. C* 72 (Oct. 2011). 6 pages, 3 figures, published in EPJ C. (v2: as published), 1960. 6 p. DOI: 10.1140/epjc/s10052-012-1960-y. arXiv: 1110.6119. URL: https://cds.cern.ch/record/1394173.

[70] T. Golling et al. “The ATLAS Data Quality Defect Database System”. In: *Eur. Phys. J. C* 72 (2012). 6 pages, 3 figures, published in EPJ C. (v2: as published), p. 1960. DOI: 10.1140/epjc/s10052-012-1960-y. arXiv: 1110.6119. URL: https://cds.cern.ch/record/1394173.

[71] C Cuenca Almenar et al. *ATLAS Online Data Quality Monitoring*. Tech. rep. Geneva: CERN, July 2010. DOI: 10.1016/j.nuclphysbps.2011.04.038. URL: https://cds.cern.ch/record/1281356.

[72] C Cuenca Almenar et al. “ATLAS online data quality monitoring”. In: (May 2010). URL: http://cds.cern.ch/record/1267384.

[73] Li-Gang Xia. *Understanding the boosted decision tree methods with the weak-learner approximation*. Warwick University, 2018, p. 2.

[74] Alex Smola and S.V.N. Vishwanathan. “Introduction to Machine Learning”. In: Cambridge University Press, 2008. Chap. 1.

[75] Zoubin Ghahramani. *Unsupervised Learning*. Gatsby Computational Neuroscience Unit, University College London, 2004, pp. 3–4.

[76] Aized Soofi and Arshad Awan. “Classification Techniques in Machine Learning: Applications and Issues”. In: *Journal of Basic Applied Sciences* 13 (Aug. 2017), pp. 459–465. DOI: 10.6000/1927-5129.2017.13.76.

[77] Dmitry Ignatov and Andrey Ignatov. “Decision Stream: Cultivating Deep Decision Trees”. In: *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*. 2017, pp. 905–912. DOI: 10.1109/ICTAI.2017.00140.

[78] Lior Rokach and Oded Maimon. “Decision Trees”. In: vol. 6. Jan. 2005, pp. 165–192. DOI: 10.1007/0-387-25465-X_9.

[79] Marc Romanycia and Francis Pelletier. “What is a heuristic?” In: *Computational Intelligence* 1 (Jan. 1985), pp. 47–58. DOI: 10.1111/j.1467-8640.1985.tb00058.x.

[80] Harris Drucker and Corinna Cortes. “Boosting Decision Trees”. In: *Proceedings of the 8th International Conference on Neural Information Processing Systems*. NIPS’95. Denver, Colorado: MIT Press, 1995, pp. 479–485. DOI: 10.5555/2998828.2998896.

[81] Leo Breiman. *Bias, Variance, and Arcing Classifiers*. Tech. rep. University of California Berkeley, 1996.

[82] Sebastian Ruder. *An overview of gradient descent optimization algorithms*. 2016. DOI: 10.48550/ARXIV.1609.04747. URL: https://arxiv.org/abs/1609.04747.

[83] Tianqi Chen and Carlos Guestrin. “XGBoost: A Scalable Tree Boosting System”. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD ’16. San Francisco, California, USA: Association for Computing Machinery, 2016, pp. 785–794. ISBN: 9781450342322. DOI: 10.1145/2939672.2939785. URL: https://doi.org/10.1145/2939672.2939785.

[84] Enas Elgeldawi et al. “Hyperparameter Tuning for Machine Learning Algorithms Used for Arabic Sentiment Analysis”. In: *Informatics* 8.4 (2021). ISSN: 2227-9709. DOI: 10.3390/informatics8040079. URL: https://www.mdpi.com/2227-9709/8/4/79.

[85] Tötsch N and Hoffmann D. “Classifier uncertainty: evidence, potential impact, and probabilistic treatment”. In: *PeerJ Computer Science* 7.12 (2021). DOI: 10.7717/peerj-cs.398.

[86] Cyril Goutte and Eric Gaussier. “A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation”. In: *Advances in Information Retrieval*. Ed. by David E. Losada

and Juan M. Fernández-Luna. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 345–359. DOI: 10.1007/978-3-540-31865-1_25.

[87] David Powers and Ailab. "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness correlation". In: *J. Mach. Learn. Technol* 2 (Jan. 2011), pp. 2229–3981. DOI: 10.9735/2229-3981.

[88] Lucas Baier et al. "Handling Concept Drifts in Regression Problems - the Error Intersection Approach". In: *CoRR* abs/2004.00438 (2020). arXiv: 2004.00438. URL: https://arxiv.org/abs/2004.00438.

[89] Geoffrey Webb et al. "Analyzing concept drift and shift from sample data". In: *Data Mining and Knowledge Discovery* 32 (Sept. 2018). DOI: 10.1007/s10618-018-0554-1.

[90] T. Hoens, Robi Polikar, and Nitesh Chawla. "Learning from streaming data with concept drift and imbalance: An overview". In: *Progress in Artificial Intelligence* 1 (Apr. 2012). DOI: 10.1007/s13748-011-0008-0.

[91] Jie Lu et al. "Learning under Concept Drift: A Review". In: *IEEE Transactions on Knowledge and Data Engineerings* 31.12 (2018), pp. 1–10. DOI: 10.1109/TKDE.2018.2876857.

[92] Stefan Mattig. "The online luminosity calculator of ATLAS". In: *J. Phys. Conf. Ser.* 331 (2011). Ed. by Simon C. Lin, p. 022035. DOI: 10.1088/1742-6596/331/2/022035.

[93] F. Akesson and E. Moyse. "Event data model in ATLAS". In: *14th International Conference on Computing in High-Energy and Nuclear Physics*. 2005, pp. 255–258.

[94] P F Åkesson et al. *ATLAS Tracking Event Data Model*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-SOFT-PUB-2006-004. Geneva: CERN, 2006. URL: http://cds.cern.ch/record/973401.

[95] A Salzburger. *The ATLAS Track Extrapolation Package*. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-SOFT-PUB-2007-005. Geneva: CERN, 2007. URL: http://cds.cern.ch/record/1038100.

[96] Tianqi Chen and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. San Francisco, California, USA: ACM, 2016, pp. 785–794. ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939785. URL: http://doi.acm.org/10.1145/2939672.2939785.

[97] Carmen Esposito et al. “GHOST: Adjusting the Decision Threshold to Handle Imbalanced Data in Machine Learning”. In: *Journal of Chemical Information and Modeling* 61.6 (2021). PMID: 34100609, pp. 2623–2640. DOI: 10.1021/acs.jcim.1c00160. eprint: https://doi.org/10.1021/acs.jcim.1c00160. URL: https://doi.org/10.1021/acs.jcim.1c00160.

[98] Pauli 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.

[99] Rene Andrae, Tim Schulze-Hartung, and Peter Melchior. *Dos and don’ts of reduced chi-squared*. 2010. DOI: 10.48550/ARXIV.1012.3754. URL: https://arxiv.org/abs/1012.3754.

[100] Scott M Lundberg and Su-In Lee. “A Unified Approach to Interpreting Model Predictions”. In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 4765–4774. URL: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.