



UNIVERSITÀ DEGLI STUDI DI MILANO

FACOLTÀ DI SCIENZE E TECNOLOGIE

CORSO DI LAUREA IN FISICA

TESI DI LAUREA TRIENNALE

A machine learning approach to the electrons and photon classification with the ATLAS detector at the LHC

Autore:

Pietro Daniele

Matricola:

906962

Codice P.A.C.S.:

07.05.-t

Relatore:

Prof. Leonardo Carlo Carminati

Corelatori:

Dott. Ruggero Turra

Dott. Davive Mungo

Anno accademico 2019-2020

Contents

1	LHC and ATLAS	3
1.1	The LHC	3
1.1.1	Lattice Layout	3
1.1.2	The CERN accelerator complex	4
1.1.3	Proton-proton collisions	5
1.2	ATLAS	6
1.2.1	Coordinate system	7
1.2.2	Magnet system	7
1.2.3	Inner detector	8
1.2.4	Calorimetry system	9
1.2.5	Muon spectrometer	10
1.2.6	Trigger system	11
2	Electron and photon reconstruction	13
2.1	The topo-cluster reconstruction	13
2.2	Track reconstruction	14
2.3	Track-cluster matching and photon conversion reconstruction	15
2.4	Supercluster reconstruction	15
2.5	Creation of electrons and photons for analysis	16
2.6	Identification	16
2.6.1	Electron identification	17
2.6.2	Photon identification	19
2.7	Electron and photon isolation	19
3	Machine Learning	22
3.1	Supervised Learning	22
3.1.1	Model and Parameters	22
3.1.2	Objective Function	23
3.2	Gradient boosted trees	23
3.2.1	Decision Tree	23
3.2.2	Decision Tree Ensembles	23
3.2.3	Tree Boosting	24
3.2.4	Learn the tree structure	26
3.3	LightGBM	26

4	BDT: Electron and photon classification	27
4.1	Dataframe and BDT training	27
4.1.1	Dataframe	27
4.1.2	Discriminating features	27
4.2	Parameters and Hyperparameter optimization	28

Chapter 1

LHC and ATLAS

1.1 The LHC

The CERN Large Hadron Collider is a two-ring, superconducting accelerator and collider installed in the long LEP tunnel (27 km)[1] and it provides pp collisions and heavy-ion (e.g. Pb-Pb) collision.

Inside the accelerator, two high-energy particle beams travel at close to the speed of light before they are made to collide. The beams travel in opposite directions in separate beam pipes (two tubes kept at ultrahigh vacuum). They are guided around the accelerator ring by a strong magnetic field maintained by superconducting electromagnets[2],[3].

1.1.1 Lattice Layout

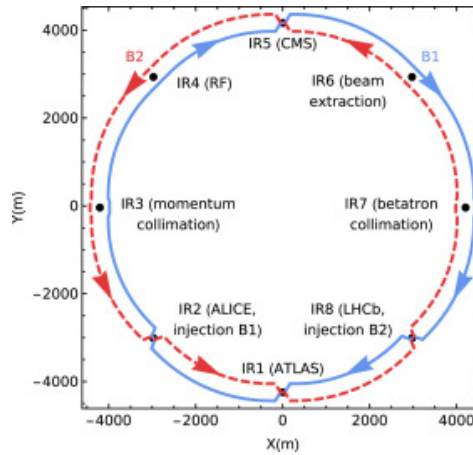


Figure 1.1: LHC structure

The basic layout of the LHC follows the LEP tunnel geometry. The LHC has eight arcs and straight sections. Each straight section is approximately 528 m long and can serve as an experimental or utility insertion. The two high luminosity experimental insertions are located at

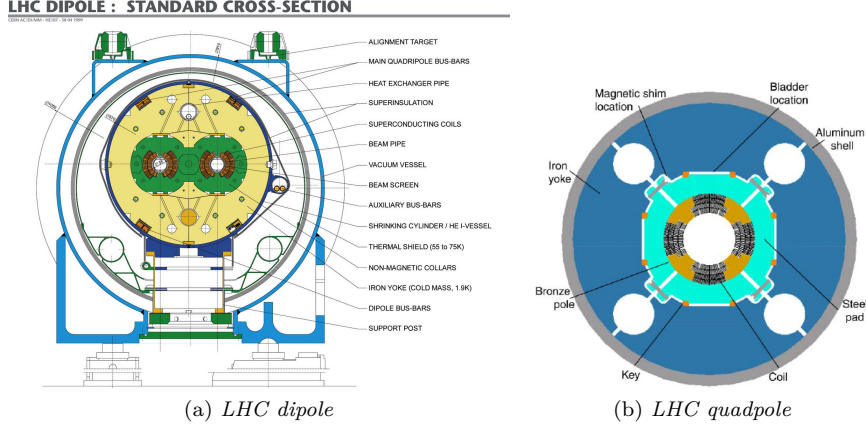


Figure 1.2: LHC's superconducting magnets

diametrically opposite straight sections: the ATLAS experiment is located at point 1 (IR1) and the CMS experiment at point 5 (IR5).

Two more experimental insertions are located at point 2 and point 8 which also contain the injection systems for Beam 1 and Beam 2, respectively. The injection kick occurs in the vertical plane with the two beams arriving at the LHC from below the LHC reference plane. The beams only cross from one magnet bore to the other at these four locations.

The remaining four straight sections do not have beam crossings. Insertion 3 and 7 each contain two collimation systems. Insertion 4 contains two RF systems: one independent system for each LHC beam. The straight section at point 6 contains the beam dump insertion where the two beams are vertically extracted from the machine using a combination of horizontally deflecting fast-pulsed ('kicker') magnets and vertically-deflecting double steel septum magnets. Each beam features an independent abort system. [1]

The protons travel inside along the LHC ring in opposite direction. The LHC beams are controlled by superconducting magnets, which have a working temperature of 1.9 K. There are two kinds of superconducting magnets (Figure 1.2):

- the superconducting dipole magnets, which thanks to a 8.33 T magnetic field drive protons along the ring (circular orbit);
- superconducting quadrupole magnets, which keep the beams focused.

1.1.2 The CERN accelerator complex

[4] Before being insert into LHC ring, particle beam is accelerated by the CERN accelerator complex. It is a succession of machines with increasingly higher energies. Each machine injects the beam into the next one, which takes over to bring the beam to an even higher energy, and so on. In the LHC, the last element of this chain, each particle beam is accelerated up to the record energy of 6.5 TeV. Specifically, in LHC protons are obtained by stripping electrons from hydrogen atoms, which are taken from a bottle containing hydrogen. Then protons are injected into the PS Booster (PSB) at an energy of 50 MeV from Linac2. The booster accelerates them to 1.4 GeV. The beam is then fed to the Proton Synchrotron (PS) where it is accelerated to 25 GeV. Protons are then sent to the Super Proton Synchrotron (SPS) where they are accelerated to 450 GeV. They are finally transferred to the LHC (both in a clockwise and an anticlockwise direction)

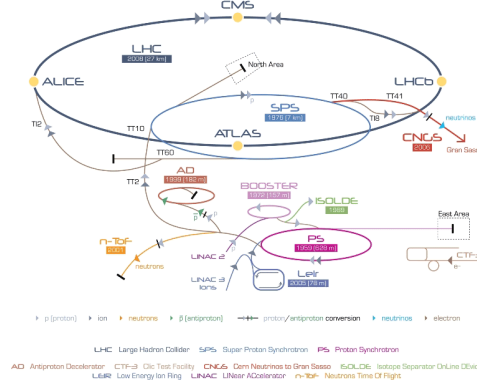


Figure 1.3: LHC and CERN's complex accelerator

where they are accelerated for 20 minutes to 6.5 TeV. Beams circulate for many hours inside the LHC beam pipes under normal operating conditions. In addition to accelerating protons, the accelerator complex can also accelerate lead ions.

1.1.3 Proton-proton collisions

[1]The number of events per second generated in the LHC collisions is given by:

$$N_{event} = L\sigma_{event}$$

where σ_{event} is the cross section for the event under study and L the machine luminosity. L depends only on the beam parameters and can be written for a Gaussian beam distribution as:

$$L = \frac{N_b^2 n_b f_{rev} \gamma_r}{4\pi \epsilon_n \beta^*} F$$

where N_b is the number of particles per bunch, n_b the number of bunches per beam, f_{rev} the revolution frequency, γ_r the relativistic gamma factor, ϵ_n the normalized transverse beam emittance, β^* the beta function at the collision point and F the geometric luminosity reduction factor due to the crossing angle at the IP.

The total inelastic proton-proton cross-section is about 80 mb at $\sqrt{s} = 14$ TeV. [2] Therefore, the event rate R , defined as the number of events produced per second by the pp interactions, is expected to be:

$$R = \sigma L = 80mb \times 10^{34}cm^{-2}s^{-1} \simeq 10^9s^{-1}$$

There are two types of pp collisions:

- **Soft collisions:** they are the most of the collisions and they are large-distance collisions between the two incoming protons. They are called "soft" because the momentum transfer of the interaction is small. Due to this feature, particle scattering at large angle is suppressed and so, after collisions, particles have a large longitudinal momentum, but small transverse momentum ($\langle p_T \rangle \simeq 500MeV$) relative to the beam line. The final states arising from such interactions are called minimum bias events. They are not interesting.
- **Hard collisions:** monochromatic proton beams can be seen as beams of partons (quarks and gluons) with a wide band of energy. Occasionally, head-on collisions occur between two

partons of the incoming protons. These are interactions at small distances, and therefore are characterised by large momentum transfers ("hard scattering"). In this case, particles in the final state can be produced at large angles with respect to the beam line (high p_T) and massive particles can be created. These are the interesting physics events. They are, however, rare compared to the soft interactions.

In the hard-scattering interactions of quarks and gluons at a hadron collider, the effective centre-of-mass energy of the interaction ($\sqrt{\hat{s}}$) is smaller than the centre-of-mass energy of the machine (\sqrt{s}) and is given by:

$$\sqrt{\hat{s}} = \sqrt{x_a x_b s}$$

where x_a and x_b are the fractions of the proton momentum carried by the two colliding partons. If $x_a \simeq x_b$, then the above relation becomes

$$\sqrt{\hat{s}} \simeq x\sqrt{s}$$

Therefore, in order to produce a particle of mass 100 GeV, two quarks (or gluons) which carry only 1% of the proton momentum are needed ($x \sim 0.01$), whereas a particle of mass 5 TeV can only be produced if two partons with $x \sim 0.35$ interact. The momentum distributions of quarks and gluons inside the proton are called parton distribution functions.

1.2 ATLAS

[5]ATLAS is one of two general-purpose detectors at the Large Hadron Collider (LHC). It investigates a wide range of physics, from the search for the Higgs boson to extra dimensions and particles that could make up dark matter. Although it has the same scientific goals as the CMS experiment, it uses different technical solutions and a different magnet-system design.

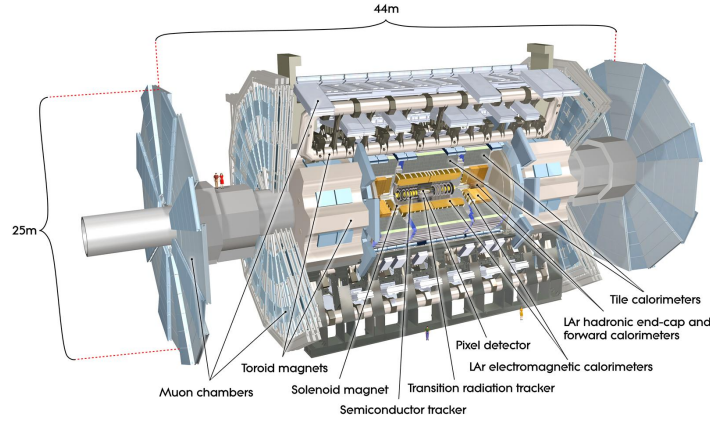


Figure 1.4: ATLAS structure

Beams of particles from the LHC collide at the centre of the ATLAS detector making collision debris in the form of new particles, which fly out from the collision point in all directions. Six different detecting subsystems arranged in layers around the collision point record the paths,

momentum, and energy of the particles, allowing them to be individually identified. A huge magnet system bends the paths of charged particles so that their momenta can be measured.

The interactions in the ATLAS detectors create an enormous flow of data. To digest the data, ATLAS uses an advanced “trigger” system to tell the detector which events to record and which to ignore. Complex data-acquisition and computing systems are then used to analyse the collision events recorded. At 46 m long, 25 m high and 25 m wide, the 7000-tonne ATLAS detector is the largest volume particle detector ever constructed.

1.2.1 Coordinate system

[6]The origin of the coordinate system is set in the nominal point of interaction. The beam direction defines the z-axis and the x-y plane is transverse to the beam direction. X-axis points from the interaction point to the centre of the LHC ring and Y-axis points upwards. The side-A of the detector is defined as that with positive z and side-C is that with negative z.

Polar coordinate are also used: azimuthal angle ϕ is measured as usual around the beam axis, and the polar angle θ is the angle from the beam axis. Using θ , pseudo rapidity is defined as $\eta = -\ln \tan \frac{\theta}{2}$.

The transverse momentum p_T , the transverse energy E_T , and the missing transverse energy E_T^{miss} are defined in the x-y plane unless stated otherwise. The distance ΔR in the pseudo rapidity-azimuthal angle space is defined as $\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2}$.

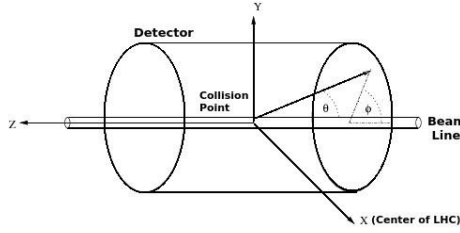


Figure 1.5: ATLAS coordinate system

1.2.2 Magnet system

[6]ATLAS features a unique hybrid system of four large superconducting magnets. This magnetic system is 22 m in diameter and 26 m in length, with a stored energy of 1.6 GJ.

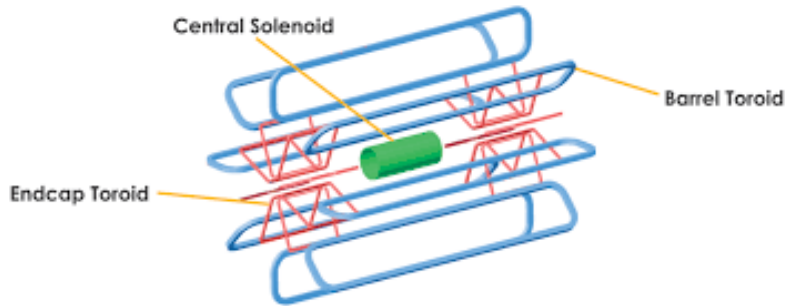


Figure 1.6: ATLAS magnet system

The ATLAS magnet system consists of:

- a solenoid, which is aligned on the beam axis and provides a 2 T axial magnetic field for the inner detector, while minimising the radiative thickness in front of the barrel electromagnetic calorimeter;
- a barrel toroid and two end-cap toroids, which produce a toroidal magnetic field of approximately 0.5 T and 1 T for the muon detectors in the central and end-cap regions, respectively.

1.2.3 Inner detector

[6] The ATLAS Inner Detector (ID) is the inner-most ATLAS layer and it is immersed in a 2 T solenoidal field. It is designed to provide hermetic and robust pattern recognition, excellent momentum resolution and both primary and secondary vertex measurements for charged tracks, with a lower limit in p_t (nominally 0.5 GeV, but as low as 0.1 GeV) and within the pseudorapidity range $|\eta| < 2.5$. It also provides electron identification over $|\eta| < 2.0$ and a wide range of energies (between 0.5 GeV and 150 GeV).

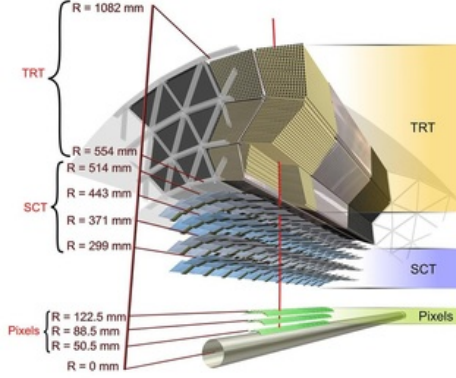


Figure 1.7: ATLAS Inner Detector structure

ID is composed by three independent but complementary sub-detectors:

- **Pixel Detector:** [7] it is the inner-most part of the ATLAS tracking system. It consists of 4 layers of barrel pixel detector and two end caps of three pixel disks each. The innermost pixel layer is a high-resolution pixel detector, called Insertable B-Layer (IBL). The Pixel Detector sits inside the 2T solenoidal magnetic field and contributes to the charged particle tracking of the ATLAS Inner Detector in the pseudo rapidity range of $|\eta| < 2.5$. Due to its high spatial resolution and 3-dimensional space-point measurement the Pixel Detector has a key-role in reconstruction of charged particle tracks. The 4-Layer Pixel Detector will be crucial in the reconstruction of primary and secondary vertices which is essential for the detection of long-lived particles.
- **Semiconductor Tracker (SCT):** [8] it consists of 61 m^2 of active silicon-strip detector modules and it is immersed in a 2 T solenoidal magnetic field. The SCT covers the radial

region from 30 to 52 cm, with hermetic azimuthal coverage out to $|\eta| = 2.5$. Four cylindrical layers in the central region form the “barrel” detector, and nine annular disks on each end of the barrel form the “endcaps”. The layout has been designed so that energetic charged particles will pass through at least four layers everywhere in the acceptance region.

- **Transition Radiation Tracker (TRT):** [9] it is the outmost of the three tracking sub-systems of the ATLAS Inner Detector. TRT is a straw-tube tracker. When a charged particle traverses the TRT, it ionises the gas inside the straws. The resulting free electrons drift towards the wire, where they are amplified and read out.

The spaces between the straws are filled with polymer fibres (barrel) and foils (endcaps) to create transition radiation, which may be emitted by highly relativistic charged particles as they traverse a material boundary. This effect depends on the relativistic factor $\gamma = E/m$ and is strongest for electrons, so it could be a discriminating factor. This design makes the TRT complementary to the silicon-based tracking devices: the intrinsic single-point resolution of $120 \mu\text{m}$ is larger than that of the silicon trackers, but this is compensated by the large number of hits per track (typically more than 30). Furthermore, the high sampling frequency of the wire signals enables the TRT to provide timing information on the nanosecond level.

1.2.4 Calorimetry system

[11] Calorimeters measure the energy a particle loses as it passes through the detector, so the energy of all charged and neutral particles. It is usually designed to stop entire or “absorb” most of the particles coming from a collision, forcing them to deposit all of their energy within the detector. Calorimeters typically consist of layers of “passive” or “absorbing” high-density material interleaved with layers of an “active” medium such as solid lead-glass or liquid argon. Calorimeters can stop most known particles except muons and neutrinos.

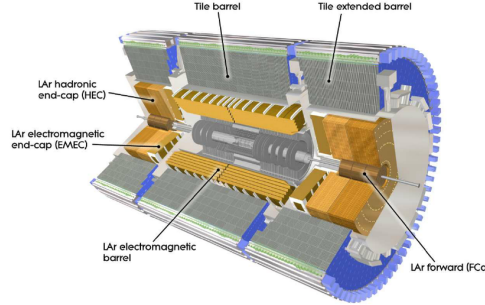


Figure 1.8: ATLAS calorimetry system

The components of the ATLAS calorimetry system are: [10]

- **electromagnetic calorimetry:** the main part of ATLAS EM calorimeter is a lead–liquid argon (LAr) sampling detector with accordion-shaped electrodes and lead absorber plates over its full coverage. The calorimeter is divided into a Barrel part and two End-Caps. Each End-Cap is divided into two coaxial wheels: an outer wheel and an inner wheel covering, respectively, $1.375 < |\eta| < 2.5$ and $2.5 < |\eta| < 3.2$. The absorber lead thickness is constant over large areas. The argon gap thickness is constant in the Barrel but changing with the radius in the End-Cap. In the range $|\eta| < 1.8$, the calorimeter is preceded by a presampler

to recover the energy lost in the upstream material (cryostat, super-conducting coil, inner detector, etc.).

- **hadronic calorimeters:** in the range $|\eta| < 1.6$, the ATLAS hadronic calorimeter is an iron-scintillating tiles calorimeter. For rapidity larger than 1.6, the hadronic calorimeter is an LAr calorimeter. The Hadronic Tile calorimeter is located behind the solenoid coil and the EM calorimeter. It is a sampling calorimeter using iron as absorber material and scintillating tiles as active material. The Hadronic End-Cap calorimeter (HEC) is an LAr sampling calorimeter which provides hadronic coverage for $1.5 < |\eta| < 3.2$.
- **forward calorimeters (FCal):** in the forward region ($3.1 < |\eta| < 4.9$), the EM calorimetry is done by another type of LAr calorimeter. The Forward Calorimeter (FCAL) consists of copper rods parallel to the beam axis inside an outer tube with 250 mm liquid argon gap in between.

1.2.5 Muon spectrometer

[12]The muon spectrometer forms the outer part of the ATLAS detector and occupies by far the largest volume. It was designed to serve two purposes: an independent muon trigger and high quality stand-alone muon reconstruction over a wide range in transverse momentum, pseudo-rapidity ($|\eta| < 2.4$ trigger, $|\eta| < 2.7$ momentum) and azimuthal angle. This is achieved by the use of a large toroidal magnet system together with trigger and high precision tracking chambers.

The magnet system of the muon spectrometer consists of three air-core toroids. Each toroid is build up of eight super conducting coils assembled in a radial configuration. Accurate knowledge of the field is required in order not to degrade the momentum resolution. During ATLAS running a large number of magnetic field sensors will measure the local field ensuring a knowledge of the bending power with a precision better than 0.3%.

Figure 1.9a shows the transverse view of the barrel part of the muon spectrometer. In the barrel a particle typically traverses three measurement planes. The inner most stations are situated directly after the hadronic calorimeter, just outside the toroidal magnetic field. They are equipped with MDT chambers which allow a high precision measurement of the muon trajectory. The second layer of stations is situated in the magnet. The stations in this layer consist of a combination of one MDT chamber and two Resistive Plate Chambers (RPC). A third layer of stations is located just outside the magnetic field. The outer stations are formed by a combination of a MDT chamber and a RPC.

The design of the muon system in the end-caps is different as it is not possible to install stations inside the end-cap magnets and the background rate is much higher. In the end-cap, the first layer of stations sits in front of the magnet. The region closest to the beam pipe, where the background counting rates are highest, is equipped with Cathode Strip Chambers (CSC) instead of MDT chambers because of their higher rate capability. MDT chambers provide the remaining coverage. Thin Gap Chambers (TGC) are installed providing the trigger signal. The second station layer is installed behind the end-cap magnet and is equipped with one layer of MDT chambers and two layers of TGCs. The outer station layer is equipped with MDT chambers.

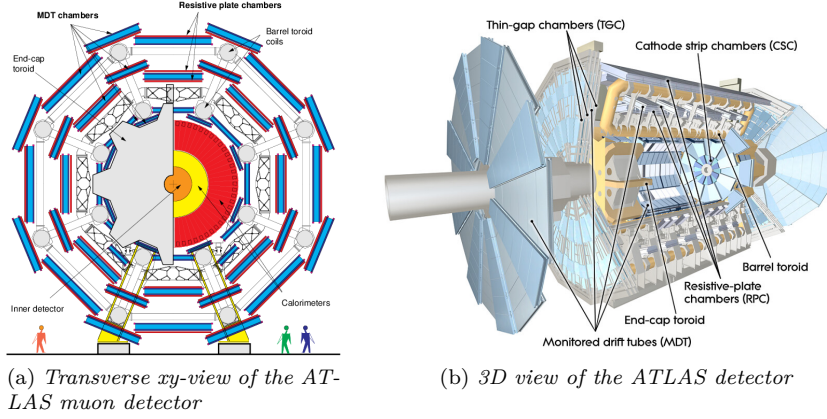


Figure 1.9: ATLAS muon detector

1.2.6 Trigger system

[13]ATLAS is designed to observe up to 1.7 billion proton-proton collisions per second, with a combined data volume of more than 60 million megabytes per second. However, only some of these events will contain interesting characteristics that might lead to new discoveries. To reduce the flow of data to manageable levels, ATLAS uses a specialised two-level online event selection system - the Trigger System - which selects events with distinguishing characteristics that make them interesting for physics analyses.

The two independent levels are, a hardware-based first level (L1) and a software-based high level trigger (HLT) [14].

- The L1 trigger is implemented in fast custom-made electronics and runs with a fixed latency of $2.5\mu\text{s}$. L1 reduces the event rate from the LHC interaction rate of 40 MHz to ≈ 100 kHz. Up to 512 decision items are built, based on Regions of Interest (RoI) in η/φ retrieved from the muon (L1Muon) and calorimeter (L1Calo) systems. The L1 trigger decision is formed by the Central Trigger Processor (CTP). The L1Topo trigger, which is a system introduced for Run-2, performs selections based on geometric or kinematic association between trigger objects received from L1Calo or L1Muon. Another system introduced for Run-2 is the Fast Tracker (FTK), which provides hardware-based tracking information at the rate of the events accepted by L1 for later usage in the HLT.
- In the HLT, offline-like reconstruction algorithms run in a large farm of $\approx 40,000$ processor cores and a decision is formed typically within 300 ms. The HLT is a software trigger providing typically 2500 independent trigger chains. These are sequences of offline-like algorithms executed within the L1 RoIs. Furthermore, full-event reconstruction is possible at the HLT. Events accepted by the HLT are written into different data streams to be used for physics analysis, trigger level analysis, monitoring or detector calibration. Depending on the datastream, the full event or only partial event information is written out, allowing for higher rates without consuming a significant amount of the available bandwidth.

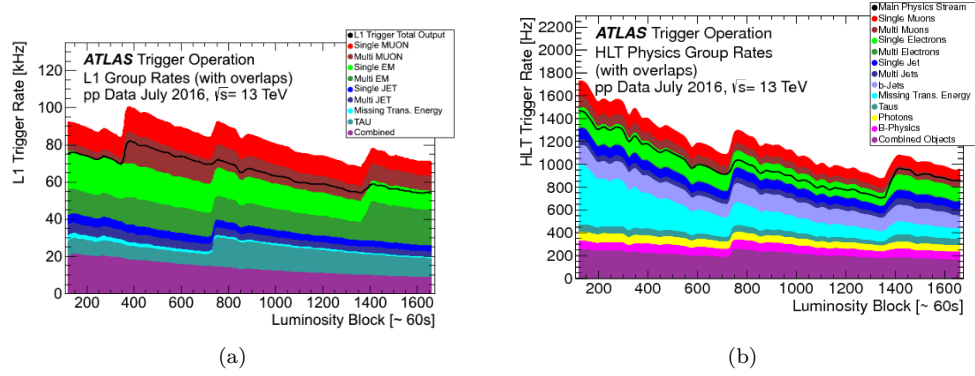


Figure 1.10: L1 (a) and HLT (b) physics trigger rates grouped by trigger signatures as a function of the luminosity block number, in a fill taken in July 2016 with a peak instantaneous luminosity of $1.2 \times 10^{34} \text{cm}^{-2} \text{s}^{-1}$ and an average (peak) pile-up

Chapter 2

Electron and photon reconstruction

Events with electrons and photons in the final state are important signatures for many physics analyses envisaged at the LHC. Their reconstruction mainly exploits data coming from the electromagnetic calorimeter (clusters) and the Inner Detector (ID) systems (tracks) [15][16]:

- an electron is defined as an object consisting of a cluster built from energy deposits in the calorimeter (supercluster) and a matched track (or tracks);
- a converted photon is a cluster matched to a conversion vertex (or vertices), and an unconverted photon is a cluster matched to neither an electron track nor a conversion vertex. About 20% of photons at low $|\eta|$ convert in the ID, and up to about 65% convert at $|\eta| \simeq 2.3$.

The reconstruction with $|\eta| < 2.5$ is based on an algorithm. It first prepares the tracks and clusters it will use. It selects clusters of energy deposits measured in topologically connected EM and hadronic calorimeter cells, denoted topo-clusters. These clusters are matched to ID tracks, which are re-fitted accounting for bremsstrahlung. The algorithm also builds conversion vertices and matches them to the selected topo-clusters. The electron and photon supercluster-building steps then run separately using the matched clusters as input. After applying initial position corrections and energy calibrations to the resulting superclusters, the supercluster-building algorithm matches tracks to the electron superclusters and conversion vertices to the photon superclusters. The electron and photon objects to be used for analyses are then built, their energies are calibrated, and discriminating variables used to separate electrons or photons from background are added.

2.1 The topo-cluster reconstruction

The topo-cluster reconstruction algorithm begins by forming proto-clusters in the EM and hadronic calorimeters using a set of noise thresholds in which the cell initiating the cluster is EM required to have significance $|\zeta_{cell}^{EM}| \geq 4$ where

$$\zeta_{cell}^{EM} = \frac{E_{cell}^{EM}}{\sigma_{cell}^{EM}}$$

E_{cell}^{EM} is the energy cell at the EM scale and σ_{cell}^{EM} is the expected cell noise, which includes the known electronic noise and an estimate of the pile-up noise corresponding to the average instantaneous luminosity expected for Run 2. In order to suppress the formation of noise cluster, in this initial stage, cells from the presampler and the first LAr EM calorimeter layer are excluded from initiating proto-clusters. An important role is played by the neighbouring cells. If they have a significance $|\zeta_{cell}^{EM}| \geq 2$, these cells are collected by proto-cluster. Each neighbour cell passing the threshold of $|\zeta_{cell}^{EM}| \geq 2$ becomes a seed cell in the next iteration, collecting each of its neighbours in the proto-cluster. If two proto-clusters EM contain the same cell with $|\zeta_{cell}^{EM}| \geq 2$ above the noise threshold, these proto-clusters are merged. A crown of nearest-neighbour cells is added to the cluster independently on their energy ($|\zeta_{cell}^{EM}| \geq 0$). This set of thresholds is commonly known as '4-2-0' topo-cluster reconstruction.

Energy becomes an important feature when a cell has $|E_{cell}^{EM}| > 500$ MeV. A cell with this energy, at least four neighbours, and when none of the neighbours has a larger signal, is a local maximum. Proto-clusters with two or more local maxima are split into separate clusters.

2.2 Track reconstruction

Track finding is one of the most challenging tasks in reconstructing events from proton-proton collisions recorded by the ATLAS detector. The process consists in finding a track in the ID which can be matched to the energy clusters. The ID track reconstruction consists of several sequences with different strategies and the main sequence is referred to as inside-out track finding: [19]

- **Space point formation:** the initial step of the ID reconstruction consists of the cluster and drift circle creation and the transformation of clusters in the silicon detectors into 3D space points. Clusters are formed by finding connected cells in the pixel and strip detectors. [18] From these clusters, three-dimensional measurements referred to as space-points are created. In the pixel detector, each cluster equates to one space-point, while in the SCT, clusters from both stereo views of a strip layer must be combined to obtain a three-dimensional measurement.
- **Space point seeded track finding:** track seeds are formed from sets of three space-points in the silicon-detector layers. [19] Seeds can be built from space points in the pixel detector only (referred to as PPP seeds), the strip detector only (SSS) or any mixed setup (PSS,PPS). To reduce the number of potential seeds, initial cuts are applied and dedicated care is taken not to extensively use space points in multiple seeds. Seeds that pass the initial requirements are then input to a track finding algorithm that uses a combinatorial Kalman filter technique and aims to complete the track candidates within the silicon detector.
- **Ambiguity solving:** track candidates are then further processed in an ambiguity solving module that aims to eliminate track candidates from random hit combinations (often referred to as "fakes") or track duplicates, which can be identified by measurements that are shared with other track candidates. The ambiguity solving relies on a scoring function applying positive scores for unique measurements and good fit quality, while penalising missing measurements where they would be expected (also called holes) or shared measurements with other track candidates.
- **TRT extension:** tracks that successfully pass the ambiguity solving stage and are within the coverage of the TRT detector are then extended into the TRT and completed for measurements in the outermost tracking detector. A successful TRT extension increases the momentum resolution significantly by exploiting the longer lever arm for field integration.

2.3 Track-cluster matching and photon conversion reconstruction

2.4 Supercluster reconstruction

The reconstruction of electron and photon superclusters proceeds independently, each in two stages [16]:

1. in the first stage, EM topo-clusters are tested for use as seed cluster candidates, which form the basis of superclusters;
2. in the second stage, EM topo-clusters near the seed candidates are identified as satellite cluster candidates, which may emerge from bremsstrahlung radiation or topo-cluster splitting.

Superclusters are built through various steps:

- the initial list of EM topo-clusters is sorted according to descending E_T , calculated using the EM energy.
- the clusters are tested one by one in the sort order for use as seed clusters. There are two seed's kind:
 - i. electron supercluster seed: a cluster with a minimum E_T of 1 GeV and matched to a track with at least four hits in the silicon tracking detectors.
 - ii. photon supercluster seed: a cluster with E_T greater than 1.5 GeV with no requirement made on any track or conversion vertex matching.

A cluster cannot be used as a seed cluster if it has already been added as a satellite cluster to another seed cluster.

- if a cluster meets the characteristics of the previous point, the algorithm attempts to find satellite clusters, using the process summarized in figure 2.1. For both electrons and photons, a cluster is considered a satellite if it falls within a window of $\Delta\eta \times \Delta\varphi = 0.075 \times 0.125$ around the seed cluster barycentre, as these cases tend to represent secondary EM showers originating from the same initial electron or photon. For electrons, this window could be larger, $\Delta\eta \times \Delta\varphi = 0.125 \times 0.300$, and its 'best-matched' track is also the best-matched track for the seed cluster. For photons with conversion vertices made up only of tracks containing silicon hits, a cluster is added as a satellite if its best-matched (electron) track belongs to the conversion vertex matched to the seed cluster. These steps rely on tracking information to discriminate distant radiative photons or conversion electrons from pile-up noise or other unrelated clusters.

The seed clusters with their associated satellite clusters are called superclusters.

- The final step in the supercluster-building algorithm is to assign calorimeter cells to a given supercluster. Only cells from the presampler and the first three LAr calorimeter layers are considered, except in the transition region of $1.4 < |\eta| < 1.6$, where the energy measured in the scintillator between the calorimeter cryostats is also added. To limit the superclusters' sensitivity to pile-up noise, the size of each constituent topo-cluster is restricted to a maximal width of 0.075 or 0.125 in the η direction in the barrel or endcap region, respectively. Because the magnetic field in the ID is parallel to the beam-line,

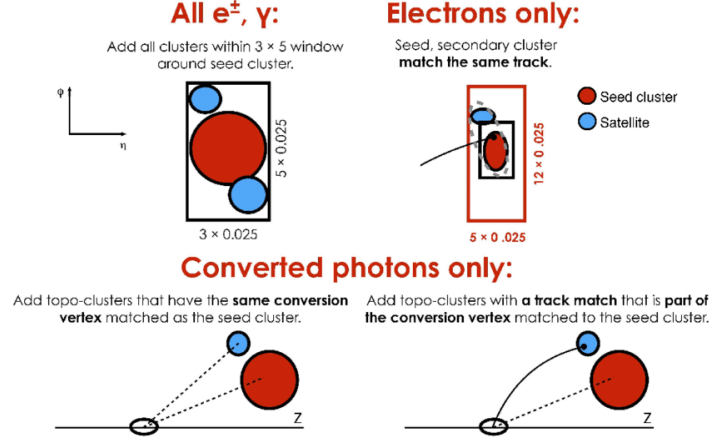


Figure 2.1: Diagram of the superclustering algorithm for electrons and photons. Seed clusters are shown in red, satellite clusters in blue.

interactions between the electron or photon and detector material generally cause the EM shower to spread in the φ direction, so the restriction in η still generally allows the electron or photon energy to be captured. No restriction is applied in the φ -direction.

2.5 Creation of electrons and photons for analysis

[16]After the electron and photon superclusters are built, tracks are matched to electron superclusters and conversion vertices to photon superclusters. Then the analysis-level electrons and photons are created. Because electron and photon superclusters are built independently, a given seed cluster can produce both an electron and a photon. In such cases, the procedure presented in figure 2.2 is applied. The purpose is that if a particular object can be easily identified only as a photon (a cluster with no good track attached) or only as an electron (a cluster with a good track attached and no good photon conversion vertex), then only a photon or an electron object is created for analysis; otherwise, both an electron and a photon object are created. Furthermore, these cases are marked explicitly as ambiguous, allowing the final classification of these objects to be determined based upon the specific requirements of each analysis.

2.6 Identification

[20]Excellent electron and photon identification capabilities are crucial for many aspects of the ATLAS physics program, from standard model measurements (including Higgs boson) to new physics searches. The identification of prompt photons and the rejection of backgrounds, mostly coming from photons from hadron decays, relies on the high granularity of the ATLAS calorimeter. Electron identification is based on a likelihood (LH) discrimination to separate isolated electron candidates from candidates originating from photon conversions, hadron misidentification and heavy flavor decays.

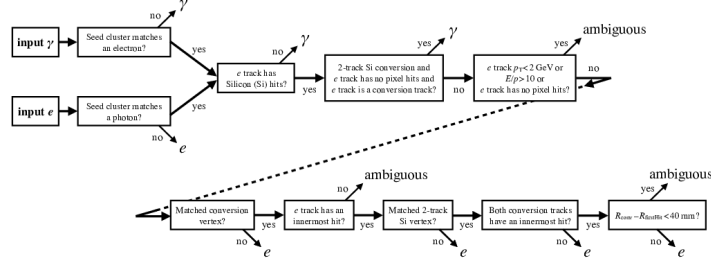


Figure 2.2: Flowchart showing the logic of the ambiguity resolution for particles initially reconstructed both as electrons and photons.

2.6.1 Elettron identification

[16]The quantities used in the electron identification are chosen according to their ability to discriminate prompt isolated electrons from energy deposits from hadronic jets, from converted photons and from genuine electrons produced in the decays of heavy-flavour hadrons. The variables can be grouped into properties of (Table 2.1):

- the primary electron track, which is required to fulfil a set of quality requirements, namely hits in the two inner tracking layers closest to the beam line, as well as a number of hits in the silicon-strip detectors. The transverse impact parameter of the track and its significance are used to construct the likelihood discriminant.
- the lateral development of the electromagnetic shower, which is characterized with variables calculated separately in the first and second layer of the electromagnetic calorimeter. To reject clusters from multiple incident particles, w_{stot} is used. The lateral shower development is measured with R_φ and R_η .
- the longitudinal development of the electromagnetic shower, for them the numbers of cells contributing to the energy measurement in each layer are chosen dynamically in the supercluster approach, compared with fixed numbers of cells in fixed-size clusters. The supercluster approach inherently suppresses noise in the calorimeter cells, resulting in lower values and narrower distributions. The electron identification uses f_1 and f_3 .
- the spatial compatibility of the primary electron track with the reconstructed cluster, which are matched using $\Delta\eta_1$ and $\Delta\phi_{res}$.

Then the combination of information from the tracker and the matching, information from the electromagnetic calorimeter, and hadronic leakage are put together in likelihoods for a reconstructed electron to originate from signal, L_S , or background, L_B . They are calculated from

Type	Description	Simbol
Hadronic leakage	- Ratio of E_T in the first layer of the hadronic calorimeter to E_T of the EM calorimeter (in the pseudorapidity range $ \eta < 0.8$ or $ \eta > 1.37$)	R_{had1}
	- Ratio of E_T in the first layer of the hadronic calorimeter to E_T of the EM calorimeter (in the pseudorapidity range $0.8 < \eta < 1.37$)	R_{had}
Layer 3 of the EM calorimeter	Ratio of the energy in the back layer to the total energy in the EM calorimeter. This variable is only used below 100 GeV because it is known to be inefficient at high energies.	f_3
Layer 2 of the EM calorimeter	- Lateral shower width $\sqrt{(\sum_i E_i \eta_i^2)/(\sum_i E_i) - ((\sum_i E_i \eta_i)/(\sum_i E_i))^2}$,where E_i is the EM calorimeter energy and η_i is the pseudorapidity of cell i and the sum is calculated within a window of 3×5 cells	w_n^2
	- Ratio of the energy in 3×3 cells over the energy in 3×7 cells centered at the electron cluster position	R_ϕ
	- Ratio of the energy in 3×7 cells over the energy in 7×7 cells centered at the electron cluster position	R_η
Layer 1 of the EM calorimeter	- Shower width, $\sqrt{(\sum_i E_i (i - i_{max})^2)/(\sum_i E_i)}$ where i runs over all strips in a window of $\Delta\eta \times \Delta\varphi \sim 0.0625 \times 0.2$, corresponding typically to 20 strips in η , and i_{max} is the index of the highest-energy strip	w_{stot}
	- Ratio of the energy difference between the largest and second largest energy E ratio deposits in the cluster over the sum of these energies	E_{ratio}
	- Ratio of the energy in the strip layer to the total energy in the EM accordion calorimeter	f_1
Track conditions	- Number of hits in the innermost pixel layer; discriminates against photon conversions	n_{Blayer}
	- Number of hits in the Pixel Detector	n_{Pixel}
	- Number of total hits in the pixel and SCT detectors	n_{Si}
	- Transverse impact parameter with respect to the beam-line	d_0
	- Significance of transverse impact parameter defined as the ratio of d_0 and its uncertainty	d_0/σ_{d_0}
	- Momentum lost by the track between the perigee and the last measurement point divided by the original momentum	$\Delta p/p$
TRT	Likelihood probability based on transition radiation in the TRT	eProbabilityHT
Track-cluster matching	- $\Delta\eta$ between the cluster position in Layer 1 and the extrapolated track	$\Delta\eta_1$
	- $\Delta\Phi$ between the cluster position in the Layer 2 and the track extrapolated from the perigee	$\Delta\Phi_2$
	- Defined as $\Delta\Phi_2$, but the track momentum is rescaled to the cluster energy before extrapolating the track from the perigee to the Layer 2 of the calorimeter.	$\Delta\phi_{res}$
	- Ratio of the cluster energy to the track momentum	E/p

Table 2.1: Discriminating variables used for electron identification [20]

probability density functions (pdfs), P :

$$L_{S/B}(\mathbf{x}) = \prod_{i=1}^n P_{S/B,i}(x_i)$$

For signal and background the pdfs take the values $P_{S,i}(x_i)$ and $P_{B,i}(x_i)$, respectively, for the quantity i at value x_i . The likelihood discriminant d_L is defined as the natural logarithm of the ratio of L_S and L_B :

$$d_L = \ln \frac{L_S}{L_B}$$

and each electron is assigned a specific value.

Using the same variables but different values of d_L , three different operating points: Loose, Medium and Tight. They are sorted with increasing threshold values, and are chosen in order to have efficiency for electrons with $E_T > 40$ GeV of 93%, 88%, and 80% respectively. This means that they are inclusive, and one the subset of the other.

2.6.2 Photon identification

[16]The photon identification criteria are designed to efficiently select prompt, isolated photons and reject backgrounds from hadronic jets. The photon identification is constructed from one-dimensional selection criteria, or a cut-based selection, using the shower shape variables (Table 2.2).

The variables using the EM first layer play a particularly important role in rejecting π^0 decays into two highly collimated photons. The primary identification selection is labelled as Tight, with less restrictive selections called Medium and Loose, which are used for the R_{had} , R_{had1} , R_η , and w_{η_2} shower shape variables. The Medium selection adds a loose cut on E_{ratio} . Because the reconstruction of photons in the ATLAS trigger system does not differentiate between converted and unconverted photons, the Loose and Medium identification criteria are the same for converted and unconverted photons. The Tight identification criteria are designed to select a subset of the photon candidates passing the Medium criteria.

Because the shower shapes vary due to the geometry of the calorimeter, the cut-based selection of Loose, Medium and Tight are optimized separately in bins of $|\eta|$. The Tight identification is also optimized in separate bins of E_T . The Tight identification is performed separately for converted and unconverted photons. The shower shapes of converted photons differ from unconverted photons due to the opening angle of the e^+e^- conversion pair, which is amplified by the magnetic field, and from the additional interaction of the conversion pair with the material upstream of the calorimeters.

The tight selections are separately optimised for unconverted and converted photons, to account for the generally broader lateral shower profile of the latter. The thresholds of the selection criteria are different in seven intervals of the reconstructed photon $|\eta|$ to account for the calorimeter geometry, and for different effects on the shower shapes from the material upstream of the calorimeter.

2.7 Electron and photon isolation

The isolation progress, or isolation cut, is applied in order to reduce the backgrounds. Different criteria are defined, with different levels of efficiency in rejecting the background. The isolation working points uses variables from the tracking and from the calorimeter. [21]

Category	Description	Name	<i>loose</i>	<i>tight</i>
Acceptance	$\eta < 2.37$, with $1.37 < \eta < 1.52$ excluded	-	*	*
Hadronic leakage	- Ratio of E_T in the first layer of the hadronic calorimeter to E_T of the EM calorimeter (used over range $ \eta < 0.8$ or $ \eta > 1.37$)	R_{had1}	*	*
	- Ratio of E_T in the first layer of the hadronic calorimeter to E_T of the EM calorimeter (in the pseudorapidity range $0.8 < \eta < 1.37$)	R_{had}	*	*
EM Middle layer	- Ratio of $3 \times 7 \eta \times \phi$ to 7×7 cell energies	R_η	*	*
	- Lateral width of the shower	$w_{\eta 2}$	*	*
	- Ratio of $3 \times 3 \eta \times \phi$ to 7×7 cell energies	R_ϕ		*
EM Strip layer	- Shower width calculated from three strips around the strip with maximum energy deposit	w_{s3}		*
	- Total lateral shower width	w_{stot}		*
	- Energy outside the core of the three central strips but within seven strips divided by energy within the three central strips	F_{side}		*
	- Difference between the energy associated with the second maximum in strip layer(L1) and the energy reconstructed in the strip with the minimum value found between the first and second maxima	ΔE		*
	- Ratio of the energy difference associated with the largest and second largest energy deposits to sum of these energies	E_{ratio}		*

Table 2.2: Discriminating variables used for *loose* and *tight* photon identification [20]

[16]The activity near leptons and photons can be quantified from the tracks of nearby charged particles, or from energy deposits in the calorimeters, leading to two classes of isolation variables:

- The raw calorimeter isolation($E_{T,raw}^{isol}$) is built by summing the transverse energy of positive-energy topological clusters (in EM scale) whose barycentre falls within a cone centred around the electron or photon cluster barycentre. The raw calorimeter isolation includes the EM particle energy ($E_{T,core}$), which is subtracted by removing the energy of the EM calorimeter cells contained in a $\Delta\eta \times \Delta\phi = 5 \times 7$ (in EM-middle-layer units) rectangular cluster around the barycentre of the EM particle cluster. Other correction are done to account for the pile-up.

The fully corrected calorimeter isolation variable is computed as:

$$E_T^{coneXX} = E_{T,raw}^{isolXX} - E_T^{core} - E_{T,leakage}(E_T, \eta, \Delta R) - E_{T,pile-up}(\eta, \Delta R)$$

where XX refers to the size of the employed cone.

- The track isolation variable (p_T^{coneXX}) is computed by summing the transverse momentum of selected tracks within a cone centred around the electron track or the photon cluster direction (excluding tracks matched to the electron or converted photon). [21] In order to compensate for very busy environment at high p_T the cone is of variable size:

$$\Delta R = \min\left(\frac{10}{p_T[\text{GeV}]}, \Delta R_{max}\right)$$

where ΔR_{max} is the maximum cone size (typically 0.2).

The tracks considered are required to have $p_T > 1$ GeV and $|\eta| < 2.5$, at least seven silicon (Pixel + SCT) hits, at most one shared hit, at most two silicon holes and at most one pixel hole. In addition, for electron isolation, the tracks are required to have a loose vertex association, i.e. the track was used in the primary vertex fit, or it was not used in any vertex fit but satisfies $|\Delta z_0| \sin\theta < 3$ mm, where $|\Delta z_0|$ is the longitudinal impact parameter relative to the chosen primary vertex; for photon isolation, all selected tracks satisfying $|\Delta z_0| \sin\theta < 3$ mm are used.

Chapter 3

Machine Learning

Machine Learning (ML) has been a useful tool for the advances of both data analysis and artificial intelligence (AI) [22]. ML is an application of AI with the aim of generating systems with the ability to automatically learn and improve from experience without being explicitly programmed. The ML algorithm is exposed to a training dataset in order to generate a mathematical model, which is then applied in real-life performance tasks.

ML can be divided into three main categories [23]:

1. **Supervised Learning:** the goal is to learn a mapping from inputs \mathbf{x} to outputs y , given a labeled set of input-output pairs $D = \{(x_i, y_i)\}_{i=1}^N$. Here D is called the training set, and N is the number of training examples. In the simplest setting, each training input \mathbf{x}_i is a D -dimensional vector of numbers, which are called features, attributes or covariates. In general, however, \mathbf{x}_i could be a complex structured object.

The form of the output, or response variable, can in principle be anything, but most methods assume that y_i is a categorical or nominal variable from some finite set, $y_i \in \{1, \dots, C\}$ or that y_i is a real-valued scalar. When y_i is categorical, the problem is known as classification or pattern recognition, and when y_i is real-valued, the problem is known as regression.

2. **Unsupervised learning:** here the algorithm is only given inputs, $D = \{(x_i)\}_{i=1}^N$, and the goal is to find “interesting patterns” in the data. This is sometimes called knowledge discovery.
3. **Reinforcement learning:** this is useful for learning how to act or behave when given occasional reward or punishment signals. A reinforcement learning algorithm, or agent, learns by interacting with its environment. The agent receives rewards by performing correctly and penalties for performing incorrectly. The agent learns without intervention from a human by maximizing its reward and minimizing its penalty.

3.1 Supervised Learning

Let's focus on the supervised learning. [24]

3.1.1 Model and Parameters

As written above, the model in supervised learning usually refers to the mathematical structure of by which the prediction y_i is made from the input \mathbf{x}_i .

The parameters are the undetermined part that we need to learn from data. In linear regression problems, the parameters are the coefficients θ . Usually θ is used to denote the parameters (there are many parameters in a model).

3.1.2 Objective Function

In order to train the model, we need to define the a function of the parameters to measure how well the model fit the training data. This function is called: *Objective function*.

A characteristic of objective functions is that they are composed of two parts (also dependent on parameters): training loss L and regularization term Ω :

$$obj(\theta) = L(\theta) + \Omega(\theta)$$

The training loss measures how predictive our model is with respect to the training data. Common choices of L are the mean squared error, which is given by:

$$L(\theta) = \sum_i (y_i - y_{i_{pred}})^2$$

or logistic loss, to be used for logistic regression:

$$L(\theta) = \sum_i [y_i \ln(1 + e^{-y_{i_{pred}}}) + (1 - y_i) \ln(1 + e^{-y_{i_{pred}}})]$$

The regularization term controls the complexity of the model, which helps to avoid overfitting. The general principle is a simple and predictive model. The tradeoff between the two is also referred as bias-variance tradeoff in machine learning.

3.2 Gradient boosted trees

There are many models, which ML algorithms can use, and one of them is a Decision Tree ensemble. If it is trained, it becomes a Gradient Boosted Tree. The basic unit is decision tree.

3.2.1 Decision Tree

Decision Trees (DTs) are a supervised learning method used for classification and regression. [25]A decision tree is a hierarchical structure consisting of nodes and directed edges. The tree has three types of nodes:

- a root node that has no incoming edges and zero or more outgoing edges.
- internal nodes, each of which has exactly one incoming edge and two or more outgoing edges.
- leaf or terminal nodes, each of which has exactly one incoming edge and no outgoing edges.

3.2.2 Decision Tree Ensembles

[24]The tree ensemble model consists of a set of classification and regression trees (CART). A CART is a bit different from decision trees, in which the leaf only contains decision values. In CART, a real score is associated with each of the leaves. This also allows for a principled, unified approach to optimization.

Usually, a single tree is not strong enough to be used in practice. So the ensemble model is used. In this model the prediction scores of each individual tree are summed up to get the final score:

$$y_{i_{pred}} = \sum_{k=1}^K f_k(x_i), f_k \in \mathcal{F} \quad (3.1)$$

where K is the number of trees, f is a function in the functional space \mathcal{F} , and \mathcal{F} is the set of all possible CARTs. The objective function to be optimized is given by:

$$obj(\theta) = \sum_i^n l(y_i, y_{i_{pred}}) + \sum_{k=1}^K \Omega(f_k) \quad (3.2)$$

so the functions f_k are the parameters.

3.2.3 Tree Boosting

After introducing the model, let's focus on trees' training. An essential element of tree training, as is always for all supervised learning models, is an objective function, which will be optimize:

$$obj = \sum_i^n l(y_i, y_{i_{pred}}^{(t)}) + \sum_{i=1}^t \Omega(f_i) \quad (3.3)$$

There are many training methods and one of them is *Additive Training*. In this case, as mentioned above, the parameters of trees are the function f_i , each containing the structure of the tree and the leaf scores. Due to the difficulty of training many trees at the same time, it is used an additive approach:

$$\begin{aligned} y_{i_{pred}}^{(0)} &= 0 \\ y_{i_{pred}}^{(1)} &= f_1(\mathbf{x}_i) = y_{i_{pred}}^{(0)} + f_1(\mathbf{x}_i) \\ y_{i_{pred}}^{(2)} &= f_1(\mathbf{x}_i) + f_2(\mathbf{x}_i) = y_{i_{pred}}^{(1)} + f_2(\mathbf{x}_i) \\ &\dots \\ y_{i_{pred}}^{(t)} &= \sum_{k=1}^t f_k(\mathbf{x}_i) = y_{i_{pred}}^{(t-1)} + f_t(\mathbf{x}_i) \end{aligned} \quad (3.4)$$

In each step, the tree used is the one that optimizes objective function, which (3.3) in additive approach becomes:

$$\begin{aligned} obj^{(t)} &= \sum_i^n l(y_i, y_{i_{pred}}^{(t)}) + \sum_{i=1}^t \Omega(f_i) \\ &= \sum_i^n l(y_i, y_{i_{pred}}^{(t-1)} + f_t(\mathbf{x}_i)) + \Omega(f_t) + constant \end{aligned} \quad (3.5)$$

Then using a taylor expansion of the loss function up to the second order in (3.5):

$$obj^{(t)} = \sum_i^n l(y_i, y_{i_{pred}}^{(t-1)} + g_i f_t(\mathbf{x}_i)) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i) + \Omega(f_t) + constant \quad (3.6)$$

where g_i and h_i are:

$$\begin{aligned} g_i &= \partial_{y_{i_{pred}}^{(t-1)}} l(y_i, y_{i_{pred}}^{(t-1)}) \\ h_i &= \partial_{y_{i_{pred}}^{(t-1)}}^2 l(y_i, y_{i_{pred}}^{(t-1)}) \end{aligned} \quad (3.7)$$

So the specific objective at step t , without all the constants, becomes:

$$\sum_i^n [g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i)] + \Omega(f_t) \quad (3.8)$$

The other part of objective is the regulation term, which represents the complexity of the tree f_t . f_t can be defined as:

$$f_t(\mathbf{x}) = w_{q(\mathbf{x})}, w \in R^T, q : R^d \rightarrow \{1, 2, \dots, T\} \quad (3.9)$$

where w is the vector of scores in leaves, q is a function assigning each data point to the corresponding leaf, and T is the number of leaves. So the complexity Ω becomes:

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 \quad (3.10)$$

So, using eq. (3.6), (3.9), (3.10), the objective value with the t -th tree is:

$$\begin{aligned} obj^{(t)} &\simeq \sum_{i=1}^n [g_i w_{q(\mathbf{x}_i)} + \frac{1}{2} h_i w_{q(\mathbf{x}_i)}^2] + \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 \\ &= \sum_{j=1}^T [(\sum_{i \in I_j} g_i) w_j + \frac{1}{2} (\sum_{i \in I_j} h_i + \lambda) w_j^2] + \gamma T \end{aligned} \quad (3.11)$$

where $I_j = \{i | q(\mathbf{x}_i) = j\}$ is the set of indices of data points assigned to the j -th leaf. The equation (3.11) can be simplified by defining $G_j = \sum_{i \in I_j} g_i$ and $H_j = \sum_{i \in I_j} h_i$:

$$obj^{(t)} = \sum_{j=1}^T [G_j w_j + \frac{1}{2} (H_j + \lambda) w_j^2] + \gamma T \quad (3.12)$$

Due to w_j are independent with respect to each other, the form $G_j w_j + \frac{1}{2} (H_j + \lambda) w_j^2$ is quadratic, the best w_j for a given structure $q(x)$ is:

$$w_j^{best} = -\frac{G_j}{H_j + \lambda} \quad (3.13)$$

Then using the best w_j , the objective reduction becomes the best objective reduction, which measures how good a tree structure $q(x)$ is:

$$obj^{(t)_{best}} = -\frac{1}{2} \sum_{j=1}^T \frac{G_j}{H_j + \lambda} + \gamma T \quad (3.14)$$

3.2.4 Learn the tree structure

Now thanks to the equation (3.14) it is possible to measure how good a tree is, ideally it is possible to enumerate all possible trees and pick the best one. In practice this is intractable (there are infinite possible trees), so it will be optimized one level of the tree at a time, the *additive approach*: an additional split is added to the tree if it brings a gain to its object function.

When trying to split a leaf into two leaves, and the score it gains is:

$$Gain = \frac{1}{2} \left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma \quad (3.15)$$

This formula can be decomposed as:

- the score on the new left leaf
- the score on the new right leaf
- the score on the original leaf
- regularization on the additional leaf

All possible splits are explored and the best one is added. If the Gain is negative this means that the split have the increase in the complexity score bigger than the reduction in training loss. It would be better not to add that branch. This is exactly the pruning techniques, which reduces the complexity by removing sections of the tree.

3.3 LightGBM

[26]LightGBM is a gradient boosting framework, released by Microsoft, that uses tree based learning algorithms. It is based on GDBT. LightGBM grows trees leaf-wise (best-first) [31]. It will choose the leaf with max delta loss to grow. Holding "leaf" fixed, leaf-wise algorithms tend to achieve lower loss than level-wise algorithms.

Leaf-wise may cause over-fitting when "data" is small, so LightGBM includes the *max_depth* parameter to limit tree depth. However, trees still grow leaf-wise even when *max_depth* is specified. So LGBM has the possibility to control the model and the training through specific parameters.

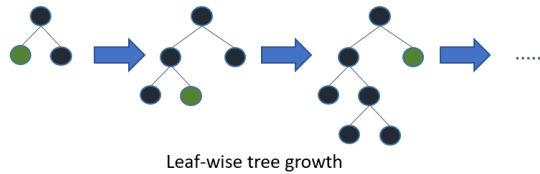


Figure 3.1: LGBM: Leaf-wise tree growth

Chapter 4

BDT: Electron and photon classification

The concepts illustrated in the previous chapters are now applied to a specific case: a dataframe of MC simulated single particles, which will be analysed in more detail in the next section.

The reconstructed particles can be of two types:

- **single reconstruction:** when a particle is only reconstructed as electron or as photon;
- **double reconstruction:** when a particle cannot be classified in a single mode (el or ph) is reconstructed in both modes and is called *ambiguous*. The final classification is

In this second case the ML applies. A supervised machine learning (Gradient BDT) is used to create and train a model. Then this model is applied to the double reconstruction to classify electrons and photons and increase the reconstruction efficiency.

4.1 Dataframe and BDT training

4.1.1 Dataframe

4.1.2 Discriminating features

In order to create a good model, the choice of the features it will have to work on is crucial. The most useful characteristics are those that present the greatest differences between photons and electrons, which are therefore more discriminating. It is also important to use features that guarantee the generality of the model. The features can be divided into two categories: electron and photon features.

Electron features

Electron features are those that characterize the particles reconstructed as electrons. Electrons have many hits in the first layers of detectors, as opposed to photons. This is because the most part of photons is still unconverted. So electron reconstructed as electrons (`tur_el/reco_el`) should have more hits than photons reconstructed as electrons (`true_ph/reco_el`). This can be seen in the characteristics `'el_track_hasInnermostHits'` (a boolean, which says if the particle has hit the first expected layer or not), `'el_trkPixelHits'` (a pixel hit counter), `'el_trkSiHits'` (a silicon hit counter) shown in the figure 4.1.

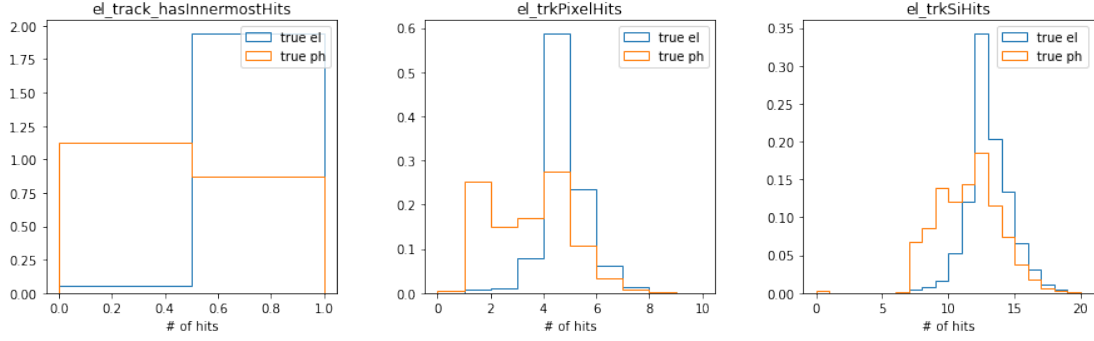


Figure 4.1

The track quality, as it is determined from the pixel and SCT hits, is different in the two cases therefore the physics quantities reconstructed from the track. These quantities (plots in Figure 4.2) are:

- 'el_track_ep',
- 'el_trackz0',
- 'el_cl_pt',
- 'el_cl_eta'

Photon features

As electron, the photon quality track (of the pair e^+e^-) is discriminating and therefore also the Pixel/SCT hits and p_t .

Other useful photon features for classification are those related to conversion:

- '**ph_zconv**' and '**ph_Rconv**': these characteristics contain information of the conversion point in the coordinates R and z. True_el/reco_ph have a conversion point close to the point of interaction as their track starts from the beginning.
- '**ph_pt1conv**', '**ph_pt2conv**' and '**ph_ptconv**'. All these three features have been scaled with 'ph_cl_pt' to eliminate all kinematics information
- '**pt1conv/ptconv**' contains information on conversion symmetry. This characteristic for true_el/reco_ph tends to 1, since most of them have a single track.

4.2 Parameters and Hyperparameter optimization

In order to create a good predictive model, the choice of parameters is crucial, therefore the hyperparameter optimization is used. Hyperparameters are LGMB adjustable parameters chosen for the training of a model, which regulate the training process.

The objective of the hyperparameter optimization is to search the various configurations of the hyperparameters, contained in the hyperparameter space, to find a configuration with the best performance. The hyperparameter space consists of two types of hyperparameters: discrete

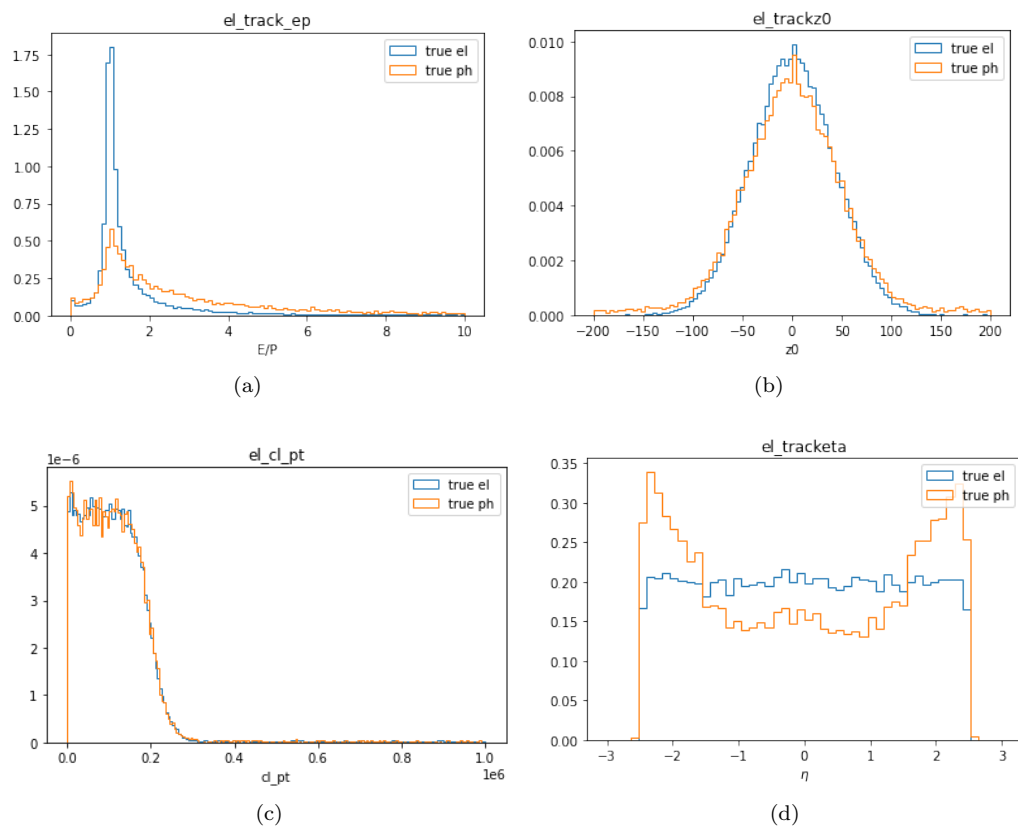


Figure 4.2: Electron physics quantities reconstructed from the track

Parameter	Value
Metric	xentropy
Objective	xentropy
Bagging seed	42
Feature fraction seed	42
Is unbalance	True
Number of leaves	70
Feature fraction	0.8279495901643081
Bagging	0.8039490145162762
Learning rate	0.06057403872801198

Table 4.1: Parameters used in the model's training

or continuous. Let's focus on the continuous ones. Continuous hyperparameters are specified as a uniform distribution over a continuous range.

So after defining the space of the hyperparameters, we move on to the search for the best configuration. This search can be of various types:

- **random search:** in random search the hyperparameter values are randomly selected from the defined search space. It allows the hyperparameter space to include both discrete and continuous hyperparameters;
- **grind search:** grid sampling performs a simple grid search on all possible values in the defined search space;
- **bayesian search:** bayesian sampling uses the knowledge of the previous samplings when choosing hyperparameter values, effectively attempting to improve the reported primary metric. The sample is selected based on the performance of the previous sample, so that the new sample improves the primary metric reported.

The bayesian search is used by the Python library *Hyperopt*, which is implemented in the code in order to perform the optimization.

The training parameters are listed in the Table 4.1 and the ones that have been optimized are:

- **bagging:** in bagging, several models of the same type are trained on different datasets (aggregation, typical of ensemble learning), each obtained from the initial dataset by random sampling with replacement (bootstrap).
- **number of leaves:** it is the maximum number of leaves, that a tree could have. It is useful to avoid overfitting.
- **feature fraction:** it is used when boosting is random forest. 0.82 feature fraction means LightGBM will use 82% of parameters randomly in each iteration for building trees.
- **learning rate:** it determines the effect of each tree on the final outcome.

Durante hyperparam sono state utilizzate 200 ricerche

Bibliography

- [1] European organization for nuclear research, *LHC design report*, CERN libraries, Geneva (2004). <http://cds.cern.ch/record/782076/files/>
- [2] F. Gianotti, *Collider physics: LHC*, EP Division, CERN, Geneva, Switzerland. <https://cds.cern.ch/record/458489/files/p219.pdf>
- [3] *The Large Hadron Collider (LHC)* <https://home.cern/science/accelerators/large-hadron-collider>
- [4] <https://cds.cern.ch/record/2255762/files/CERN-Brochure-2017-002-Eng.pdf>
- [5] <https://home.cern/science/experiments/atlas>
- [6] The ATLAS Collaboration et al 2008 JINST3 S08003 <https://iopscience.iop.org/article/10.1088/1748-0221/3/08/S08003/pdf>
- [7] H. Pernegger, *The Pixel Detector of the ATLAS Experiment for LHC Run-2*, ATL-INDET-PROC-2015-001. <https://cds.cern.ch/record/1985432/files/ATL-INDET-PROC-2015-001.pdf>
- [8] J.R. Pater *The ATLAS SemiConductor Tracker operation and performance*, 2012 JINST7 C04001. <https://iopscience.iop.org/article/10.1088/1748-0221/7/04/C04001/pdf>
- [9] Adrian Vogel *ATLAS Transition Radiation Tracker (TRT): Straw Tube Gaseous Detectors at High Rates*, CERN, Geneva, ATL-INDET-PROC-2013-005, Apr 2013, ATL-INDET-PROC-2013-005. <https://cds.cern.ch/record/1537991>
- [10] http://ific.uv.es/~cabrera/teaching/atlas_i.pdf
- [11] <https://atlas.cern/discover/detector/calorimeter>
- [12] *The ATLAS muon spectrometer: calibration and pattern recognition*, N. Van Eldik, CERN-THESIS-2007-045, 2007, <http://cds.cern.ch/record/1044839>
- [13] *Trigger and Data Acquisition System* <https://atlas.cern/discover/detector/trigger-daq>
- [14] *The Run-2 ATLAS Trigger System: Design, Performance and Plan*, Martin zur Nedden, CERN, ATLAS Collaboration, Geneva, Dec 2016, ATL-DAQ-PROC-2016-039, <https://cds.cern.ch/record/2238679>

Chapter 2

- [15] Wu Xin, Clark Allan and Campanelli Mario *Electron and photon identification in ATLAS*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006 https://link.springer.com/chapter/10.1007/978-3-540-32841-4_21
- [16] *Electron and photon performance measurements with the ATLAS detector using the 2015-2017 LHC proton-proton collision data* <https://iopscience.iop.org/article/10.1088/1748-0221/14/12/P12006>
- [17] *A neural network clustering algorithm for the ATLAS silicon pixel detector*, The ATLAS collaboration <https://doi.org/10.1088/1748-0221/14/12/P12006>
- [18] <https://link.springer.com/content/pdf/10.1140/epjc/s10052-019-7140-6.pdf>
- [19] *Optimisation of the ATLAS Track Reconstruction Software for Run-2*, Andreas Salzburger, CERN, Switzerland, <http://iopscience.iop.org/1742-6596/664/7/072042>
- [20] *Electron and photon identification with the ATLAS detector*, Proklova Nadezda, ATLAS Collaboration, Aug 2018, ATL-PHYS-SLIDE-2018-606, <http://cds.cern.ch/record/2634679>
- [21] *SEARCH FOR SUPERSYMMETRY WITH A COMPRESSED MASS SPECTRUM USING THE ATLAS DETECTOR*, Rossini Lorenzo, <http://hdl.handle.net/2434/683343>

Chapter 3

- [22] *Lifelong Machine Learning*, Zhiyuan Chen and Bing Liu, Morgan & Claypool Publishers, Nov 2016, <https://www.cs.uic.edu/~liub/lifelong-machine-learning-draft.pdf>
- [23] *Machine Learning: A Probabilistic Perspective*, Kevin P. Murphy, The MIT Press, [https://scholar.google.it/scholar?q=K.P.+Murphy,+Machine+Learning.+A+Probabilistic+Perspective+,+MITpress+\(2012\)&hl=it&as_sdt=0&as_vis=1&oi=scholart](https://scholar.google.it/scholar?q=K.P.+Murphy,+Machine+Learning.+A+Probabilistic+Perspective+,+MITpress+(2012)&hl=it&as_sdt=0&as_vis=1&oi=scholart)
- [24] <https://xgboost.readthedocs.io/en/latest/tutorials/model.html>
- [25] *4Classification:Basic Concepts,Decision Trees, andModel Evaluation*, <https://www-users.cs.umn.edu/~kumar001/dmbook/ch4.pdf>
- [26] *LightGBM's documentation* <https://lightgbm.readthedocs.io/en/latest/>