# REAL-TIME ANOMALY DETECTION OF GAMMA-RAY BURSTS FOR THE CHERENKOV TELESCOPE ARRAY USING DEEP LEARNING

---


*Presentata da*
Leonardo Baroncelli

*Supervisore*
Andrea Bulgarelli

*Coordinatore Dottorato*
Daniele Bonacorsi

*Co-supervisore*
Antonio Zoccoli

**Alma Mater Studiorum - Università di Bologna**


**Settore Concorsuale**: 02/A1 - FISICA SPERIMENTALE DELLE INTERAZIONI FONDAMENTALI

**Settore Scientifico Disciplinare**: ING-INF/05 - SISTEMI DI ELABORAZIONE DELLE INFORMAZIONI

**DOTTORATO DI RICERCA IN**
**DATA SCIENCE AND COMPUTATION**

Ciclo 34

**Esame finale anno 2023**

*Real-Time Anomaly Detection of Gamma-Ray Bursts for the Cherenkov Telescope Array using Deep Learning,* © May 2023

Author:
Leonardo Baroncelli

Supervisors:
Prof. Antonio Zoccoli
Dott. Andrea Bulgarelli

Institute:
ALMA MATER STUDIORUM - Università di Bologna - Via Zamboni, 33 - 40126 Bologna

# CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# ABSTRACT

*Context*. The Cherenkov Telescope Array (CTA) will be the next-generation ground-based observatory to study the universe in the very-high-energy domain. It will bring advancements to gamma-ray astronomy by deploying over sixty highly sensitive Cherenkov telescopes, with a sensitivity improvement of one order of magnitude compared to current Imaging Atmospheric Cherenkov Telescopes (IACTs). The observatory will rely on a Science Alert Generation (SAG) system to analyze the real-time data from the telescopes and generate science alerts. The SAG system will play a crucial role in the search and follow-up of transients from external alerts, enabling multi-wavelength and multi-messenger collaborations. The observatory will operate arrays on sites in both hemispheres to provide full sky coverage. It will maximize the potential for the detection of the rarest phenomena, such as gamma-ray bursts (GRBs), which are the science case for this study.
*Aims*. The thesis aims to investigate the usage of anomaly detection for real-time gamma-ray analysis. A deep learning-based technique has been developed to pursue this goal.
*Results*. This study presents an anomaly detection method based on deep learning for detecting gamma-ray burst events in real-time. The performance of the proposed method is evaluated and compared against the Li&Ma standard technique in two use cases of *serendipitous discoveries* and *follow-up observations*, using short exposure times. The method shows promising results in detecting GRBs and is flexible enough to allow real-time search for transient events on multiple time scales. The method does not assume background nor source models and doe not require a minimum number of photon counts to perform analysis, making it well-suited for real-time analysis.
*Conclusions*. Future improvements involve further tests, relaxing some of the assumptions made in this study as well as post-trials correction of the detection significance. Moreover, the ability to detect other transient classes in different scenarios must be investigated for completeness. The system can be integrated within the SAG system of CTA and deployed on the onsite computing clusters. This would provide valuable insights into the method's performance in a real-world setting and be another valuable tool for discovering new transient events in real-time. Overall, this study makes a significant contribution to the field of astrophysics by demonstrating the effectiveness of deep learning-based anomaly detection techniques for real-time source detection in gamma-ray astronomy.

# SOMMARIO

*Contesto*. Il Cherenkov Telescope Array (CTA) sarà l'osservatorio terrestre di prossima generazione per lo studio dell'universo nel dominio delle altissime energie. Sarà composto da più di sessanta telescopi Cherenkov di nuova generazione che migliorano la sensibilità di un ordine di grandezza rispetto agli attuali Imaging Atmospheric Cherenkov Telescopes (IACT), portando grandi contributi all'astrofisica delle alte energie. L'osservatorio sfrutterà il sistema software di Science Alert Generation (SAG) per l'analisi in tempo reale dei dati osservativi e per generare automaticamente allerte scientifiche. Il sistema SAG svolgerà un ruolo da protagonista nella ricerca e nel follow-up di fenomeni transienti a seguito di allerte esterne, consentendo collaborazioni multi-wavelength e multi-messenger. L'osservatorio comprenderà due array di telescopi su due siti in entrambi gli emisferi per fornire la piena copertura del cielo, massimizzando la capacità di rilevare i fenomeni più rari, come i lampi di raggi gamma (GRBs), che sono il caso scientifico di questo studio.
*Obiettivi*. Lo scopo di questo studio è indagare l'utilizzo della tecnica di anomaly detection per l'analisi in tempo reale dei dati gamma. È stata quindi sviluppata una tecnica basata sul deep learning per perseguire questo obiettivo.
*Risultati*. Questo studio presenta una tecnica di anomaly detection basata sul deep learning per la rilevazione in tempo reale di GRBs. Le prestazioni della tecnica proposta sono valutate e confrontate con la tecnica standard di Li&Ma, nei due casi d'uso scientifici di *serendipitous discoveries* e *follow-up observations*, considerando brevi tempi di esposizione. La tecnica proposta mostra risultati promettenti e è abbastanza flessibile da consentire la ricerca di eventi transienti su più tempi scala. Non necessita di fare ipotesi sui modelli del background e della sorgente e non richiede un numero minimo di conteggi di fotoni per eseguire l'analisi, rendendola adatta per l'analisi in tempo reale.
*Conclusioni*. Miglioramenti futuri includono ulteriori test, accantonando alcune delle ipotesi semplificative assunte in questo studio, così come la correzione post-trial della significatività di rilevazione. Inoltre, dovrà essere testata la capacità di rilevare altre classi di transienti oltre ai GRBs. L'integrazione all'interno del sistema SAG e la messa in produzione nei centri di calcolo onsite, fornirebbe preziose informazioni sulle prestazioni del metodo con dati non simulati. Nel complesso, questo studio fornisce un contributo significativo al campo dell'astrofisica delle alte energie e dimostra l'efficacia della tecnica di anomaly detection per la rilevazione in tempo reale di fenomeni transienti.

# ACKNOWLEDGEMENTS

Completing a Ph.D. thesis is a long and challenging journey that would not have been possible without many people's invaluable support, encouragement, and guidance. I want to thank the following individuals and institutions who played a significant role in making this possible.

First and foremost, I would like to thank Dott. Andrea Bulgarelli for his unwavering support, guidance, and mentorship throughout my Ph.D. studies. I am equally grateful to Dott. Antonio Zoccoli, for being my supervisor despite his busy schedule. I want to express my sincere gratitude to Dott. Ruben Lopez-Coto and Dott. Domenico Beneventano for their invaluable contribution as external reviewers of my Ph.D. thesis. I am deeply grateful to Fondazione Golinelli for providing me with the scholarship that made my Ph.D. studies possible.

I would also like to thank my colleagues Ambra di Piano, Giovanni de Cesare, Ettore Bronzini, Gabriele Panebianco, Nicolò Parmiggiani, Antonio Addis, Sami Caroff, Pierre Aubert, Enrique Garcia and Thomas Vuillaume for the valuable discussions and for creating a supportive and stimulating research environment.

My heartfelt thanks also go to Mónika Takács for the experiences we have shared and the love we have exchanged. She will be forever in my heart.

Lastly, I would like to express my gratitude to my parents, that supported me in every way possible.

I am genuinely grateful to those I did not mention here who have supported me during this period. Your support, encouragement, and guidance have made this Ph.D. journey a memorable and rewarding experience.

# DEDICATION

*To Giancarlo, my beloved grandpa*

# THESIS OUTLINE

This Ph.D. thesis investigates deep learning-based anomaly detection for real-time gamma-ray analysis in the astrophysics domain. A method was developed to detect gamma-ray burst events in real-time and was evaluated against the Li&Ma standard technique. The proposed method shows promising results in detecting gamma-ray bursts. It is flexible enough for real-time search on multiple time scales. Hence it has the potential to be integrated within the Science Alert Generation system of the Cherenkov Telescope Array, the next-generation observatory for observing the universe in the very-high-energy domain. This study provides valuable insights into the effectiveness of deep learning-based anomaly detection techniques for real-time gamma-ray analysis and significantly contributes to this field.

## 0.1 CONTRIBUTIONS

Our key contributions include the following:

1. An anomaly detection method has been developed to perform real-time source detection of gamma-ray transients.
2. This work contributed to developing the Science Alert Generation system in the context of the Array Control and Data Acquisition control software of the Cherenkov Telescope Array Observatory (CTAO).

## 0.2 OUTLINE

The remainder of this thesis is organized as follows:

**Chapter 1** gives the reader the required background to understand the context of this work. Section 1.1 discusses gamma-ray astronomy and ground-based gamma-ray imaging with Cherenkov telescopes. Section 1.2 introduces the CTAO, its science goals, its telescope array, and its associated computing and software systems, including the Science Alert Generation system. Finally, it explores the scientific use cases of serendipitous discoveries and follow-up observations linked to the real-time detection of transient events. Section 1.3 covers gamma-ray data analysis techniques, including the full field of view maximum likelihood and the aperture photometry. It also describes

the Li&Ma significance estimation method in the context of a reflected regions background estimation algorithm. Finally, Section 1.4 covers the study of gamma-ray bursts, the transient phenomena this work is focused on.

**Chapter 2** introduces the concept of anomaly detection for time series analysis, discussing the major existing contributions to the field. Section 2.1 introduces the definition and properties of time series data and the concept of anomaly. It discusses several techniques from the scientific literature, classifying them using a taxonomy. Section 2.2 focuses on anomaly detection techniques based on deep learning. The method developed in this work belongs to this category. Section 2.3 lists several contributions made within the astrophysics domain, proving that these techniques are becoming increasingly popular for analyzing astrophysical data.

**Chapter 3** presents the proposed method to address the real-time source detection problem introduced in Section 1.2.4. It is organized as follows: Section 3.1 describes the proposed anomaly detection technique. It presents the data pipeline that has been developed to generate input data, the deep learning architectures that have been investigated, and the training process. The evaluation of the models will be addressed in Section 4.1. Section 3.2 describes the p-value analysis to associate each classification with a Gaussian statistical significance of positive detection. Section 3.3 and Section 3.4 explain how the overall system works, from configuration to inference. Section 3.5 investigates several problems that can arise during the telescope observations and how those problems can affect the proposed system.

**Chapter 4** presents the results of the p-value analyses and performance benchmarks. Section 4.1 reintroduces the scientific use cases and the assumptions made in these scenarios. It then describes the test set generation process. Section 4.2 presents the results of the p-value analysis. Section 4.3 shows a comparison between the two investigated autoencoder architectures. Section 4.4 outlines the performances of the proposed anomaly detection method against the Li&Ma standard technique. The key performance indicators used for the comparison are introduced. The results for both use cases of serendipitous discoveries and follow-up observations are presented in the short-term and very short-term scenarios.

**Chapter 5** summarizes the key outcomes from this study to reach final conclusions. It also highlights potential improvement areas, including further testing and feature developments. Lastly, the future outlook for this research will be discussed at the conclusion of the chapter.

# 1

# GAMMA-RAY ASTRONOMY AND THE CHERENKOV TELESCOPE ARRAY OBSERVATORY

This introductory chapter gives the reader the required background to understand the context of this work. It is organized as follows: Section 1.1 discusses gamma-ray astronomy and ground-based gamma-ray imaging with Cherenkov telescopes. Section 1.2 introduces the Cherenkov Telescope Array Observatory (CTAO), the science goals, an overview of the telescope array, and its associated computing and software systems, including the Science Alert Generation System. Finally, it explores the scientific use cases of serendipitous discoveries and follow-up observations linked to the real-time detection of transient events. Section 1.3 covers gamma-ray data analysis techniques, including the full field of view maximum likelihood and the aperture photometry. It also describes the Li&Ma significance estimation method in the context of a reflected regions background estimation algorithm. Finally, Section 1.4 covers the study of gamma-ray bursts, the transient phenomena this work is focused on.

## 1.1 GAMMA-RAY ASTRONOMY

Gamma-ray astronomy studies the most energetic electromagnetic radiation in the universe, with energies ranging from a few hundred KeV to several PeV. The gamma photons are produced by some of the universe's most extreme and violent processes, including supernovae, black holes, and neutron stars [FM95]. Therefore, the study of gamma rays provides a unique window into the physics of these extreme environments. Two main components dominate the gamma-ray sky: known sources and the diffuse gamma-ray background emission [Ack+15]. Known sources are objects identified and studied in detail, such as active galactic nuclei (AGN), pulsars, and supernovae. These sources are relatively bright and can be

easily detected by gamma-ray telescopes [Abd+10]. On the other hand, the diffuse gamma-ray background is a faint and diffuse emission present throughout the entire gamma-ray sky [Ack+15]. This background emission is thought to be produced by the collective emission of many faint sources that are too faint to be detected individually [Abd+10]. It is also thought to be produced by the interaction of cosmic rays with the interstellar medium and by the decay of radioactive isotopes. Cosmic rays are another type of high-energy radiation. They are produced by accelerating charged particles, such as electrons or protons, to very high energies through various processes, including the collapse of massive stars, the acceleration of particles in magnetic fields, and the collision of particles in high-energy environments [Bul+19]. One of the main challenges in studying the gamma-ray sky is the difficulty separating known sources' contribution from the diffuse background emission [Ack+15]. This is because the background emission is much brighter than the individual sources, making it difficult to study them in detail [Abd+10]. In addition, gamma-ray is background dominated since, among the detected events, hadrons are 1 thousand times more frequent than gamma photons. The study of gamma rays has a long history, dating back to the discovery of cosmic rays by Victor Hess in 1912. However, it was not until the development of space-based instruments in the 1960s and 1970s that gamma-ray astronomy became a mainstream field of study. One of the first satellite missions to significantly contribute to the field was the Gamma Ray Observatory (GRO), launched by NASA in 1991. GRO used instruments to detect and measure gamma rays from various sources, including the Crab Nebula, the Cygnus X-1 binary system, and the Galactic Center [Mat+96]. In the decades since the launch of GRO, advances in detector technology and instrumentation have allowed for the development of increasingly sensitive gamma-ray telescopes. The Swift Gamma-Ray Burst Mission, launched by NASA in 2004, is specifically designed to detect and study gamma-ray bursts [Geh+04]. The AGILE satellite, launched by the Italian Space Agency in 2007, is equipped with a gamma-ray imager instrument called GRID. AGILE has made several important contributions to gamma-ray astronomy, including detecting the gamma-ray emissions from the Milky Way's central region and detecting a class of gamma-ray sources called *microquasars* [Tav+09]. The Fermi Gamma-ray Space Telescope was launched by NASA in 2008 and has discovered many new gamma-ray sources, including active galactic nuclei, pulsars, and gamma-ray bursts [Abd+10]. Ground-based telescopes provide an alternative approach to studying the high-energy gamma-ray spectrum, complementing the data obtained by satellite-based detectors. Space telescopes have been developed because gamma-ray radiation is opaque to the Earth's atmosphere. However, space satellites can capture gamma rays up to a certain energy. For gamma rays with higher energies, the dimensions and density of the detector make it infeasible to launch them into orbit. In order to

capture the most energetic gamma rays, ground-based telescopes use the earth's atmosphere as a detector. These telescopes, such as the Cherenkov Telescope Array (CTA), are designed to detect the secondary products of gamma rays as they interact with the atmosphere. The most common secondary product detected is Cherenkov radiation, but other products, such as muons, can also be observed. The atmospheric Cherenkov technique is one way to detect gamma-ray radiation that ground-based gamma-ray telescopes can use. It works by detecting the faint flashes of blue light created when high-energy gamma rays collide with the upper atmosphere. The first generation of atmospheric Cherenkov telescopes (IACTs) was the Whipple 10m telescope, the HEGRA array, and the CAT telescope. These telescopes were followed by the current generation of IACTs, including the High Energy Stereoscopic System (H.E.S.S.) [Ste00], the Major Atmospheric Gamma Imaging Cherenkov (MAGIC) telescopes [PM99], and the Very Energetic Radiation Imaging Telescope Array System (VERITAS) [Wee+02]. The CTA observatory currently under construction is the next generation of ground-based gamma-ray telescopes. CTA will consist of two arrays of telescopes, one in the northern and one in the southern hemispheres, with more than 60 telescopes. This array will have a large field of view (FOV) and a sensitivity that is an order of magnitude better than current IACTs, making it possible to study gamma-ray sources with unprecedented precision. Section 1.2.2 will provide more details on the telescope array configurations.

### 1.1.1 *Ground-Based Gamma-Ray Imaging with Cherenkov Telescopes*

Cherenkov telescopes are designed to detect gamma-ray photons by observing the Cherenkov radiation produced when high-energy particles pass through the Earth's atmosphere. High energy particles interact with Earth's atmosphere, producing subatomic particles and radiation cascades. These cascades are also known as air or particle showers. The particles in these showers can travel faster than light in the air, which creates a flash of blue and ultra-violet light (Cherenkov effect), similar to how a sonic boom is created by an aircraft exceeding the speed of sound. While the light is spread over a large area, the cascade only lasts a few billionths of a second [Ong09]. The telescopes' mirrors reflect the Cherenkov radiation, and the camera, typically composed of photomultiplier tubes or charged coupled devices, captures the signal. The shower produced by a gamma photon projects an approximately elliptical shape on the camera, which is analyzed to reconstruct the primary gamma ray's origin, energy, and direction. This process is shown in Figure 1.1.

Electromagnetic showers, like those initiated by gamma rays, are different from hadronic showers initiated by cosmic rays (protons and nuclei) that have larger structures and can therefore be distinguished. The air showers initiated by electron-

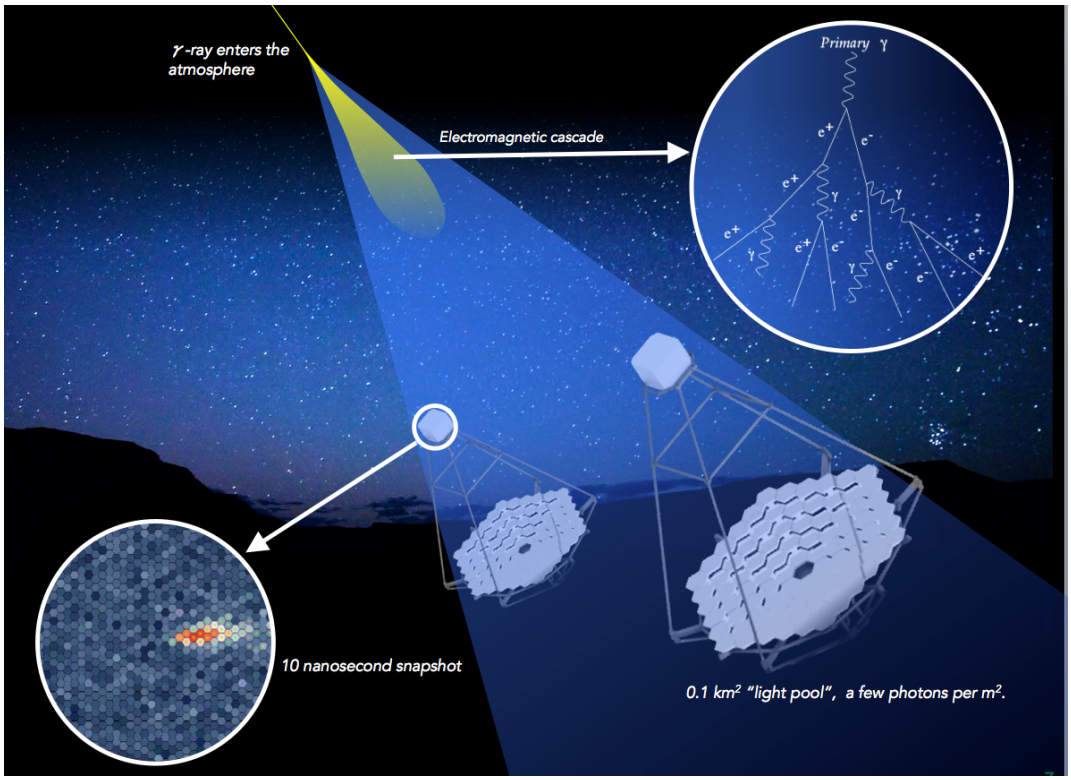


Figure 1.1: Image explaining how Cherenkov Telescopes detect Cherenkov light produced by a primary gamma-ray interacting with the Earth's atmosphere. Credits to [Con23a].

s/positrons constitute an irreducible isotropic background for the telescopes. The efficiency in discriminating between hadronic and gamma-ray showers is one parameter, among many others, that defines the telescope's sensitivity [Tam20]. The flux quantity ($ph * cm^{-2} * s^{-1}$) describes how much light a source emits per unit of time and surface, and sensitivity is the minimum flux value detectable with a confidence level of $5\sigma$. It depends on several parameters, such as the analysis algorithm, the observing conditions, the instrument response function, and the sub-array configuration. After the raw data is collected, an image-cleaning process occurs to reconstruct the shower information needed to perform the parametrization of the shower itself. A. M. Hillas introduced the fundamental image parameters in 1985 when a single telescope configuration was still used [Hil85]. These parameters are:

- Size: total number of photoelectrons in the shower image. It is proportional to the energy of the incoming primary gamma-ray or particle;
- Length: is half of the major axis of the shower image;
- Width: is half of the minor axis of the shower image;
- Frac: measures the general concentration of light;

- Miss: is the perpendicular distance of the center of the field (where the source is supposed to be in a single telescope configuration and pointing in the center of the field of view) from the image axis;
- Azimuthal-Width: is the image width relative to a new axis which joins the center of the field to the centroid of the image;
- Distance: is the distance of the brightest point from the center of the field.

These parameters allow for evaluating the obtained Cherenkov images and achieving good gamma-ray/hadron separation. New parameters are used to track the source position when the source is not in the camera center (see Section 1.3.2.2):

- Alpha: is the angle between the major axis of the ellipse and the direction of the source position from the center of gravity of the image;
- Dist: is the distance of the center of gravity of the image from the source position.

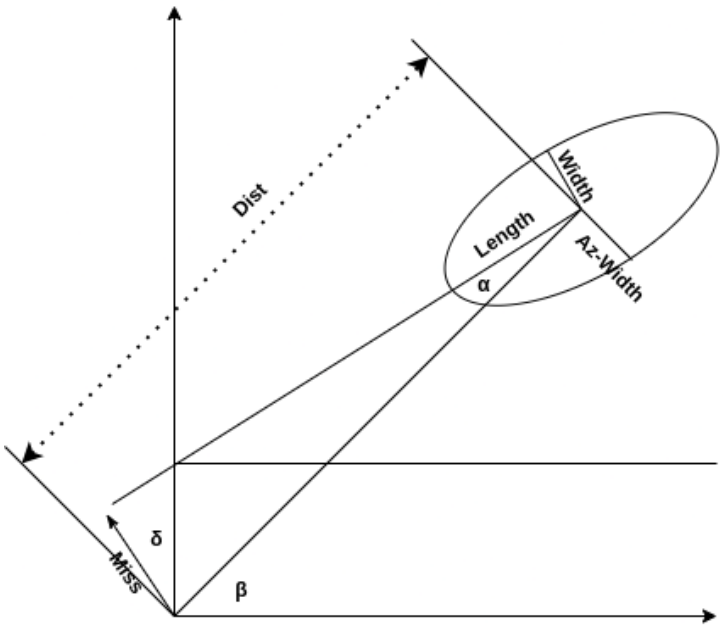


Figure 1.2: Parametrization of a shower image through Hillas parameters.

Figure 1.2 shows the Hillas parametrization of a shower image. As we will see in Section 1.2.2, the CTA Observatory will use an array of Cherenkov telescopes to observe individual showers to reconstruct the three-dimensional origin of gamma-ray showers. This technique, known as stereoscopic reconstruction, allows more effective reconstruction and discrimination of gamma-ray showers from hadronic showers produced by isotropic cosmic rays. The stereo reconstruction increases the background suppression efficiency by about 100 times, improves the angular resolution, and enables morphological studies of extended gamma-ray sources [Spu18].

These observations are analyzed using reconstruction algorithms based on Hillas' parameters and implemented using statistical techniques like the Random Forest regression method [Aa08]. After the data is reconstructed, it is collected in the form of a photons list, i.e., a collection of individual gamma-ray photons along with the detection time, reconstructed direction, and reconstructed energy.

## 1.2 THE CHERENKOV TELESCOPE ARRAY OBSERVATORY

The CTA Observatory is a next-generation ground-based gamma-ray observatory currently under development. CTA is a collaboration of over 1200 scientists from 32 countries and is expected to be the most sensitive and highest-resolution gamma-ray observatory ever built [Sci].

### 1.2.1 *Science goals*

CTA will address a wide range of major questions in and beyond astrophysics. The Core Programme [Sci] provides a comprehensive discussion. In brief, the science goals can be grouped into three broad themes.

The first theme is understanding the origin and role of relativistic cosmic particles. Relativistic particles play a significant role in many astrophysical systems, from pulsars and supernova remnants to active galactic nuclei and clusters of galaxies. However, the relationship between these particles, the turbulent motions of gas, and magnetic fields within our own galaxy and their overall impact on star formation and galaxy evolution is not well understood. CTA will provide the first high-resolution measurements of cosmic-ray protons and nuclei in astrophysical systems, giving insight into the processes of acceleration, transport, and feedback mechanisms. Historically, non-thermal effects in astrophysical systems haven't been taken into account or approximated due to a lack of data. The insights from CTA will significantly contribute to our understanding of galaxy and cluster evolution in the era of precision astrophysics. The main goal of gamma-ray astrophysics has been to identify where particle acceleration occurs and determine the main contributors to locally measured cosmic rays, mostly protons and nuclei. Progress has been made in the last decade. Still, key questions remain unanswered, such as whether supernova remnants are the only major contributors to Galactic cosmic rays and the sources of high-energy cosmic electrons and ultra-high-energy cosmic rays. The question of how and where particles are accelerated in the universe is also important, as well as the role these particles play in the evolution of their host objects and how they are transported to large distances.

The second theme is about probing extreme environments. The acceleration of particles to extremely high energies is often linked to extreme environments, such

as those found near neutron stars and black holes or in relativistic outflows or explosions. Very high-energy (VHE) emissions from these accelerated particles can probe these environments, providing access to time and distance scales that are not accessible through other wavebands. VHE emission also often escapes from systems where UV and X-ray emission is absorbed, offering information independent of assumptions about magnetic field strengths. Furthermore, VHE photons from distant objects can probe the intervening space. CTA will allow us to measure the redshift evolution of the UV-IR background and, thus, the star-formation history of the universe and probe magnetic fields in cosmic voids at levels many orders of magnitude below the reach of any other technique. CTA will also determine if VHE photons heat the gas in these under-dense regions, potentially suppressing the formation of dwarf satellite galaxies.

The third theme is exploring frontiers in physics. CTA has the potential to make significant discoveries in the field of fundamental physics. CTA will reach the expected thermal relic cross-section for self-annihilating dark matter for a wide range of dark matter masses, including those inaccessible to the Large Hadron Collider. The long travel times of gamma rays from extra-galactic sources, combined with their short wavelength, make them a sensitive probe for energy-dependent variations of the speed of light due to quantum-gravity-induced fluctuations of the metric. CTA will be sensitive to these effects on the expected characteristic scale, the Planck scale. Gamma rays may also couple with other light particles, such as axion-like particles (ALPs), under the influence of intergalactic magnetic fields. This effectively makes the universe more transparent to gamma rays and introduces a spectral modulation. Each of these effects would represent a major discovery and justify the effort of constructing and operating CTA. CTA's increased sensitivity and energy coverage bring these effects within reach and could allow for further discoveries in fundamental physics.

### 1.2.2 *The telescope array*

Cherenkov telescopes can be disposed to form a detector array. The array captures the elusive cascades of gamma-ray photons produced when high-energy gamma rays hit the Earth's atmosphere. These cascades are incredibly rare, with only one gamma-ray photon per square meter per year from a bright source or one per square meter per century from a faint source. CTA will use more than 60 telescopes distributed between two array sites in the northern and southern hemispheres to improve the chances of capturing these elusive signals. The northern hemisphere array will have a more limited size and focus on the low and mid-energy range of CTA, between 20 GeV and 5 TeV. Meanwhile, the southern hemisphere array, which has a prime view of the rich central region of our galaxy, will cover the

mid- to the high-energy range of CTA, spanning gamma-ray energies from 150 GeV to 300 TeV [Ach+19]. Figure 1.3 shows the sensitivity of the northern and southern telescope arrays compared to other instruments. To cover the full CTA

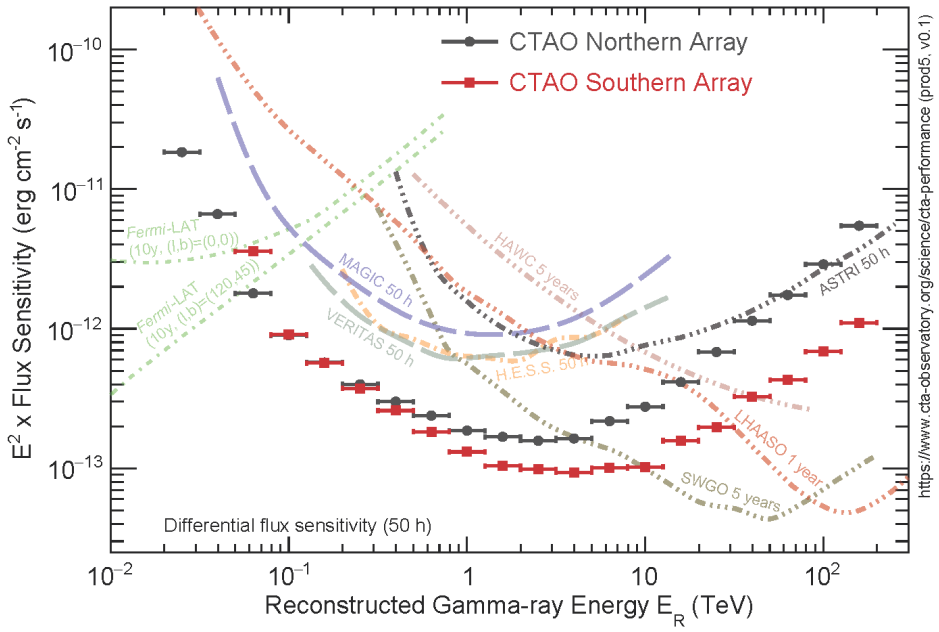


Figure 1.3: The sensitivity of the CTA Northern and Southern arrays compared to other instruments. Credits to [OC21]

energy range, three classes of telescopes are required: Small-Sized Telescope (SST), Medium-Sized Telescope (MST), and Large-Sized Telescope (LST). The SSTs are the most sensitive to high-energy gamma rays and are, therefore, more suitable for the southern site's detection of higher-energy gamma rays. In the southern hemisphere, the galactic center has unobstructed visibility, a rich source of various types of emission. As a result, detecting low-energy gamma rays in this region is particularly challenging due to the high signal-to-noise ratio and the contribution of the diffuse background from the interstellar gas.

Scientists have conducted various simulations to determine the best configurations of arrays to maximize performance, including sub-array configurations. These configurations (shown in Figure 1.4 and Figure 1.5) could allow for parallel and independent observations, resulting in optimized observation time and a more specific focus on science [Ach+19]. The Alpha Configuration is the approved layout of the telescope arrays in both the northern and southern hemispheres. This configuration includes 13 telescopes, 4 LSTs, and 9 MSTs, distributed over an area of about $0.5km^2$ in the CTA Northern Array and 51 telescopes over a $3km^2$ area, consisting of 14 MSTs and 37 SSTs, in the CTA Southern Array. The collaboration between the northern and southern hemispheres will allow CTA to expand its reach and study the universe more comprehensively [Ach+19]. The northern

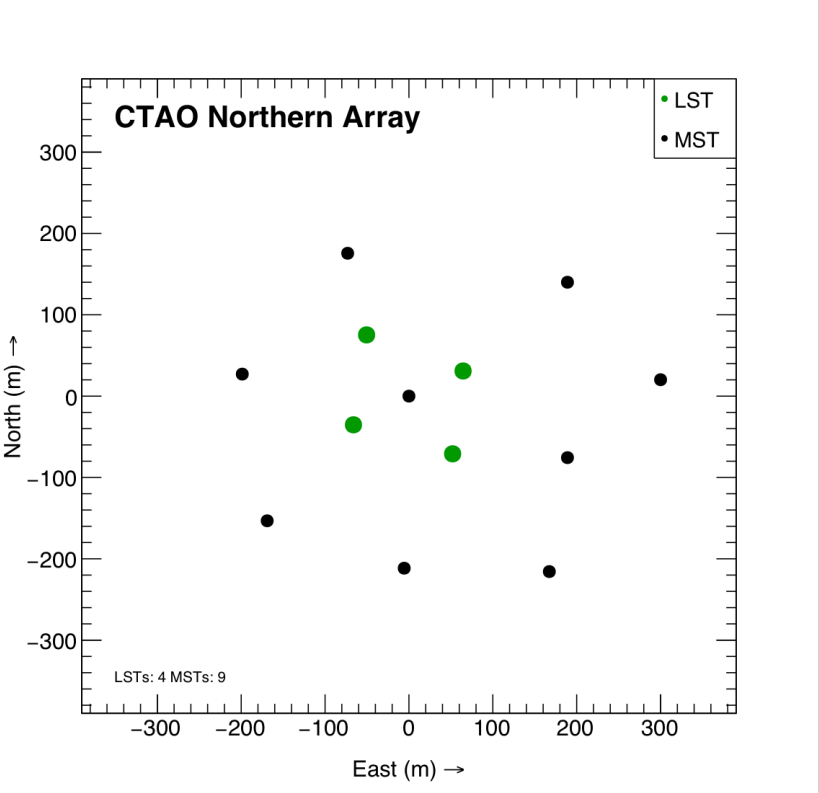


Figure 1.4: Layout of the CTA Northern array on La Palma (Spain), including the elements defined in the Alpha Configuration. Credits to [OC21]

hemisphere array will be located at the existing Instituto de Astrofisica de Canarias' (IAC's) Observatorio del Roque de Los Muchachos on the island of La Palma in the Canary Islands. This site already hosts an operating gamma-ray observatory, the Major Atmospheric Gamma Ray Imaging Cherenkov (MAGIC) telescopes, and various optical telescopes of various sizes. The Southern Hemisphere Array will be located in the Atacama Desert in Chile, near the European Southern Observatory's (ESO's) existing Paranal Observatory.

In 2022 the European Union's National Recovery and Resilience Plan (PNRR) was released following the COVID-19 pandemic. The Italian National Institute for Astrophysics (INAF), in collaboration with the Italian National Institute for Nuclear Physics (INFN) and in partnership with CTAO, submitted the CTA+ Project to Italy's Ministry for University and Research. The project was awarded 70 million euros after a competitive review by international referees and was announced in June 2022. The program includes Work Packages (WP) to aid in implementing CTAO as part of Italy's participation in the CTAO ERIC. The WP focuses on upgrading the CTAO Southern Array in Chile and includes two new Large-Sized Telescopes (LSTs), electromechanical structures for five new Small-Sized Telescopes

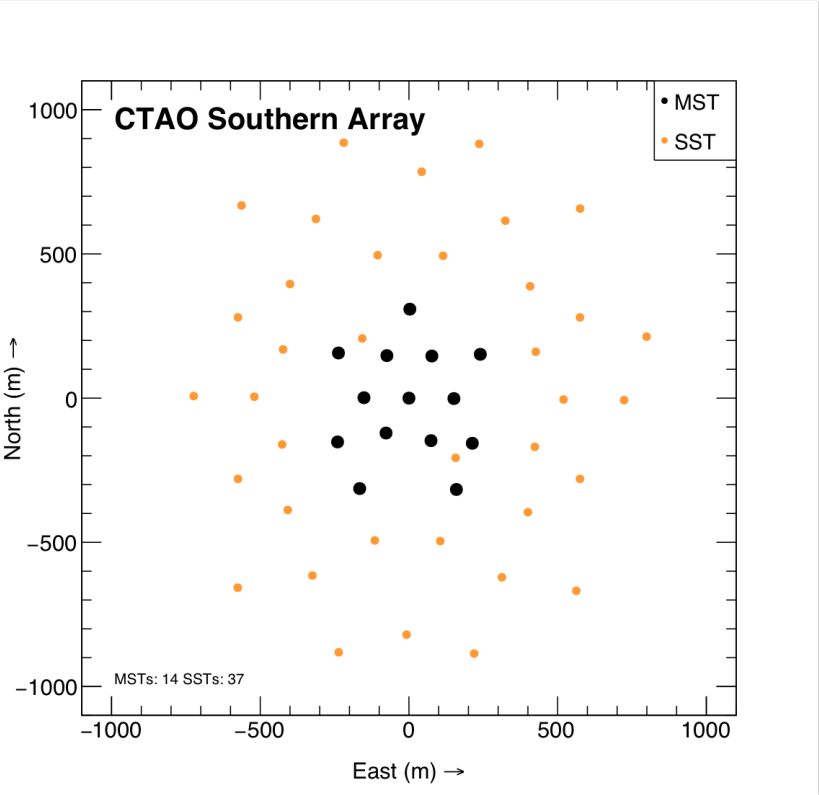


Figure 1.5: Layout of the CTA Southern array in the Atacama Desert (Chile), according to the Alpha Configuration. Credits to [OC21]

(SSTs), improvements for transient source science, and R&D on new detectors for Cherenkov telescopes [Con23a].

#### 1.2.2.1 *Large-Sized Telescopes (LST)*

This section describes the Large-Sized Telescope (LST) ([Bar20], [Con23b], and [Con23a]). The LST project is a collaboration involving over 100 scientists from 10 countries, including Brazil, Croatia, France, Germany, India, Italy, Japan, Poland, Spain, and Sweden. The LST is an alt-azimuth telescope that uses a reflective surface of 400 square meters designed to capture images of low-energy gamma rays, which produce a small amount of Cherenkov light. The LST uses a large mirror with a $23m$ diameter parabolic reflective surface supported by a tubular structure of reinforced carbon fiber and steel tubes (Figure 1.6). The LST mirror collects and focuses the Cherenkov light into the camera, comprising photomultiplier tubes that convert the light into electrical signals that dedicated electronics can process. The camera shares many elements with the NectarCAM for the MSTs. It has a total field of view of about 4.3 degrees and is designed to be compact and lightweight while providing optimal performance at low energies. Each pixel incorporates a photosensor and the corresponding readout electronics. These electronics are based on

Figure 1.6: The LST Prototype, LST-1, on the CTA-North site in La Palma. Credits to https://www.flickr.com/photos/cta_observatory/albums/72157671493684827

the Domino Ring Sampler Version 4 (DRS4) chip, developed at the Paul Scherrer Institute in Switzerland and currently used by several experiments, including the MAGIC Cherenkov telescopes. The camera trigger strategy is based on the shower topology and the temporal evolution of the Cherenkov signal produced in the camera.

Although the LST is 45 meters tall and weighs around 100 tonnes, it is designed to be extremely nimble and able to re-position within 20 seconds. This re-positioning speed and low energy threshold are critical for CTA studies of galactic transients, high red-shift active galactic nuclei, and gamma-ray bursts. The LSTs will also expand the science reach to cosmological distances and fainter sources with soft energy spectra. Four LSTs will be placed at the center of the northern hemisphere array to detect high-energy photons between 20 and 150 GeV. The LSTs also have a very good sensitivity to energies of a few TeV, which is, however, more efficiently covered by Medium-Sized Telescopes (MSTs).

The LST prototype, LST-1, was completed in October 2018 in La Palma, Canary Islands, Spain, on the site of the Observatorio del Roque de Los Muchachos. The prototype is foreseen to become the first LST telescope of CTA and, in fact, the first telescope on a CTA site to be operated by the observatory. It will need to undergo a critical design review to verify that the design complies with CTA science goals, operational needs, safety standards, etc., before CTA formally accepts it. On October 10th, 2018, over 200 guests from around the world gathered to celebrate the

Figure 1.7: The MST design explores two different concepts. On the one hand, inspired by HESS and VERITAS current observatories, a 12 meters diameter modified Davis–Cotton (DC-MST) optical layout and a PMT camera. On the other, a novel 10 meters diameter Schwarzschild–Couder (SC-MST) optics incorporating a novel Silicon Photo-Multiplier (SiPM) camera. Credits to [Bar20].

inauguration of the prototype Large-Sized Telescope (LST-1). On December 14th, 2018, the LST-1 prototype recorded its first Cherenkov light. On November 23rd, 2019, LST-1 successfully detected its first gamma-ray signal when pointing to the Crab Nebula. Between January and February 2020, the LST-1 prototype observed the Crab Pulsar, a neutron star at the center of the Crab Nebula. These observations were used to verify the telescope's performance and capabilities.

#### 1.2.2.2 *Medium-Sized Telescopes (MST)*

This section describes the Medium-Sized Telescope (MST), referring [Bar20] and [Con23b]. The Medium-Sized Telescope (MST) is developed by an international team of institutes and universities from Austria, Germany, France, Brazil, Poland, Spain, Switzerland, and Italy. This telescope is considered the *workhorse* of the CTA, with sensitivity in the core energy range of CTA, from about 150 GeV to 5 TeV. The CTA plans to include 23 MSTs, with 14 in the southern and 9 in the northern hemispheres. The MST mirror will be about 12 meters in diameter and have two camera designs using photomultiplier tubes (PMTs). The MST is a modified Davies-Cotton telescope with a reflector diameter of 12 meters on a polar mount and a focal length of 16 meters. To create a uniform reflector, the MST will have up to 90 hexagonal-shaped mirrors aligned with an active mirror control assem-

Figure 1.8: The ASTRI telescope prototype, a novel dual-mirror Schwarzschild-Couder telescope design proposed for the CTA. Credits to [Con23a]

bly. The MST cameras will have a large field of view of about 8 degrees, making it easier to observe gamma-ray sources that may be concentrated in one area of the sky or widely spread apart. Two camera concepts are in development for the MST: NectarCAM and FlashCAM. NectarCAM uses the 'Nectar' analog pipeline ASIC for signal capture with GSample/s sampling rate and shares many design features/components with the LST camera. NectarCAM is composed of 265 individual and easily removable modules. FlashCAM design follows a horizontal architecture with the photon detector plane (PDP), the readout electronics (ROS), and the data acquisition system (DAQ) as key building blocks. The PDP contains photomultiplier tubes (PMTs) arranged in a hexagonal structure. An MST prototype was deployed in Berlin in 2012 and is currently undergoing performance testing. The main purpose of the prototype is to validate the design of the individual components, test the interfaces between the mating assemblies, and define the assembly process of the product. The prototype has a fully functional drive system, cameras for pointing and tracking, sensors designed to record the behavior response of the structure and drive system, and a weather station. The prototype is a fully-functioning telescope but doesn't include the entire camera assembly and its readout. Camera demonstrators were built, tested, and validated in parallel by the two camera sub-projects.

#### 1.2.2.3 *Small-Sized Telescopes (SST)*

This section describes the Small-Sized Telescope (SST) referring [Tag+22] and [Con23b]. The Small-Sized Telescopes will make up the largest number of telescopes in the CTA, with 37 planned to be spread out over several square kilometers in the southern hemisphere array only. This is because very high-energy gamma-ray showers produce a large amount of Cherenkov light over a large area. The SST's smaller mirror is sensitive to the highest energy gamma rays (between a few TeV and 300 TeV). The SSTs wide coverage and high sensitivity improve CTA's chances of detecting the highest energy gamma rays. Three different SST implementations were proposed for the final SST design: ASTRI-Horn, GCT, and SST-1M. In 2018, a harmonization process was initiated. In June 2019, the council accepted the CTA Management proposal, stating that *the CTA-SST design should be based on the ASTRI/CHEC design, taking into account the experience gained from all designs*. The SST design is a dual-mirror Schwarzschild-Couder aplanatic configuration, and thanks to its small plate scale, it uses a novel compact camera based on SiPM sensors. The 4.3m diameter primary mirror is segmented into hexagonal facets, and the 1.8m secondary mirror is monolithic. The SST's camera, also known as the CHEC camera, uses custom peak-hold application-specific integrated circuits (ASICs) for signal capture. The dual-mirror design allows for the same angular resolution and collecting area across a wide field of view with a short focal length. The camera comprises 2048 silicon photo-multiplier pixels forming approximately a 9x9 degrees field of view. The CHEC is unique as an SST dual-mirror camera in its ability to capture Cherenkov light not as fixed images but as movies consisting of hundreds of frames, each lasting one billionth of a second. The SST collaboration benefits from the research and development work previously carried out within the ASTRI, CHEC, and GCT projects to develop end-to-end SST dual-mirror telescopes. The ASTRI project is led by the Italian National Institute of Astrophysics (INAF) with the collaboration of several Italian universities, the Italian National Institute of Nuclear Physics (INFN), Universidade de São Paulo in Brazil, and North-West University in South Africa. The ASTRI project designed and installed an end-to-end dual-mirror prototype of the CTA small-size telescope (SST) on Mt. Etna (Sicily) and will install a dual-mirror SST mini-array composed of nine units at the CTA Southern site. The ASTRI mini-array will extend the current IACTs sensitivity well above a few tens of TeV [Ver16]. The CHEC project, led by MPIK, is an international collaboration between the University of Adelaide, the University of Amsterdam, DESY Zeuthen, Durham University, the Erlangen Centre for Astroparticle Physics (ECAP), the University of Leicester, the University of Liverpool, Nagoya University, and the University of Oxford. The Observatoire de Paris-Meudon carries out the GCT telescope project.

### 1.2.3 *Computing and software*

The CTA Computing Department faces the challenge of designing and implementing a system that supports all aspects of the CTA Observatory, from accepting observation proposals to scheduling observations, controlling the telescopes, processing and archiving data, and making it accessible to the public using open standards and FAIR (findability, accessibility, interoperability, and reusability) principles. The department is responsible for all stages of development, from architectural design to construction, validation, deployment, and maintenance. The Observatory's technical challenges and long lifetime will require new techniques and technologies to meet scientific demands. Even when built, maintaining the software and hardware systems for the 30-year lifespan of the observatory and the science data archive for an additional 10 years will not be a simple task. Therefore, software systems engineering is just as crucial as the code itself [Oa17]. In addition, the CTA array sites will generate a vast amount of data from the telescopes in hundreds of petabytes annually. This data will then be compressed and reduced to a few petabytes per year before being transferred to off-site data centers for processing and storage. Additionally, tens of petabytes of simulated data will also be generated and processed. The software systems that will satisfy all the CTA requirements are shown in Fig-

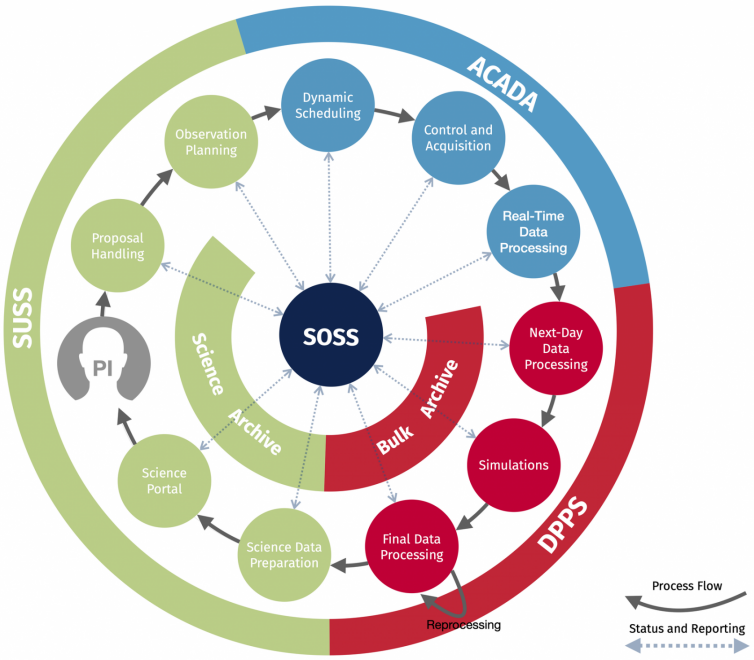


Figure 1.9: The diagram below provides a general overview of how the software systems interact with the main processes involved in the science operations of the observatory, including the submission, execution, and return of processed data related to a scientific proposal. Credits to [Con23b].

ure 1.9: the Array Control and Data Acquisition System (ACADA), the Data Processing and Preservation System (DPPS), the Science User Support System (SUSS), and the Science Operations Support System (SOSS) [Con23b]. The ACADA (Acquisition and Control and Data Analysis) system manages the supervision and control of telescopes and calibration instruments at both CTA array sites, including the efficient execution of scheduled and dynamically triggered observations. It also manages raw data acquisition and compression and generates automatic science alerts. The ACADA provides a user interface for site operators and astronomers [Oa17]. It's composed of several subsystems such as the Resource Manager and Central Control systems [Ma19], the Human Machine Interface system [Sad+17], the Short-Term Scheduler system [Ca14], and the Science Alert Generation system [Ba14]. The latter allows the observatory to reconstruct and analyze the data, detect sources, and issue candidate science alerts in real-time. The DPPS (Data Processing and Preservation System) is responsible for transforming the raw data products generated by ACADA into low-level science data products suitable for analysis, which are then delivered to the SUSS (Science User Support System) for dissemination. The DPPS ensures that all data products are preserved and replicated in at least two off-site data centers, have traceable and reproducible provenance, and are of the highest scientific quality. It also provides continuous monitoring and quality reporting for its sub-systems and produces high-level science quality metrics and reports related to the provided services. The DPPS is implemented as a distributed system, deployed as a set of Data Processing and Preservation Nodes, running at the CTA-North and CTA-South data centers, on three CTA off-site data centers, and at the SDMC Data Center [Con23a]. The SUSS manages the software system for high-level science operations workflows, from proposals to data delivery and user support. It is the main access point for exchanging science-related products with science users. It provides software for observation planning, the automatic generation and verification of high-level science data products, the Science Archive, Science Analysis Tools, and Science Portal, which provides access to software applications, services, data, and software products [Oa17]. The SOSS (Science Operations Support System) is a collection of software tools that support the systems involved in science operations workflows, such as ACADA, DPPS, and SUSS [Con23a]. The respective systems can access and share science operations-related information and configurations. It includes the means to track the state of proposals and observations throughout their life cycle and the state of the CTA Observatory throughout the science operations workflow and science performance.

### 1.2.4 *The Science Alert Generation System*

As described in [Bul+15] and [Bul+21], the Science Alert Generation system will perform real-time scientific analysis to issue science alerts. A science alert is a notification within the astrophysics community to share information about a transient phenomenon (such as an AGN's gamma-ray flare, gamma-ray bursts, gravitational waves, or galactic transients) that has been observed. Sharing this information through specialized communication networks allows coordination among different observatories to enable multi-wavelength and multi-messenger analysis. Those are rapidly developing fields that study celestial objects and phenomena using a wide range of electromagnetic and non-electromagnetic signals. This includes everything from radio waves and visible light to gamma rays, gravitational waves, and particles, such as cosmic rays and neutrinos. One of the key advantages of multi-wavelength and multi-messenger astronomy is that it allows astronomers to study the universe from a more comprehensive perspective. By combining data from different wavelengths and messengers, astronomers can better understand the physical processes in the universe. One of the most significant developments in multi-wavelength and multi-messenger astronomy has been detecting gravitational waves, ripples in space-time predicted by Einstein's theory of general relativity. The acceleration of massive objects, such as the merging of binary black holes or neutron stars, produces these waves. The first gravitational wave detection was made in 2015 by the Laser Interferometer Gravitational-Wave Observatory (LIGO) [Abb+16b]. Since then, several other gravitational wave detections have been made, opening up a new window onto the universe [Abb+16a]. The Science Alert Generation system will have a key role in the GW follow-up program [SA+19]. Another important aspect of multi-wavelength and multi-messenger astronomy is the study of transient phenomena, such as supernovae, gamma-ray bursts, and fast radio bursts. These events are often brief and elusive and can be studied more effectively by combining data from multiple telescopes and instruments. For example, the combination of data from X-ray, optical, and radio telescopes has allowed for the study of the afterglows of gamma-ray bursts (GRBs), which are thought to be the result of the collapse of massive stars or the merging of binary neutron stars [FM95]. To achieve collaboration between observatories, the transient phenomena must be observed in real-time on a short timescale. Furthermore, suppose an external observatory identifies a transient. In that case, it is essential to react quickly, change the pointing of the telescopes, and analyze the data in an automated and fast way to confirm the transient event and to conduct follow-up observations [Bul+21].

#### 1.2.4.1 *Requirements*

Technical challenges, such as limited network bandwidth at the observatory sites and a high expected data rate, make it difficult to perform real-time analysis off-site. As a result, an on-site analysis pipeline is necessary to access raw data, perform calibration, and produce scientific results [Bul+15]. The real-time analysis must meet challenging requirements to process the large amount of data produced by CTA in real-time. The Science Alert Generation system must be able to generate candidate science alerts within 20 seconds from the last acquired event, with a maximum telescope positioning time of 90 seconds in response to external or internal triggers. These candidate science alerts will be sent to the Transients Handler system, which will evaluate the results of the SAG to generate the final science alert to the community within 5 seconds of receiving it. If we also consider the 5 seconds required to acquire data, CTA will be capable of issuing science alerts with a maximum latency of 30 seconds [Bul+21]. Furthermore, the SAG system must be available during observations for at least 95% of the time to enable real-time follow-up of external alerts and internal alerts triggering serendipitous discoveries. This will require re-scheduling observations to follow up the phenomena in real-time and maximize the coordinated outcome of the facilities' network [DP+20]. According to the CTA design requirements, the real-time search for transient events should be performed on multiple time scales (from minutes to hours) with a sensitivity not worse than two times the nominal CTA sensitivity. [Fio+15] performed a preliminary evaluation of the real-time analysis sensitivity as a function of the CTA high-level technical performance (e.g., effective area, point spread function) and the observing time.

#### 1.2.4.2 *Structure*

The SAG system, shown in Figure 1.10, analyzes the real-time raw data captured by the telescopes' cameras to detect gamma-ray events, managing trigger rates of tens of kHz. It is a distributed system, and it consists of four main software packages written in different programming languages:

- SAG-RECO: *Image Parameter Extractor* and *Low-Level Reconstruction Pipeline*.
- SAG-DQ: *Online Data Quality Software*.
- SAG-SCI: *High-Level Analysis Pipeline*.
- SAG-SUP: *SAG Pipeline Sub-array Supervisor* and *SAG Supervisor*.

The SAG-RECO software package includes the *Image Parameter Extractor* and the *Low-Level Reconstruction Pipeline*. It processes the data stream from the telescopes' cameras, producing photon lists data with a maximum latency of 15 seconds. This software package is developed in C++ and exploits high-performance computing

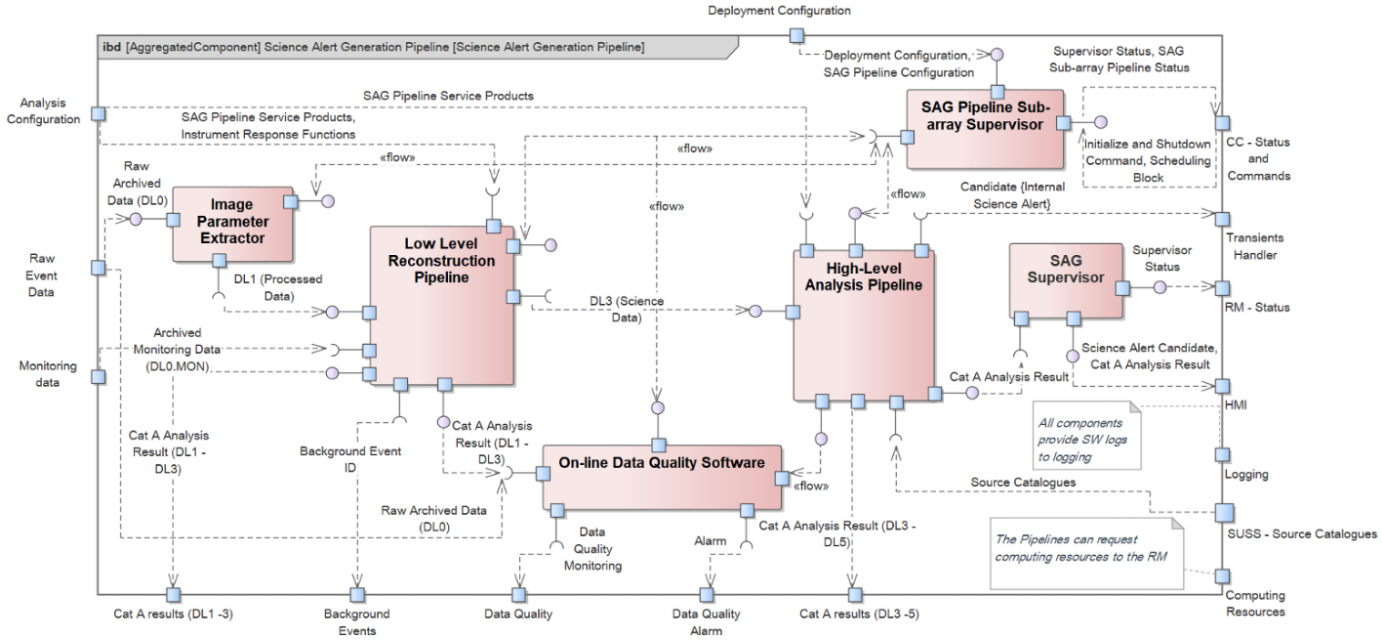


Figure 1.10: Context diagram for the SAG system. The SAG system is a distributed software package that analyzes raw data captured by telescopes' cameras to detect gamma-ray events in real-time. It consists of four main software packages written in different programming languages, which process the data stream, assess data quality, simplify the development of real-time scientific analysis pipelines, and supervise the operations of the analysis pipelines. The SAG-RECO package (*Image Parameter Extractor* and *Low-Level Reconstruction Pipeline*) is developed in C++, and the SAG-DQ (*Online Data Quality Software*) and SAG-SCI (*High-Level Analysis Pipeline*) packages are developed in Python. The SAG-SUP package (*SAG Pipeline Sub-array Supervisor* and *SAG Supervisor*) is implemented in Java using the Alma Common Software framework. Credits to [Bul+22].

techniques through an efficient data format generator, low-level optimization of CPU pipeline usage and vectorization of existing algorithms, and a fast integer compression algorithm [Aub18]. The SAG-DQ software package performs online data quality analysis to assess any degradation of the observing condition (due to hardware, software, or environmental problems). Two Python libraries have been developed to meet the data quality analysis requirements. The first package implements the core logic of data quality analysis, while the second manages the parallel execution of multiple data quality analysis pipelines. Both libraries have been developed by the author of this thesis [Bar+22]. The SAG-SCI software package implements a Python framework designed to simplify the development of real-time scientific analysis pipelines. This framework enables developers and researchers to focus on the scientific aspects of the pipelines, integrate existing science tools, and provide a common pipeline architecture with automation. It can also be easily configured with new or existing science tools. The scientific analyses are performed in parallel and can be prioritized, producing quick-look results and science-alert candidates [Par+22]. Finally, the SAG-SUP software package supervises the operations of the SAG-RECO, SAG-DQ, and SAG-SCI analysis pipelines, and it interfaces with the rest of the ACADA system. The *SAG Pipeline Sub-Array Supervisor* supervises the operation of the low-level, data quality, and high-level scientific analysis pipelines associated with an observation schedule. The *SAG Supervisor* manages the generation of scientific monitoring results and provides input to the ACADA *Reporting Subsystem* for generating reports. These two components are implemented in Java using the Alma Common Software framework [Chi+01]. The ALMA Common Software (ACS) has been developed to control the Atacama Large Millimeter Array (ALMA) observatory operations in the Chilean Atacama desert at a height of more than 5000 meters above sea level. Distributed Objects (DOs) serve as the core of ACS, an object model of controlled devices implemented as CORBA network objects. Each ACADA subsystem implements DOs that allow method-call interaction between object-oriented software distributed on different nodes. ACS allows the implementation of DOs using three different programming languages: C++, Python, and Java. The *SAG Pipeline Sub-Array Supervisor* ACS component has been implemented in Java, and it is, in turn, supervised by the *Array Supervisor* ACS component of the *Central Control* system. The observation schedule of the array is set by the human operator, which uses the graphic user interface provided by the *Human-Machine Interface* system. The observation schedule is encoded in a *scheduling block* object interchanged by the ACADA systems. In particular, the *Central Control* system reads the observation metadata and instantiates the resources needed to execute the observation, given the target and the telescope array configuration. Among them, the CC creates the Science Alert Generation (SAG) subsystem component to handle the real-time analysis pipelines, and it's

configured with the scheduling block. The SAG system relies on the Slurm Workload Manager software to supervise the analysis pipelines. The author of this thesis is the lead developer of the SAG-SUP software package.

### 1.2.5 *The scientific use cases*

The growing interest in multi-messenger and multi-wavelength astronomy has greatly enhanced the search for transients. As stated in Section 1.2.1, the KSP for transients plans to invest significant observation time per year (Figure 1.11). The main source classes targeted by this KSP are Gamma-Ray Bursts (GRBs), Galactic transients, X-ray, optical, and radio transients, High-energy neutrino transients, GW transients, and serendipitous VHE transients [Sci]. GRBs are expected to be a major component of this KSP and will be detected based on external alerts (covered in Section 1.2.5.2), but with the possibility of serendipitous discoveries (covered in Section 1.2.5.1).

| | | Observation times (h/yr/site) | | |
|---|---|---|---|---|
| Priority | Target class | Early phase | Years 1-2 | Years 3-10 |
| 1 | GW transients | 20 | 5 | 5 |
| 2 | HE neutrino transients | 20 | 5 | 5 |
| 3 | Serendipitous VHE transients | 100 | 25 | 25 |
| 4 | GRB | 50 | 50 | 10 |
| 5 | X-ray/optical/radio transients | 50 | 10 | 10 |
| 6 | Galactic transients | 150 | 30 | 0(?) |
| | Total per site (h/yr/site) | 390 | 125 | 95 |
| | Total both sites (h/yr) | 780 | 250 | 190 |
| | Total in different CTA phases (h) | 1560 | 500 | 1520 |

Figure 1.11: Maximum observation times required for follow-up targets in the Transient KSP, taken from. Credits to [DP+20]

#### 1.2.5.1 *Serendipitous discoveries*

Serendipitous discoveries are unexpected findings made while studying a specific target or phenomenon. This can occur because the telescope's field of view is wider than the region of interest used to observe a particular source, which allows for simultaneously detecting signals from multiple sources. However, the likelihood of a serendipitous GRB event appearing in the field of view during an observation is minimal. The probability of making serendipitous discoveries will be heightened during the survey phases of CTA as larger areas of the sky will be systematically observed. The event must be detected as soon as possible to broadcast a science alert to other observatories to follow the same event and enable multi-wavelength and multi-messenger analysis [Bul+15].

#### 1.2.5.2 *Follow-up observation triggered by an external science alert*

Follow-up observations refer to observing a specific target in more detail after an initial discovery or detection. The follow-up observations can be triggered by a science alert, which is a notification that alerts researchers and facilities of an interesting or unusual event happening. Follow-up observations can be conducted using various instruments and techniques. They are typically performed to study a target in more detail, confirm an initial discovery, and gather additional information about its properties [Sci]. When a science alert is received, the observatory will interrupt the current observation and point to a new sky region to detect the new source in the shortest possible time. If the error on the source's position is bigger than the field of view of the telescopes, multiple follow-up observations with a tiling strategy [Bul+15] are required to cover the whole localization region. Still, the latter scenario will not be covered in this work. In addition, the event's evolution cannot be observed from the beginning due to the delay between detecting the source that triggered the science alert, the time to receive it, and the time the telescopes took to change their pointing to the new sky region.

## 1.3 ANALYSIS OF GAMMA-RAY DATA

As outlined in Section 1.1.1, when a high-energy photon from a gamma-ray source collides with the Earth's atmosphere, it produces an electromagnetic shower that results in Cherenkov emission, which is then observed with an optical telescope on the ground. The light from the shower triggers the telescope, and the incoming signals are processed and classified to reconstruct the primary event's energy and arrival direction. However, observations are heavily affected by background events caused mainly by cosmic hadrons, and a significant effort is required to differentiate between electromagnetic events and background noise. The reconstruction outcome is a photon list, a collection of individual photons, their arrival time, reconstructed direction, and reconstructed energy. The high-level gamma-ray analysis starts at this stage to generate skymaps, spectra, and light curves and finally perform detections. This section will discuss the Aperture Photometry and Full-FOV Maximum Likelihood and techniques. The *Li&Ma* significance estimation method is also described in the context of a *reflected regions* background estimation algorithm. Their usage in the real-time scenario has been investigated by [Tam20], [DP+20], and [DP+21].

### 1.3.1 *The calibration database and the Instrument Response Functions*

The calibration database contains *instrument response functions* that are mandatory to perform simulation and analysis of CTA data. The instrument response function (IRF) $R(\boldsymbol{p}', E', t'|\boldsymbol{p}, E, t)$ describes the transformation from the physical properties of a photon (sky direction $\boldsymbol{p}$, energy E, and time t) to the measured characteristics of an event $\boldsymbol{p'}, E', t'$ (the instrument response function is given in units of $cm^2 sr^{-1} s^{-1} MeV^{-1}$) [Kno+16]. The IRF can be factorized in *effective area* $A_{eff}(\boldsymbol{p}, E, t)$ in units of $cm^2$, *point-spread function* $PSF(\boldsymbol{p}|p, E, t)$, and *energy dispersion* $E_{disp}(E'|\boldsymbol{p}, E, t)$. Furthermore, every IRF considers the background rates as they vary with energy and location within the field of view. The background rate mostly comprises cosmic-ray hadrons and electrons that pass the gamma-ray selection criteria (cuts) [DP+20]. The instrument's response is applied to the ideal simulation model during a simulation to obtain the data as the telescope saw it. Instead, the telescope observation data must be restored during the analysis to obtain the real quantities, considering the IRF.

### 1.3.2 *Aperture photometry*

Aperture photometry measures the light that falls inside a circular or annular region of some fixed size (aperture) around the object of interest (called the *on-region*). It then subtracts the contribution of any residual cosmic-ray background or noise sources that may be present in the surrounding area taken from one or more *off-regions*. This subtraction is designed to remove the contribution of any light or noise not directly associated with the object of interest, thus providing a more accurate measurement of its true brightness. To minimize the impact of background contamination and nearby sources on the data analysis, carefully select the regions used for the on-region and off-region. Ideally, the on-region should be chosen to contain the point spread function of the instrument at the given energy range, while the off-region should be of equal size and chosen to minimize the contamination of background or nearby sources. This ensures that the counts in the on-region are not underestimated or overestimated, providing a more accurate measurement of the gamma-ray source of interest. Still, it is also important to ensure that the background regions share similar characteristics with the source, including the instrument response function in that region. To determine the number of excess photons from the source, the normalized number of off-region events is subtracted from the on-region data. If the on-region and off-region share the same size, offset,

and exposure time, the probable number of photons contributed by the source is calculated using the following formula:

$$NS = N_{on} - \alpha * N_{off}$$

where $N_{on}$ is the counts in the source region, $N_{off}$ is the counts in the background regions, and $\alpha$ is a background scale factor. To obtain valid background statistics, it is usually necessary to take multiple observations, which introduces additional scaling factors to consider. These factors include the effective areas, exposure, and region size between the on and off observations. With the counts in the on and off regions and the excess counts, it is possible to estimate the detection significance and the source flux simply and quickly without modeling [Tam20]. A light curve can be constructed by aligning multiple flux estimates, one for each temporal bin.

#### 1.3.2.1 *Li&Ma*

Li&Ma is a statistical method used to calculate the significance of a gamma-ray signal, and it is based on the aperture photometry analysis. It can detect weak signals in background noise and is considered a standard technique in gamma-ray astronomy. The Li&Ma technique estimates the significance level with the likelihood ratio method, using the $N_{on}$ and $N_{off}$ photon counts. The null hypothesis has signal $NS = 0$. It is based on the Poisson statistics to develop the likelihood ratio $\lambda$ calculations. Wilks's theorem [Wil38] provides an analytical expression for the likelihood ratio that is asymptotically exact. The theorem demonstrates that the $-2$ times the natural logarithm of the likelihood ratio $\lambda$ follows a $\chi^2$ distribution when the null hypothesis is true. Finally, the formula to estimate the significance is given by the *equation 17* from [LM83]:

$$S = \sqrt{2}\left\{ N_{on} \ln\left[\frac{1+\alpha}{\alpha}\left(\frac{N_{on}}{N_{on}+N_{off}}\right)\right] + N_{off} \ln\left[(1+\alpha)\left(\frac{N_{off}}{N_{on}+N_{off}}\right)\right]\right\}^{\frac{1}{2}} \quad (1.1)$$

This technique has two main limitations:

1 For Wilks' theorem to hold, the statistical bins of the analysis must be independent.

2 The Li&Ma article [LM83] proves the significance equation can be applied only with a minimum number of counts: $n_{on} \geq 10$ and $n_{off} \geq 10$.

#### 1.3.2.2 *Background estimation*

Ground-based very-high-energy gamma-ray telescopes boast impressive sensitivity, but to fully realize their potential, they must address a major source of systematic error: background subtraction. Estimating background is crucial in analyzing

gamma-ray data using imaging atmospheric Cherenkov techniques. The aperture photometry analysis counts photons in the on-source region and subtracts an estimated background. Various techniques can be employed to estimate the background of an observation, each with its advantages and disadvantages. The choice of which method to use depends on the specific data analysis requirements and the observation characteristics [BFH07].

One approach is the reflected-region background model, shown in Figure 1.12. The source isn't at the center of the field of view but is displaced with an offset relative to the pointing direction of the telescope. The simplest reflection estimation uses a single off-region in the opposite direction relative to the center of the field, with the exact shape of the on-source region. To improve statistics in background measurements, the method can be expanded by utilizing multiple background regions equidistant from the telescope pointing direction. The sum of the event counts $N_{off}$ from these off regions is used to estimate the background of the on-region, scaled by the number of off regions. The reflected-regions background model relies solely on the assumption of radial symmetry of the detector's response. The acceptance function measures a detector's sensitivity to gamma rays as a function of the direction and energy of the incoming gamma ray. Under the assumption of the radial symmetry of the detector's response, the scaling coefficient $\alpha$ is defined as:

$$\alpha = \frac{1}{N_{off}}$$

where $N_{off}$ is the number of off-regions. However, this approach requires a suitable observation strategy. It cannot be applied if the observation positions are within an extended source region or if there are too many other gamma-ray sources in the field of view. In such cases, one might end up in a situation where it is impossible to define a reflected background region without overlap with a known source region, or, in the case of close-by sources just below the detection limit, one might obtain a gamma-ray contaminated background estimate.

### 1.3.3 *Full Field of View Maximum Likelihood*

The likelihood function is a mathematical tool that assesses the agreement between a set of observed data and a given statistical hypothesis or model. Specifically, it quantifies the probability of obtaining the observed data given that the hypothesis is true. The maximum likelihood estimation method can be applied to determine the values of a model's parameters that maximize the likelihood. Cash introduced the likelihood ratio for parameter estimation in high-energy astronomy in the late 1970s [Cas79]. The Full-FoV Maximum Likelihood analysis is a useful technique for detecting gamma-ray sources. It involves defining a background model that

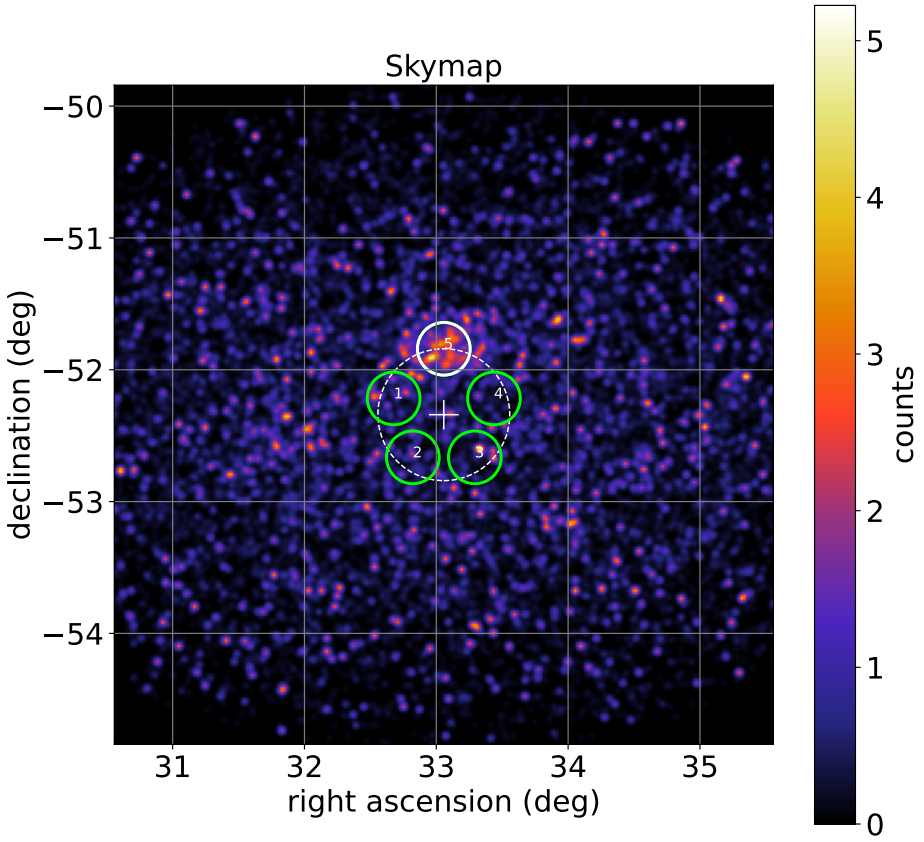


Figure 1.12: Count map of gamma-ray-like events with a reflected region background model.

predicts the expected gamma-ray count rate in the field of view in the absence of sources. This model can be based on the count rate measured in a nearby region or a more sophisticated model that takes into account the telescope's point spread function and other factors. A source model is then defined, describing the gamma-ray emitting region's spectral and spatial characteristics. The spectral model characterizes the energy distribution of the emitted photons, while the spatial model accounts for the distribution of the gamma-ray source in the sky. The point spread function is used as the spatial model for point sources, and it describes how the gamma-ray photons from a point source are spread across the instrument. The background and source models determine the expected gamma-ray count rate from a source [DP+20]. However, this analysis has two primary limitations:

1 Assumptions: it depends on the accuracy of the background and source models, which can restrict its ability to detect sources that do not conform to these models, particularly under degraded conditions in real-time pipelines.

2 Computational cost: the analysis can be computationally expensive, especially for large fields of view or multiple source searches.

## 1.4 GAMMA RAY BURSTS

The source detection method described in this thesis has been applied to gamma-ray burst detection. GRBs are among the most intense and energetic transients known to astronomers. They emit a vast amount of energy in a few seconds, equivalent to that emitted by a star like our Sun in its entire lifetime. Their origins have puzzled scientists since they were first serendipitously detected in the late 1960s by the Vela Satellite Network. Initially, it was believed that GRBs have a galactic origin, but in the early 1990s, it was realized that they are isotropically distributed and have a cosmological origin [MP92], [Mee+92]. Figure 1.13 shows the distribution of GRBs in the sky as Fermi Gamma-ray Space Telescope observed in the first six years of operations. Since the 1990s, GRBs have been the subject of intensive study, particularly after the Burst and Transient Source Experiment (BATSE) on board the Compton Gamma Ray Observatory began operations. The fluence, or energy per unit area, of a single GRB event, ranges from $10^{-7}$ to $10^{-5} erg/cm^2$, and the isotropic energy, or the energy emitted in all directions, ranges from about $10^{48}$ to $10^{55} erg$. This makes GRBs the most energetic event known to humankind [KZ15]. The nature of GRBs is still not fully understood, despite many years of research. The fireball model (represented in Figure 1.14) is one historical model that explains the mechanism of GRBs. It proposes that they are produced by highly relativistic and collimated jets and it explains the prompt emission generation by the interaction of blobs in the jets. In contrast, the interaction of the jet with ambient material produces the multiwavelength afterglow [BG11]. GRBs are traditionally grouped into long (LGRBs) and short (SGRBs), depending on whether the burst lasts more or less than 2 seconds. In 1997, scientists discovered that long GRBs occur in star-forming regions. This led Paczynski to propose that core-collapse events cause LGRBs [Pac98]. Around the same time, MacFadyen and Woosley proposed the collapsar model, which states that a GRB is produced by a jet that emerges from the center of a collapsing star and penetrates the stellar envelope [MW99]. However, it wasn't until the discovery of *SN2003dh* in association with *GRB030329* that the link between GRBs and supernovae was confirmed [Hjo+03]. Since then, several other GRB-SN associations have been discovered, further solidifying the collapsar model as the explanation for LGRBs [BNP+12]. SGRBs are thought to be the result of compact binary mergers with at least one neutron star, such as a

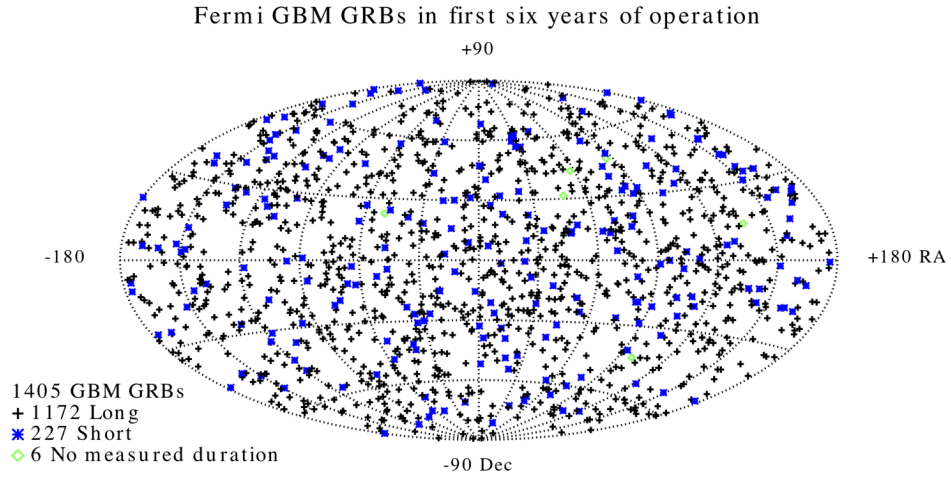


Figure 1.13: The distribution of GRBs in the sky as observed by Fermi Gamma-ray Space Telescope in the first six years of operations shows that they are isotropically distributed and independent of their brightness, duration, spectrum, or any other characteristic. Credits to [Mye23].

black hole and a neutron star (BH-NS), or two neutron stars (NS-NS) [CP95], or two black holes (BH-BH) [PLG16]. For this reason, they are also associated with gravitational wave (GW) counterparts [Abb+17]. The advanced Laser Interferometer Gravitational-Wave Observatory (LIGO) has detected several GWs, including *GW150914*, *GW151226*, *GW170104*, and *GW170817*. The observation of the association of *GW170817* and *GRB170817A* [Abb+17] confirms the NS-NS merger as a progenitor of SGRBs.

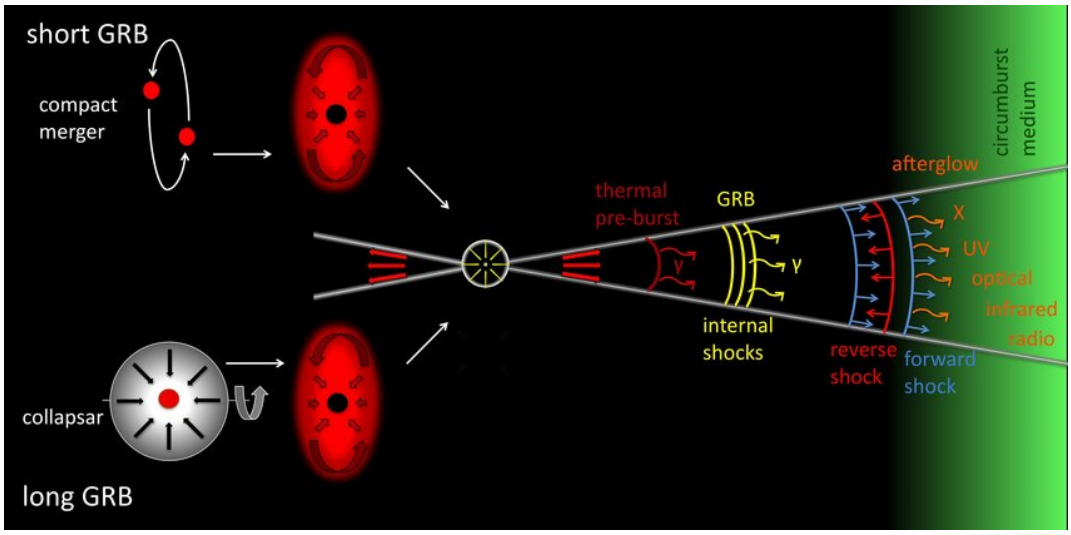


Figure 1.14: Sketch showing on the left the progenitor models for short and long GRBs and on the right the different phases involved in the *fireball* model. The internal shocks produce the gamma-ray prompt emission, and external shock with the interstellar medium, or the star wind is responsible for the afterglow phase observed in radio, optical, X-rays, and gamma-rays. Credits to [Gom12].

## 1.5 SUMMARY

The introductory chapter provided the reader with the required background to understand the context of the work. It discussed gamma-ray astronomy and ground-based gamma-ray imaging with Cherenkov telescopes. The Cherenkov Telescope Array Observatory (CTAO) was introduced, including its science goals, an overview of the telescope array, and its associated computing and software systems, including the Science Alert Generation System, and exploring the scientific use cases of serendipitous discoveries and follow-up observations for real-time detection of transient events. The chapter also covered gamma-ray data analysis techniques, including the full field of view maximum likelihood and aperture photometry and the Li&Ma significance estimation method. Finally, it covered the study of gamma-ray bursts, the transient event the work was focused on.

# 2

# ANOMALY DETECTION FOR TIMESERIES DATA

This chapter introduces the concept of anomaly detection for time series analysis, discussing the major existing contributions to the field. It is organized as follows: Section 2.1 introduces the definition and properties of time series data and the concept of anomaly. It discusses several techniques from the scientific literature, classifying them using a taxonomy. Section 2.2 focuses on anomaly detection techniques based on deep learning. The method developed in this work belongs to this category. Section 2.3 lists several contributions made within the astrophysics domain, proving that these techniques are becoming increasingly popular for analyzing astrophysical data.

## 2.1 ANOMALY DETECTION FOR TIME SERIES ANALYSIS

Anomaly detection involves the identification of patterns or events in data that are unusual or unexpected compared to a baseline or normal behavior [CBK09]. Various factors, such as errors in data collection, rare events, or the influence of external factors, can cause these anomalies. This chapter will discuss anomaly detection in the context of time series data. A time series is generally considered a collection of observations indexed in time order, defined by the following properties:

- Temporality (or temporal correlation): if each data point in the series depends on its past values.
- Dimensionality: the number of individual data attributes captured in each observation. In the case of univariate time series, each observation is defined by one data attribute. In contrast, multiple attributes define the observations that compose a multivariate time series. In the latter case, both the temporal dependence and the correlations between data attributes should be considered during the analysis. Below are the mathematical definitions of univariate and

multivariate time series, assuming that the observations composing a time series have the same temporal granularity.

**Definition 2.1.1.** [Univariate time series] A *univariate time series* $\mathbf{X} = \{\mathbf{x}_t\}_{t \in T}$ is an ordered set of real-valued observations, where each observation is recorded at a specific time $t \in T \subseteq \mathbb{Z}^+$.

**Definition 2.1.2.** [Multivariate time series] A *multivariate time series* $\mathbf{X} = \{\mathbf{x}_t\}_{t \in T}$ is an ordered set of real-valued observations, where each observation is recorded at a specific time $t \in T \subseteq \mathbb{Z}^+$ and consists of k real-valued observations, $\mathbf{x}_t = (x_{1t}, .., x_{kt})$.

Since a time series a can be arbitrarily long, it's often useful to consider subsequences defined as follow:

**Definition 2.1.3** (Subsequence)**.** $\mathbf{S} = \{\mathbf{x}_p, \mathbf{x}_{p+1}, .., \mathbf{x}_{p+n-1}\}$ is a *subsequence* of length $n \leq |T|$ of a multivariate time series $\mathbf{X}$, for $p, t \in T$ and $p \leq |T| - n + 1$.

- Stationarity: a time series is said to be stationary if its statistical properties do not change over time.

  **Definition 2.1.4** (Strongly stationary)**.** For any $\tau \in \mathbb{N}$, a continuous stochastic process $\mathbf{x} = \{x^t\}_{t \in T \subset \mathbb{Z}^+}$ is strongly stationary if following condition is satisfied:

  $$\boldsymbol{F}_{\boldsymbol{x}}(x^{1+\tau}, ..., x^{t+\tau}) = \boldsymbol{F}_{\boldsymbol{x}}(x^1, ..., x^t)$$

  where $\boldsymbol{F}_{\boldsymbol{x}}$ denotes the joint distribution function [Cho+21].

  Many real-world environments experience changes in their underlying statistical data distribution over time, a phenomenon commonly referred to as concept drift [WK94]. This can be a significant issue as it can negatively impact the performance of models trained on historical data [PY10]. For example, *seasonality* is a periodic fluctuation over a limited time scale (e.g., power consumption is high during the day and low during the night, and online sales increase rapidly over the Black Friday weekend and then decrease again). In addition, *change points* are time instants after which the underlying statistical distribution of a stream changes, for example, when the operations of a machine are stopped and restarted with a different setting. As we will see in the following sections, the scientific literature contains anomaly detection techniques designed to work exclusively with stationary or non-stationary time series, as well as techniques developed for the stationary case and then adapted for the non-stationary setting.

- Noise: since it is a common issue in real-world systems, [Tan+18] adds this property to time series data. The noise is defined as any unwanted signal changes during capture, storage, transmission, processing, or conversion. While minor fluctuations in sensors sensitivity can often cause noise and have little impact on the overall data structure, it can make it difficult to distinguish

between noise and actual anomalies in a noisy system. This can greatly impact the performance of detection models [TC02].

Before introducing the taxonomy of the anomaly detection techniques for time series data, the definition of anomaly (or outlier) must be given. Hawkins [Haw80] has provided a widely used definition:

> "An observation which deviates so much from other observations as to arouse suspicions that it was generated by a different mechanism"

Outliers in time series can have different meanings depending on the context [BG+20]. According to Aggarwal [Agg16], they can be seen as noise, erroneous, or unwanted data that are not interesting to the analyst. In these cases, it is best to delete or correct them to improve the data quality. However, in recent years, researchers have increasingly focused on detecting and analyzing unusual but interesting (for a specific domain) phenomena, such as the case of fraud detection. These outliers are often referred to as anomalies. This chapter will refer to this meaning and the term anomaly will be used from now on.

The scientific literature proposes numerous anomaly detection techniques applied to time series analysis [CBK09], [BG+20], [Cho+21], [Gar+21]. In [BG+20] review paper, the anomaly detection techniques are grouped along three axes. The first axis refers to the capability of analyzing univariate or multivariate time series. The second axis refers to the type of anomaly to be detected. Anomalies in time series data can take several forms. A point anomaly is an unusual data point in a specific instant compared to other values in the series (global) or neighboring points (local). These anomalies can occur in one variable (*O1, O2* in Figure 2.1, left panel) or multiple variables (*O1, O2, O3* in Figure 2.1, right panel). Another type of anomaly is called subsequence, which refers to consecutive points in time that exhibit unusual behavior together, even if each point is not necessarily an anomaly. These can also be global or local and affect one variable (*O1, O2* in Figure 2.2, left panel and *O3* in Figure 2.2, right panel) or multiple variables (*O1* in Figure 2.2, right panel). Finally, the entire time series can be considered anomalous when multiple variables are involved (*Variable 4* in Figure 2.3). Although these definitions of anomalies are general and usable in different real-life scenarios, the classification of what is considered abnormal varies based on one's perspective of what is considered normal. Hence they may be further subdivided into more specific groups [Cho+21], [Tan+18].

The third dimension of the taxonomy examines the type of detection method used. A univariate detection method only examines one variable over time, while a multivariate method can analyze multiple variables simultaneously. It's worth noting that even though the input data may be a multivariate time series, a univariate detection method can still be employed by analyzing each variable individually, disregarding any potential dependencies between them. However, a multivariate

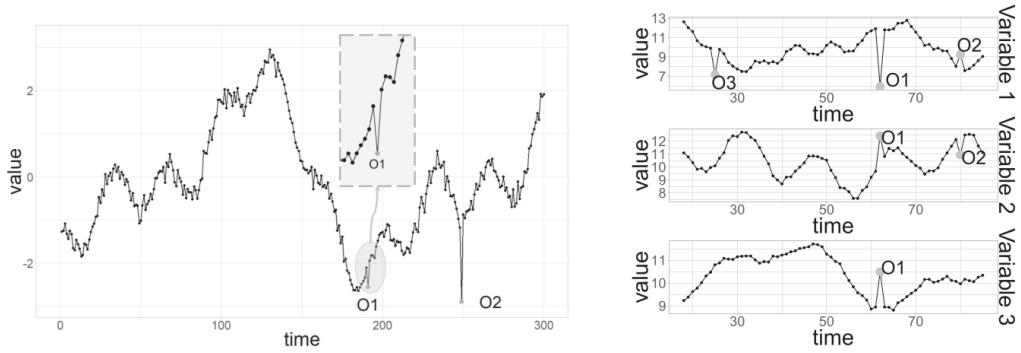


Figure 2.1: Point anomalies in time series data. Left panel: univariate time series. Right panel: multivariate time series. Credits to [BG+20].

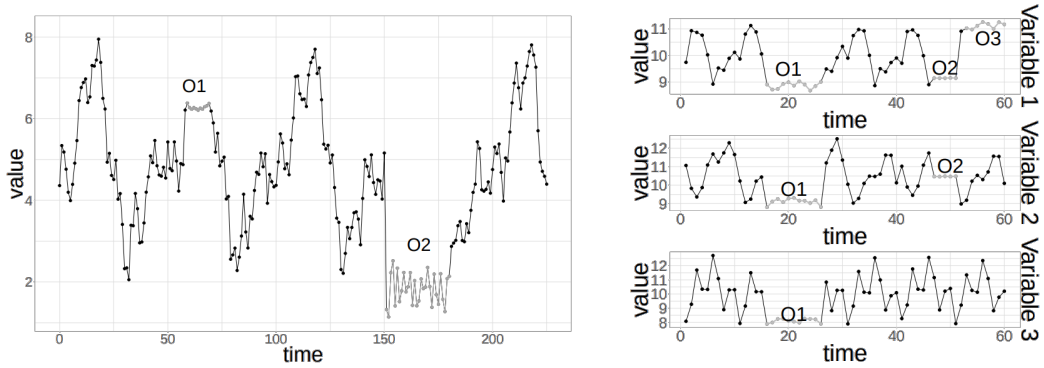


Figure 2.2: Subsequence anomalies in time series data. Left panel: univariate time series. Right panel: multivariate time series. Credits to [BG+20].

technique cannot be utilized with univariate time series data. Therefore, this dimension only applies to multivariate time series data.

The three dimensions described above are the highest-level categories of the proposed taxonomy. The authors of [BG+20] further develop the proposed taxonomy tree. In contrast, in [Cho+21], the anomaly detection techniques are divided into two main groups: *traditional approaches* and *deep learning based*. The following sections will continue to describe the taxonomy proposed in [BG+20] and then focus on deep learning-based methods mentioned in [Cho+21].

### 2.1.1 *Anomaly detection techniques for point anomalies in the univariate domain*

The set of anomaly detection techniques that can detect point anomalies is further divided by [BG+20], taking into account the following aspects:

- The technique exploits the **temporality** of the data: some techniques take into account the temporal ordering of observations, and others completely disregard this information. The latter will produce the same results even when the observations are shuffled.

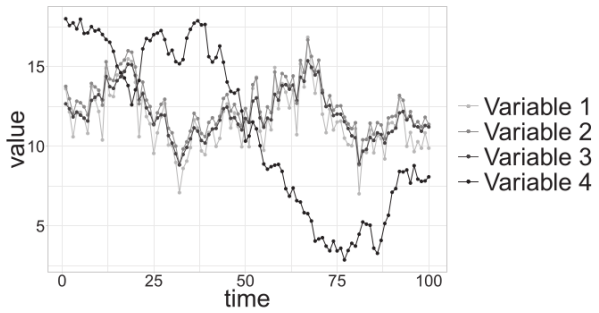


Figure 2.3: Anomalous time series (*Variable 4*) in a multivariate time series. Credits to [BG+20].

- The technique can be applied in a **streaming** context: some techniques can identify outliers in real-time, as soon as new data points arrive, without needing to wait for further data. Among these methods, some maintain a constant model throughout the stream. In contrast, others adapt and update their detection models with new information, either by completely retraining the model or through incremental learning. Techniques that cannot make instant decisions on new data points are considered unsuitable for streaming time series analysis.
- The **nature** of the technique: *model-based* techniques rely on fitting a model; *density-based* techniques use the concept of neighborhood; *histogramming* techniques compute a histogram representation of the data.

The next sections will describe further the last point of the previous listing.

#### 2.1.1.1 *Model-based techniques*

The model-based techniques are the most commonly used approach in literature and rely on fitting a model to estimate the expected value of a data point [BG+20]. The data point is considered an anomaly if:

**Definition 2.1.5.** $|x_t - \hat{x}_t| > \tau$, where $x_t$ is the observed data point and $\hat{x}_t$ is its expected value and $\tau$ is a threshold.

The expected value, $\hat{x}_t$, represents the typical or normal value of a data point in the time series, and the threshold $\tau$ represents the level of deviation from this normal value considered abnormal or significant. As we will see later in this chapter, [Cho+21] generalizes this concept with the definition of the *anomaly score*, a numerical value that indicates the likelihood of a data sample being anomalous that replaces the term $|x_t - \hat{x}_t|$ in 2.1.5. Despite calculating the expected value $\hat{x}_t$ or the anomaly score and threshold $\tau$ differently, these techniques involve explicitly or implicitly fitting a model. Model-based techniques are divided into **estimation** (shown in Figure 2.4, left panel) and **prediction** techniques (shown in Figure 2.4,

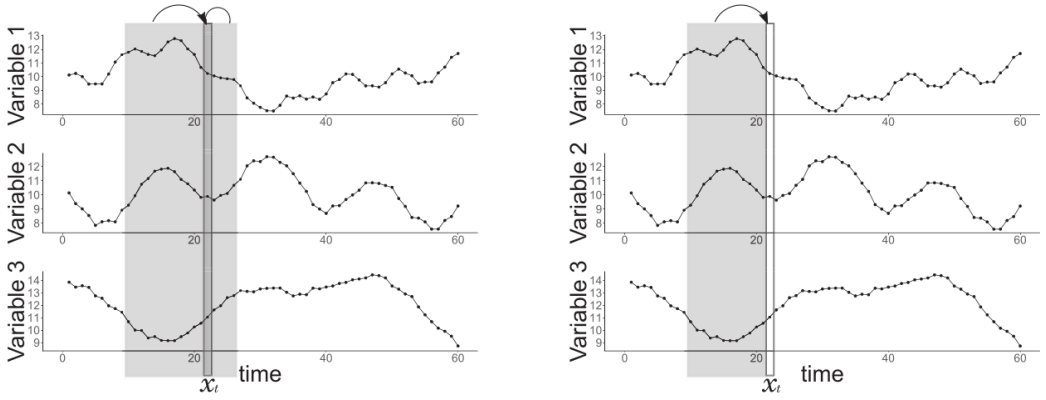


Figure 2.4: Example of estimation (left panel) and prediction (right panel) models-based approach for a multivariate time series. Consider only one variable for the univariate scenario. Credits to [BG+20].

right panel). Estimation techniques use $\{x_{t-k_1}, ..., x_t, ..., x_{t+k_2}\}$ to compute $\hat{x}_t$, while in prediction techniques, $\hat{x}_t$ is computed using only previous observations. The main practical difference between the two is that prediction techniques can be applied in the streaming scenario as they can immediately identify whether a new data point is an outlier as soon as it arrives. In contrast, estimation techniques can only do so if only the current point $x_t$ is used to compute the expected value along with some preceding points.

#### 2.1.1.2 *Density-based techniques*

The basic idea behind this method is that normal data points will be closely packed together, forming dense clusters. Anomalous data points, on the other hand, will be located in areas of the data where there are few other data points, resulting in a lower density of data in that region. The *density-based* approach relies on computing the density of data points in different regions of the data and identifying any regions with a lower density than the surrounding areas, flagging them as potential anomalies. This approach can be applied to data of any dimensionality and does not rely on prior knowledge of the normal behavior of the data [BG+20].

#### 2.1.1.3 *Histogramming techniques*

The method in question is centered around identifying points in a time series that, when removed, result in a histogram representation with less error than the original, even when the number of buckets is decreased to allow for the storage of these points separately (Figure 2.5). This approach aims to detect the points in the time series that deviate from the expected behavior, and removing them allows for a more accurate representation of the data [BG+20].

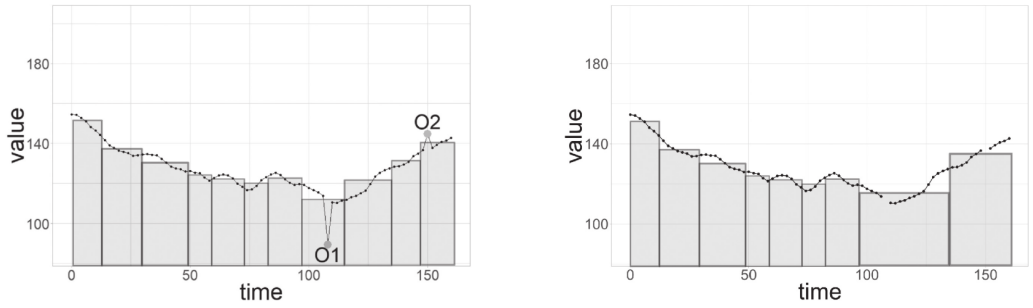


Figure 2.5: Example of the histogramming technique for univariate time series. The points $\{O1, O2\}$ are anomalous. Credits to [BG+20].

#### 2.1.1.4 *Parametric vs non-parametric techniques*

In [BG+20], the authors further develop the taxonomy tree considering the parametric and non-parametric nature of the techniques. Parametric techniques assume that the underlying distribution of the data belongs to a specific family of distributions and estimate a fixed number of parameters from the data. On the other hand, non-parametric methods do not make any assumptions about the underlying distribution and instead determine the model or family of distributions from the data with a flexible number of parameters. Additionally, semi-parametric methods combine elements of both parametric and non-parametric approaches.

### 2.1.2 *Anomaly detection techniques for point anomalies in the multivariate domain*

Multiple variables may be correlated when the input data comprises multivariate time series. Unlike the case of univariate time series, the detection method used for identifying point outliers in multivariate time series should consider multiple variables at once. Furthermore, an outlier in a multivariate time series can impact one or multiple variables simultaneously. The set of anomaly detection techniques that can detect point anomalies in the multivariate domain is further divided by [BG+20] into two groups: univariate and multivariate analyses: univariate analyses do not consider dependencies that may exist between the variables, while multivariate analyses can explore any correlation between the variables to find casual relationships. Regarding the first case, the techniques described in Section 2.1.1 can be applied. To overcome the loss of information that occurs when the correlation dependencies between the variables are not considered, dimensionality reduction steps, such as PCA, can be used to find a new set of uncorrelated variables where univariate techniques can be applied. Model-based techniques (prediction and estimation models) and histogramming techniques can be extended to conduct multivariate analysis. Another model-based technique not cited by [BG+20] but used in the literature (see Section 2.3) is *iForest*, from [LTZ08], which is fundamentally

different from existing approaches as it explicitly isolates anomalies instead of constructing a profile of normal instances. *iForest* uses sub-sampling and has a linear time complexity with low constant and memory requirements. It works well, especially in large data sets and high-dimensional problems with many irrelevant attributes.

#### 2.1.2.1 *Dissimilarity techniques*

The category of *dissimilarity-based* techniques is introduced by [BG+20]. These methods involve calculating the difference between multiple points or their representations without a model to be fitted. Given a threshold value, if the dissimilarity between a point and its expected value exceeds that threshold, the point is considered an outlier.

**Definition 2.1.6.** $s(x_t, \hat{x}_t) > \tau$, where $x_t$ is the k-dimensional data point, $\hat{x}_t$ is its expected value and $s$ measures the dissimilarity between two multivariate points.

These techniques typically do not use raw data but instead employ different representation methods, such as graphs or vectors, and the definition of the dissimilarity function $s$ changes accordingly.

### 2.1.3 *Anomaly detection techniques for subsequence anomalies in the univariate domain*

Subsequence anomalies are the second type of anomaly defined by [CBK09]. The objective is to spot a sequence of consecutive points that deviate from normal behavior. In this case, we need to take into account additional aspects.

- Unlike point anomalies, subsequences consist of multiple points, introducing a new analysis constraint, i.e., the capability to analyze subsequences of varying lengths simultaneously or rely on fixed-length subsequences. In the latter case, a sliding window over the time series can be used to obtain them. It's also worth noting that the number of subsequences the method will consider and analyze is dependent on the chosen length (i.e., the shorter the length, the higher the number of subsequences).
- Another aspect that subsequence anomaly detection methods must consider is the representation of the data. Since comparing subsequences is more challenging and costly than comparing individual points, many techniques use a representation of the subsequences instead of the raw values, such as the *discretization* method, as shown in [CBK12].
- In addition, the subsequence anomalies can be periodic, making their detection more challenging. Periodic subsequence anomalies are unusual sequences that repeat over time.

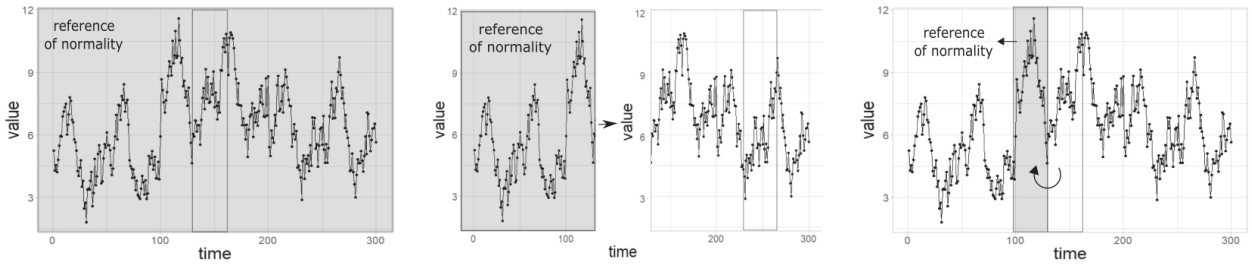


Figure 2.6: Example of different references of normality used by dissimilarity-based approaches. The panel on the left shows the same time series. The panel in the middle shows an external time series. The panel on the right shows a previous subsequence. Credits to [BG+20].

- Furthermore, instead of point anomaly detection, where some methods do not consider temporal dependencies, subsequences inherently consider temporality.
- Finally, when analyzing subsequence anomalies in a streaming context, several approaches can be considered: a single data point arrives, and a classification is made for a subsequence containing this new point; a subsequence arrives, and it's classified; a batch of data arrives, and a subsequence anomaly is found within it.

In this scenario, the anomaly detection techniques belong to the following groups: *discord detection* techniques compare each subsequence to all others using the Euclidean distance and require the user to specify the subsequence length. These methods are limited because they cannot specify whether the identified subsequences are anomalies. The *dissimilarity-based* techniques in the multivariate scenario use a reference of normality to decide whether or not a subsequence is an anomaly based on their direct comparison. This can be defined as:

**Definition 2.1.7.** $s(S, \hat{S}) > \tau$, where S is the subsequence being analyzed or its representation, $\hat{S}$ is the expected value of $S$ obtained based on the reference of normality, and $s$ measures the dissimilarity between two subsequences.

Figure 2.6 shows some examples of different references of normality ($\hat{S}$ in 2.1.7). The subsequence under analysis ($S$ in 2.1.7) and the references are then clustered, and the centroids or center of the cluster to which the $S$ belongs are computed to identify anomalies (as shown in Figure 2.7). The dissimilarity-based techniques in the multivariate scenario differ from those introduced in Section 2.1.2.1 because of the representations used to describe subsequences (graphs and vectors against clustered representations). The *prediction model-based* techniques in the multivariate scenario build a regression model using past data to predict up to n points into the future. Then, anomalies are detected by calculating the distance from the predicted subsequences to the actual ones. If the distance exceeds a threshold $\tau$ the

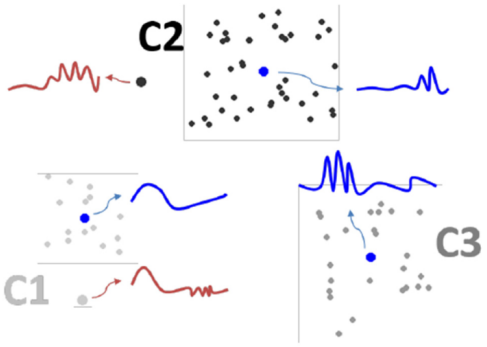


Figure 2.7: Clustering of the subsequences in a univariate time series. Cluster centroids are highlighted, and C1 and C2 clusters contain subsequence outliers. Credits to [BG+20].

subsequence is classified as an anomaly. The *frequency-based* techniques compare subsequences to a reference to normality, as done by Figure 2.6. A subsequence $S$ is an outlier if its frequency is lower than the expected one:

**Definition 2.1.8.** $|f(S) - \hat{f}(S)| > \tau$, where $f(S)$ is the frequency of occurrence of $S$, $\hat{f}(S)$ its expected frequency, and $\tau$ a predefined threshold.

Finally, the last group of multivariate subsequence anomaly detection techniques is based on *information theory*, similar to the frequency-based methods. The main idea is to find patterns in data that happen less often but still happen regularly with respect to a reference to normality. Patterns that happen less often are more surprising and carry more information. This can be achieved by looking at how often the symbols in a pattern appeared in the series and how often the pattern appeared overall.

### 2.1.4 *Anomaly detection techniques for subsequence anomalies in the multivariate domain*

The techniques mentioned by [BG+20] in this scenario share similar concepts with the technique to detect subsequence anomalies in the univariate domain, as they are simply an advanced version of simpler techniques that have already been discussed earlier. In particular, *model-based* techniques, both prediction and estimation models, performs the best in this scenario, especially if based on deep learning. The following section will focus on this group of techniques.

## 2.2 ANOMALY DETECTION WITH DEEP LEARNING

To understand why deep learning in unsupervised settings has a critical role in the anomaly detection context, we have two consider two main challenges, described

in [Cho+21]. The first is the rarity of the anomalies. This scarcity makes it time-consuming and resource-intensive to gather enough data for supervised data sets. Additionally, when labeled data is obtained, the imbalance between normal and abnormal data can negatively impact the training of models. In addition, for most real-life use cases, it is not guaranteed to be able to generate a supervised data set fully representative of the anomalous class. Another challenge regards the complexity of the data. Univariate time-series analysis is still relevant in applications that require minimal computation, such as edge computing. However, as automation and control systems become more complex, it becomes impractical to monitor individual univariate data streams separately. With many dimensions, traditional approaches often experience a decline in performance due to the *curse of dimensionality*. But, as already discussed, correlations between variables that cannot be inferred through univariate time-series analysis can also indicate anomalies.

### 2.2.1 *Deep learning architectures for anomaly detection in multivariate time series*

The past behavior of a sequence holds valuable information that can indicate potential changes in the future, and deep learning architecture can natively model the temporal context of the data. A popular method is using a Recurrent Neural Networks (RNN) model and its variants, such as Long Short-Term Memory (LSTM) [HS97] and Gated Recurrent Unit (GRU [Chu+14]. Those architectures address the vanishing or exploding gradient problem, which occurs when the gradient becomes too small or too large as the network becomes deeper. Through several *gates*, they can learn long-term dependencies by deciding which previous states keep or discard at each time step. Another approach is based on a dilated RNN model [Cha+17], which extracts multi-scale features and models long-term dependencies using a skip connection between hidden states. While Recurrent Neural Networks models are commonly used for analyzing time series data, some studies have found that Convolutional Neural Networks (CNNs) models can perform better in certain scenarios that involve short-term data [Cho+21]. CNNs utilize multiple layers of convolutions, which allow them to learn increasingly complex features as they progress through the layers. Additionally, pooling layers introduce non-linearity to the CNNs, allowing them to capture complex patterns [AMAZ17]. However, a drawback of using CNNs is that it can be challenging to understand patterns that occur over a prolonged period. To address this issue, Temporal Convolutional Networks (TCN) have been proposed by [Lea+16]. TCN has three distinct characteristics; it uses causal convolutions, meaning that future information is not considered when analyzing past data. It can handle input sequences of any length, similar to RNNs. Furthermore, it can look far into the past using deep networks and dilated convolutions to make predictions. A hybrid architecture, called Con-

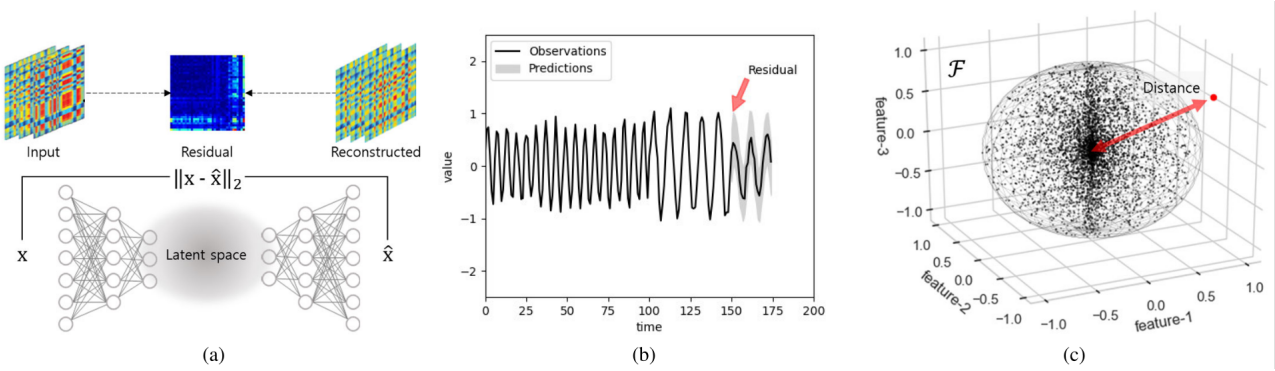


Figure 2.8: Examples of the anomaly criteria: (a) a reconstruction error; (b) a prediction error; and (c) a dissimilarity. Credits to [Cho+21].

vLSTM, has been introduced by [Shi+15] to address the spatiotemporal sequence-forecasting problem, i.e., when there's the need to consider the spatial information and temporal dependencies simultaneously. The spatial information is introduced when a multivariate time series is represented as a 2D covariance matrix. Those matrices are stacked when the time series is monitored with a sliding window, as explained in [Cho+21]. Recent architectures based on the attention layers [Vas+17], such as Transformers and Bidirectional Encoder Representations from Transformer (BERT) [Dev+18], have been widely used in the field of natural language processing and have recently been applied to the time-series anomaly detection domain due to their ability to handle long-range dependencies effectively. Hierarchical Temporal Memory (HTM) [Haw+16] is a relatively new approach to deep learning architecture and is still being actively researched. HTM has a biologically-inspired design, based on the neocortex's structure and function, which is the part of the brain responsible for sensory perception, cognition, and decision-making. HTM comprises a hierarchy of layers containing cells organized into columns. The cells in each column are jointly connected and connected to the cells in adjacent columns. The input and the previous states of the connected cells activate the columns. This allows the algorithm to learn and recognize spatial and temporal patterns, making it useful for various applications, including anomaly detection. In addition, one of the key advantages of HTM is its ability to learn and adapt to changing data patterns, making it useful for real-time applications.

#### 2.2.1.1 *Anomaly criteria*

The anomaly detection techniques based on deep learning are subdivided into three main categories by [Cho+21], with respect to the criteria they use to define anomalies. They also belong to the *model-based* category introduced by [BG+20]. As outlined in Section 2.1.1.1, a model-based technique relies on fitting a model. A loss function must be defined and minimized in this case since *backpropaga-*

*tion* is used as the learning algorithm. Once the model is trained and the data representation is learned, it is applied to production data to perform inference. The model's output is used to compute an *anomaly score*, i.e., a numerical value that indicates the likelihood of a data sample being anomalous. If the indicator it's greater than a threshold $\tau$, the sample is classified as anomalous. The strategies to compute the anomaly score can be divided into three types: the first is called *recostruction error*, and it's a generalization of the estimation-based method described in Section 2.1.1.1. This anomaly score is commonly used by AE, VAE, GAN, and Transformers. They reconstruct or generate data analogous to the input data and compute the residual between the input and generated data, as shown in Figure 2.8, panel (a). Prediction techniques compute $\hat{x_t}$ using only previous observations. Obtaining anomaly scores involves assigning a binary label according to the chance of the data point being deemed normal, as outlined in [116], [119]. The discrepancy between the predicted and actual labels demonstrates the prediction error. This method can not always be applied since labels are insufficient in many real-world scenarios. Another method to obtain anomaly scores from prediction models is very similar to 2.1.6: the model forecasts the predicted value for future time steps and the anomaly score is computed as the difference between the expected value and the actual data. The latter strategy is shown in Figure 2.8, panel (b). Finally, the third category contains *dissimilarity-based* methods that extract features from the input data, create clusters, and assess the distance between from the cluster of previous data. This dissimilarity-based method evaluates the similarity through different measures, including the Euclidean distance, Minkowski distance, cosine similarity, and Mahalanobis distance. [Cho+21] contains several references to techniques that implement the abovementioned anomaly criteria.

## 2.3 ANOMALY DETECTION IN ASTROPHYSICS

As automation and technological advancements spread across industries, many systems produce vast amounts of high-dimensional data. As shown by [CBK09], [BG+20], [Cho+21], and [Gar+21] review papers, anomaly detection techniques are being used in a wide range of contexts, such as identifying fraud in financial transactions, detecting defects in manufacturing processes, identifying cyber attacks in network security, and detecting diseases in medical imaging. These techniques are particularly useful in high-dimensional data sets where manual inspection is infeasible and when labeled data is unavailable. However, most implementations are highly specific to the individual use case and thus require domain knowledge for appropriate deployment [Cho+21].

The astrophysics domain has the same characteristics: a massive amount of high-dimensional data is being produced, and generating fully-representative super-

vised data sets is not always feasible due to limited computational resources but also the difficulties in considering all possible sources of systematics, including glitches [Sad20], and all possible anomalies, that may be unknown.

Several anomaly detection techniques are used in different astrophysics domains, from traditional methods such as PCA or Isolation Forest to deep learning-based methods. In [Pru+19], the Open Supernova Catalog (OSC) photometric data is analyzed with dimensionality reduction, and anomalies are detected with the isolation forest algorithm. The latter technique is also the backbone of *Astronomaly* [Web+20], [LB21], a general anomaly detection framework with a novel active learning approach designed to provide personalized recommendations for astronomical data, including images, light curves, and spectra. An anomaly detection algorithm based on an unsupervised Random Forest is proposed by [BP17], which is tested on more than two million galaxy spectra from the Sloan Digital Sky Survey. Furthermore, techniques based on clustering are proposed in the literature [VBK19], [GW19]. In [Doo+21], the performance of six unsupervised outlier detection methods, including the Local Outlier Factor, Isolation Forest, k-means clustering, and a convolutional autoencoder, are being compared for the analysis of images from the Sloan Digital Sky Survey. Deep learning-based techniques are also being used: [IY19] proposes an anomaly detection technique based on a variational auto-encoder for high-resolution X-ray spectroscopy, [Mar+20] explores the use of deep generative networks for detecting outliers in astronomical imaging data sets, [ZZ18] and [Sad20] use Long Short-term Memory (LSTM). In particular, [Sad20] proposes a new data-driven discovery framework developed to detect and characterize explosive astrophysical transients using multiple messengers such as neutrinos, optical supernovae, and $\gamma$-rays. Within the gravitational waves domain, [MPP21] uses semi-supervised outlier detection algorithms to search for unmodelled gravitational wave (GW) signals. Finally, [Par+23] represents a novel application of anomaly detection in the gamma-ray analysis domain. It employs an anomaly detection technique to detect gamma-ray bursts (GRBs). The model is based on a CNN autoencoder architecture, trained in an unsupervised manner on multivariate time series data extracted from the AGILE space satellite anticoincidence system.

## 2.4 SUMMARY

In this chapter, we introduced the concept of anomaly detection for time series analysis and discussed existing contributions in the field. The definition and properties of time series data and the concept of anomaly were introduced, followed by a classification of existing techniques using a taxonomy. The focus shifted to anomaly detection techniques based on deep learning, the category to which the

proposed method belongs. Lastly, several contributions made in the astrophysics field were listed, showing the growing popularity of these techniques for analyzing astrophysical data.

# 3

# ANOMALY DETECTION METHOD TO PERFORM SOURCE DETECTION

This chapter presents the proposed method to address the real-time source detection problem introduced in Section 1.2.4. It is organized as follows: Section 3.1 describes the proposed anomaly detection technique. It presents the data pipeline that has been developed to generate input data, the deep learning architectures that have been investigated, and the training process. The evaluation of the models will be addressed in Section 4.1. Section 3.2 will describe the p-value analysis to associate each positive classification with a Gaussian statistical significance. Section 3.3 and Section 3.4 explain how the overall system works, from configuration to inference. Section 3.5 investigates several problems that can arise during the observatory's operations and how those problems can affect the proposed system.

## 3.1 THE PROPOSED ANOMALY DETECTION METHOD

This section will describe the proposed anomaly detection technique. As explained in Section 2.1, the goal of anomaly detection is the identification of rare items, events, or observations which deviate significantly from the majority of the data and do not conform to a well-defined notion of normal behavior. This work applies anomaly detection analysis to light curves in the gamma-ray energy spectrum. The y-axis represents the *flux* quantity, describing how much light a source emits per unit of time and surface. The goal is to identify high-energy phenomena called *Gamma-Ray Bursts* (GRBs), described in Section 1.4. This problem is called *source detection*. It will be framed into the two use cases of *serendipitous discoveries* (Section 1.2.5.1) and *follow-up observations* (Section 1.2.5.2) that require real-time data analysis. The problem can be addressed using an anomaly detection approach considering the *normal data* as the signal coming from a sky region where no sources are present except the background. In contrast, *anomalous data* is the signal from a

sky region where an astrophysical source is present (along with the background). The proposed method belongs to the model-based family of techniques to detect multivariate sub-sequence anomalies described in Section 2.1.4. It is based on an autoencoder that learns the time series behavior of normal samples. It thereafter uses reconstruction error as anomaly criteria (Section 2.2.1.1) to detect anomalies. Consider a time series $\boldsymbol{X} = \{\boldsymbol{x_1}, \boldsymbol{x_2}, ..., \boldsymbol{x_L}\}$ of length $L$, where each point $\boldsymbol{x_i} \in \mathbb{R}^3$ is a 3-dimensional vector of flux values. As we will see in Section 3.1.1, the vector of flux values is composed of logarithmically spaced energy bins between 0.04 and 1 TeV. We will consider the scenario where multiple time series are obtained by taking a window of length $L$ over a larger time series. The value of L determines how much signal the autoencoder will use to perform the reconstruction, and it has been set to 5. The autoencoder will be trained on normal samples to minimize the reconstruction error of the decoding step. During the training, the network will learn the model of the background. Once fitted, the model encodes and decodes data samples and outputs an *anomaly score*. GRBs are then detected as noteworthy deviations from this indicator with respect to the anomaly score of the normal samples. The anomaly score is computed with a weighted mean-squared error to give more importance to prediction errors in the lower energy ranges that contain most of the signal. This is intrinsically due to the source's spectrum, which emits more in the low range of energies. Finally, a sample is classified as an anomaly if the anomaly score is greater than a certain threshold $\tau$. To determine the significance of a GRB detection, the anomaly score is mapped to a p-value through a statistical analysis described in Section 3.2.

### 3.1.1 *The data pipeline*

This section will describe the input data and the data pipeline that produces it. As described in Section 1.3.2, the flux of the electromagnetic radiation emitted by a gamma-ray source can be computed starting from the counts in the on and off regions and the excess counts. The flux measurements can derive a source's spectrum and light curve. The spectrum of a gamma-ray source is a plot of the flux as a function of energy. By measuring the spectrum, astronomers can determine the distribution of energies of the photons emitted by the source and learn about the physical processes responsible for their production. On the other hand, a light curve is a plot of the flux as a function of time. It provides information about how the intensity of the radiation changes over time. The data pipeline developed in this work generates multivariate time series representing light curves for three different energy ranges. The final step of the data pipeline extracts multiple subsequences by sliding a window over the light curves. A semi-supervised data set of background-only samples is generated at the end of the data pipeline. In partic-

ular, several multivariate sub-sequences over three channels are extracted for each photon list. Each sub-sequence point is a flux measurement aggregating one or more seconds of the original photon list. Figure 3.7 shows some generated samples (in blue). The following sections will explain in detail the components of the data pipeline.

#### 3.1.1.1 *The photons list simulator module*

The data that is needed for the photometry analysis is produced with simulations. This module aims to simulate the data acquisition of a particular sub-array configuration of CTA, considering its instrument response function (IRF), as described in Section 1.3.1. The output of a simulation is a photon list. A photon list contains the high-energy photons detected by the telescope sub-array. It records the detection timestamp for each photon, their reconstructed energy, and the direction of arrival with their corresponding errors. Monte Carlo simulation techniques are commonly used to generate photon lists, allowing for incorporating various physical processes. The simulator described in [DP22] has been adopted. This Python package adopts the *ctools* open-source software used for the scientific analysis of data from imaging atmospheric Cherenkov telescopes (IACTs), developed in the framework of CTA [Kno+16]. It has been validated on simulated and real data from H.E.S.S., and Fermi [Kno+19]. Simulations performed with [DP22] can be configured using the following parameters:

- number of simulations (trials): this parameter determines the number of statistically independent realizations under the same conditions. The only difference between the trials is the seed, making the trials statistically independent.
- simulation duration (tobs): this parameter determines the duration of the observation in seconds.
- minimum energy, maximum energy, and region-of-interest's size (emin, emax, roi): these parameters constrain the output of the simulation. The gamma-ray photon will have energy in the range $[EMIN, EMAX]$ (expressed in TeV), and their direction of arrival will be inside the region of interest (ROI), a circular region inside the field-of-view, whose radius is expressed in degrees.
- simulation type (simtype): this parameter is used to decide which models the simulation will consider: only the background model is used in a background-only simulation. In a GRB simulation, only the GRB models are considered (check the next bullet point). In addition, the results of background and GRB simulations can be merged to generate a photon list containing both.
- GRB template (runid): this parameter is needed to simulate a GRB event. As outlined in Section 1.4, the GRBs events are different in duration and lumi-

nosity, and the template is a model of the evolutionary process of the event. It is a 3-dimensional grid of fluxes, 70 temporal, and 40 energy bins. Hence, the template defines 70 light curves for each energy bin, and for each light curve, it defines 40 spectra. The simulation will integrate the light curve and the spectra within a time-energy interval in the 2D space of a sky map considering the XML spatial model (RA, DEC) of the point-like source. The templates used in this work are taken from the POSyTIVE catalog [Ber+19]. The mock GRB population used by POSyTIVE is calibrated using a 40-year data set of multi-wavelength GRB observations.

- GRB start (onset): the delay in seconds between the start of the observation and the start of the GRB event.

- GRB displacement from on-axis (offset): this parameter is the offset (in degrees) added to the on-axis pointing that represents the position of the simulated source. The value of this parameter is considered fixed and equal to 0.5° for all simulations.

- Instrument response function (IRF): different types of telescopes and array configurations have different IRFs. Only one IRF has been used to run all simulations, i.e., $North\_z40\_5h\_LST$, which models the response of a sub-array composed of 4 LST-1 telescopes located in the northern hemisphere site, observing for 5 hours at 40° zenith angle. This IRF has been chosen to model extra-galactic observations. Simulated galactic plane observations would require both the diffuse emission and known source models that are not available at the time of writing. Only one IRF has been considered because the GRBs catalog doesn't contain the *trigger times*, i.e., the timestamp associated with the start of the GRB events. Without this temporal reference is not possible to infer the position of the GRBs. Hence, they are all simulated in the same sky region. A single IRF is sufficient because only one background level is considered, observing at 40° zenith angle, an intermediate case between the zenith and the horizon.

- Calibration database (caldb): this database contains CTA's instrument response functions (IRFs), described in Section 1.3.1. The calibration database used in this work is tagged by version *prod5 v0.1* [OC21].

The extra-galactic background light (EBL) must be considered to simulate GRB events. This is the integrated intensity of all the light emitted throughout the universe's history across the electromagnetic spectrum [Coo16]. If the involved photon energies are above the threshold for electron-positron pair creation, very high energy gamma-rays ($E > 100GeV$) are absorbed from the EBL through interaction with low-energy photons [Maz+13]. The simulator [DP22] accounts for the EBL

absorption, extracting the power law spectral models from the templates and modifying them by applying an exponential cutoff with a predefined absorption level [Gil+12]. Finally, it generates new light curves considering the new absorbed spectral models. The output of the simulations is a set of photon lists in Flexible Image Transport System (FITS) format, the standard data format used in astronomy. It is used for transporting, analyzing, and archival storing scientific data sets, supporting multi-dimensional arrays. A FITS file contains one or more tables with rows and columns of information and a header containing metadata [Cen21]. A photon list contains all the detected photons, described by the detection timestamp, their reconstructed energy, and the direction of arrival. It is important to highlight that although the photon list generation is the most resources intensive process in this work, it can be executed only once for each IRF. Several photometry analyses with different integration times can be performed from this data set.

#### 3.1.1.2 *Photometry module*

The photometry module takes as input the photon lists generated by the simulator, along with several other parameters. It integrates the gamma-ray photons along three dimensions: space, time, and energy. It generates a multivariate time series of flux values for each photon list given as input. This software module has been built on the work of [Tam20], wrapping the region counting routine and the effective area computation. The rest of this software module has been developed to optimize the computations' speed as much as possible. The public API is represented by the *OnlinePhotometry* class, which accepts the parameters used in the simulation process and the configuration parameters described as follows: the *integration time* (itime) defines the size of the temporal bins used for the time integration of the photon list. For example, with 500 seconds of observation time and an integration time of 5 seconds, we obtain $500/5 = 100$ points representing a time series of photon counts. The *number of energy bins* is used to compute the energy bins, logarithmically spaced between $tmin = 0.04TeV$ and $tmax = 1TeV$. Considering more than one energy bin, the integration process will output a multivariate time series of photon counts, dividing the photon counts in each energy bin. In the latter example, the output data shape would equal $(100, 3)$. Finally, the *sub-window size* (*sws*) parameter truncates the time series to have *sws* points. In the latter example, the output shape would be $(0 : sws, 3)$. After defining the temporal end energy integration parameters, the spatial dimension must also be defined. As outlined in Section 1.3.2, the spatial region is a circular sky region defined by a center in sky coordinates and a radius expressed in degrees. Only the photons whose arrival direction falls into the circular region are counted. The circle's center is defined by adding an offset in degrees from the pointing. The pointing is read from the header of the FITS file. At the same time, the center of the region

is computed automatically with a predefined offset added to the on-axis pointing direction. This is accomplished by providing the parameters *regions_radius* and *max_offset* to the *integrate* method. To avoid underestimating or overestimating the photon counts, the value of the *regions_radius* should be equal to the instrument's point spread function (PSF). Still, since the PSF size depends on the energy range, three different region sizes would have been considered. To avoid this complexity, we assumed a fixed value of 0.2°, also because the core of the photometry tool [Tam20] corrects the photon count by a scaling factor proportional to the PSF size. The last step is to transform the time series of photon counts into flux values. The flux ($\Phi$) in $ph\,cm^{-2}\,s^{-1}$ is defined by

$$\Phi(E) = \frac{dF}{dE}(E) = \frac{dN_\gamma}{dE\ dA_{eff}\ dt_{eff}}$$

where $dN_\gamma$ is the number of excess events in $dE$ energy, $A_{eff}$ is the *effective area* in the chosen source region and $t_{eff}$ is the effective observation time. The effective area $A_{eff}(\theta, E_\gamma)$ is the geometric area where photons are collected, multiplied by an efficiency term that depends on the energy of the incoming photons and its angular distance $\theta$ from the optimal on-axis pointing direction. The CTA observatory has a higher effective area in the high-energy range but degrades slowly with respect to the pointing-source angle [Tam20]. Since the $\Phi(E)$ formula normalizes the photon counts for the effective area, taking into account the degradation of the IRF, which increases while moving away from the telescope's pointing, it's possible to extract the photon counts from multiple regions with different offsets from the optimal on-axis pointing direction. Extracting flux measurements from multiple regions in the sky simultaneously significantly boosts the data generation rate and the field-of-view coverage. The *OnlinePhotometry* class uses the *RegionsConfig* class, which responsibilities are the computation of the regions' position and their effective area. As stated at the beginning of this section, this software module has been developed to optimize the computations' speed as much as possible. The effective area must be computed for each region and each energy bin. However, if multiple photon lists have been simulated using the same IRF, the effective area can be computed only once for the first photon list and reused. *OnlinePhotometry* has a *preconfigure_regions* method that performs this computation. If this method is not called, the *integrate* method will compute the effective area for each photon list.

Another optimization trick is considering a ring of regions, i.e., all regions with the same distance from the on-axis pointing. The effective area computation for this group of regions can be done only once since the $\theta$ angle does not change. Figure 3.1 shows an example of multiple ring regions produced by the *RegionsConfig* class. The white region is centered on the source.

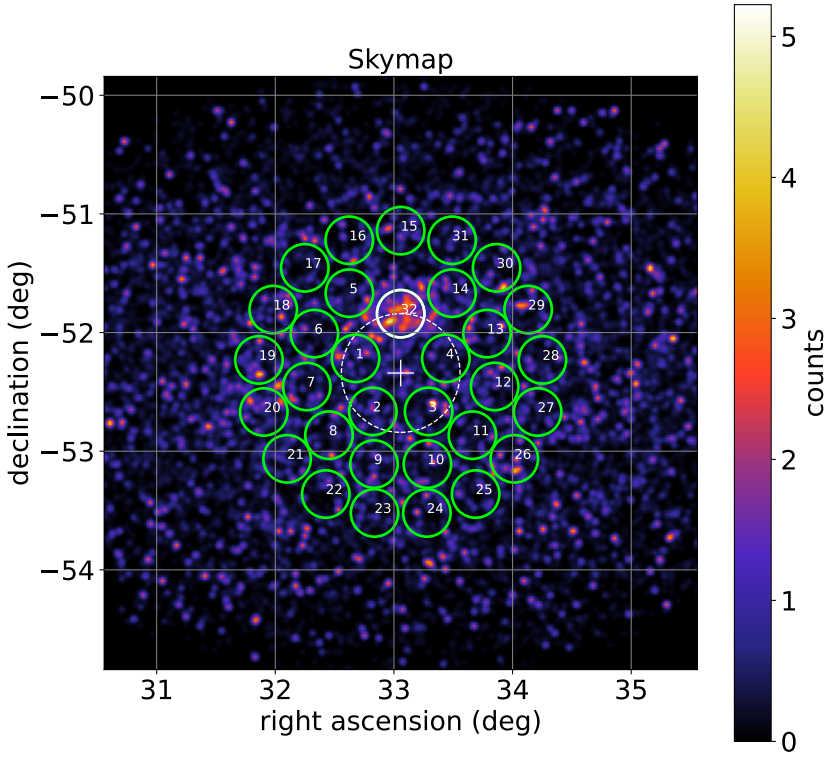


Figure 3.1: An example of the ring regions produced by the *RegionsConfig* class. The white region is centered on the source.

To conclude this section, Figure 6.25 and Figure 6.29 show two examples of the generated time series in the short-exposure scenario (integration time = 5 seconds). The first GRB has a flux greater than the background level, while the second is more difficult to detect. Appendix 6.1 shows more simulated trials.

#### 3.1.1.3 *Time series extractor module*

The last step of the data generation pipeline is to apply a time series extractor module to extract sub-sequences from the time series generated by the previous step. A sub-window of length *sws* slides over the original time series with a specific *stride*, i.e., the distance the sub-window moves at each step. The number of sub-sequences the method will extract is dependent on the chosen length (i.e., the shorter the length, the higher the number of sub-sequences) and the chosen *stride* (i.e., the shorter the *stride*, the higher the number of sub-sequences). To generate the train set the value of the stride can be arbitrary since the training is performed offline. In Section 3.2 we will require our samples to be statistically independent,

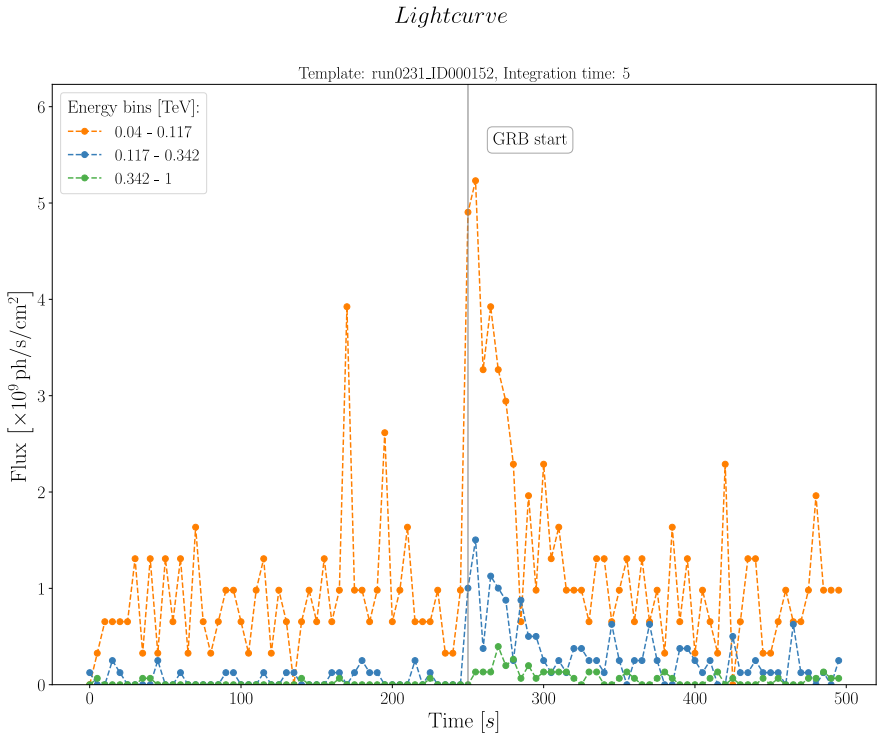


Figure 3.2: A strong GRB, simulated from template run0231_ID000152, in the short-exposure scenario (integration time = 5 seconds).

meaning no overlapping among the sub-sequences. This can be accomplished by setting the value of the *stride* equal to the *sws*. In contrast, during inference, we will need to increase the sample rate generation to increase the inference rate, so a *stride* equal to one is preferred.

As stated before, a semi-supervised data set of background-only samples is generated at the end of the data pipeline. In particular, for each photon list, several multivariate sub-sequences of length *sws*, over three channels, are computed. Each sub-sequence point is a flux measurement aggregating *itime* seconds of the original photon list.

### 3.1.2 *Autoencoder architectures*

As outlined in Section 3.1, the proposed method belongs to the model-based family of techniques to detect multivariate sub-sequence anomalies. It is based on an autoencoder that learns the time series behavior of normal samples. It thereafter uses reconstruction error as anomaly criteria to detect anomalies. The architecture of an autoencoder can exploit different types of layers. A convolutional autoencoder typically consists of multiple layers of convolutional and pooling operations. It allows the network to learn spatial hierarchies of features. On the other hand, an autoen-

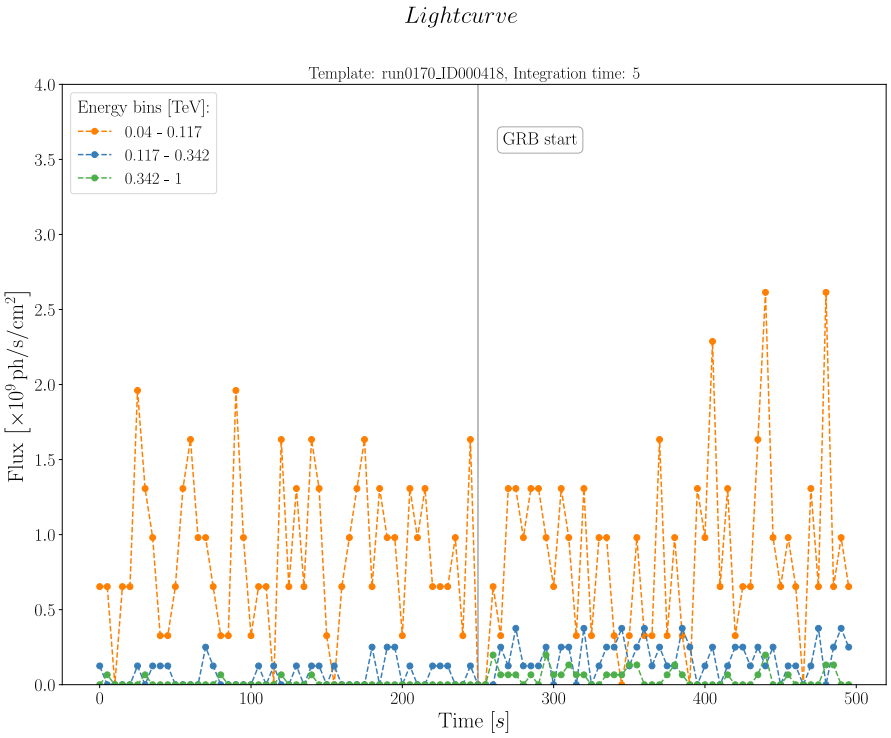


Figure 3.3: A GRB with low intensity with respect to the background level, simulated from template run0170_ID000418, in the short-exposure scenario (integration time = 5 seconds).

coder implemented with recurrent layers is designed to process data sequences, such as text or time series. The autoencoder can learn temporal dependencies and patterns in sequential data with recurrent layers. As mentioned in Section 2.2, more complex and hybrid architectures have been developed, such as Temporal Convolutional Networks to make convolutional layers understand patterns that occur over a prolonged period [Lea+16], ConvLSTM to address the spatiotemporal sequence-forecasting problem [Shi+15] and transformer (BERT), that implements the current state-of-the-art architectures [Dev+18]. In this work, two types of autoencoder architectures have been investigated, based on convolutional and recurrent layers, resulting in different performance outcomes, both in terms of false positives minimization and training requirements. The architecture has been kept small, with no more than four layers and a few thousand learnable parameters. Since the input data has a low dimensionality, too much complex architecture will be able to memorize and overfit the training data. For the same reason, dropout layers are applied in each model architecture. The dropout layer randomly sets input units to 0 with a certain frequency (20% in this case) at each step during training time, which helps prevent overfitting. More complex architectures have not been considered because of the low dimensionality of the input, in terms of time series length and channels.

### 3.1.3 *Autoencoder training*

As stated in Section 2.1, the autoencoder is trained in a semi-supervised setting to learn the time series behavior of normal samples. A semi-supervised background-only time series data set has been generated using the data processing pipeline described in Section 3.1.1. According to the Science Alert Generation (SAG) design requirements (Section 1.2.4.1), the real-time search for transient events should be performed on multiple time scales. The *integration time* setting is used to vary the time scale, integrating the photon counts in tighter or wider time bins. In [Tam20], and [DP+21], the authors explore the Li&Ma and Full-FoV Maximum Likelihood capabilities in the short exposure scenario, using very short integration times. The proposed technique is tested under the same extreme setting. In particular, two training data sets have been used, with integration times equal to 5 seconds and 1 second. I will refer to the first data set as *short-exposure analysis* and to the second as *very short-exposure analysis*. As mentioned in Section 3.1.1.2, the photon lists can be generated and written to the file system as FITS file only once (unless we want to change the simulation parameters, such as the IRF). Then, using the *OnlinePhotometry* class introduced in Section 3.1.1.2, several multivariate time series data sets can be generated. A *DataManager* class has been developed to manage the data used for the autoencoder training and testing. It wraps the *OnlinePhotometry* class to generate the multivariate time series of flux values, extracting the photon counts from multiple regions and applying normalization. This class implements a caching feature to avoid repeating the previous computation multiple times. Finally, it exposes a *get_train_set* method to apply all the required data pre-processing to generate training data. In particular, the first step is to extract sub-sequences from the time series. The stride value is equal to 5 to obtain sub-sequences that do not overlap, but shorter strides could be used. The sub-sequences samples are then split into train and validation sets. A MinMax scaler is fitted on the train set and applied to it, scaling the samples to the $[0, 1]$ interval with the formula $\frac{x-min(x)}{max(x)-min(x)}$. Table 3.1 summarizes the simulation parameters to generate the FITS data set, and Table 3.2 summarizes the parameters to generate the train set. The resulting samples in the short-exposure and very short-exposure settings are, respectively, 48960 for training, 12240 for validation, 68000 for training, and 17000 for validation.

The autoencoders have been trained with the *Adam* optimization [KB14] and a *learning rate* of 0.0001. A 20% dropout is applied after each layer. As shown by Table 3.2, 80% of the samples are used for training, while the remaining 20% is set aside for validation. As mentioned, the inputs are scaled to the $[0, 1]$ interval. The *batch size* has been set to 32 samples. The autoencoder *loss* (or *anomaly score*) is a *weighted mean squared error* (MSE), defined as the following:

| Simulation parameters | |
|---|---|
| trials | 10 |
| simtype | bkg |
| runid | run0406_ID000126 |
| scalefluxfactor | 1.0 |
| caldb | prod5-v0.1 |
| emin | 0.04 |
| emax | 1 |
| irf | North_z40_5h_LST |
| offset | 0.5° |
| roi | 2.5° |
| tobs | 18000 |

Table 3.1: The parameters used to customize the photon lists simulation for the train set generation.

| Train test (short-exposure) | |
|---|---|
| integration_time | 5 |
| number_of_energy_bins | 3 |
| normalize | True |
| sub_window_size | 5 |
| stride | 5 |
| validation_split | 80% |
| Train samples | 48960 |
| Validation samples | 12240 |

| Train test (very short-exposure) | |
|---|---|
| integration_time | 1 |
| number_of_energy_bins | 3 |
| normalize | True |
| sub_window_size | 5 |
| stride | 5 |
| validation_split | 80% |
| Train samples | 68000 |
| Validation samples | 17000 |

Table 3.2: The parameters used to configure the *DataManager* class to extract subsequences with photometry to generate the training set. The left panel shows the parameters of the short-exposure scenario, while the right panel shows the parameters of the very short-exposure scenario.

**Definition 3.1.1.** [Weighted MSE] **WMSE** $= \frac{1}{2D}\sum_{i=1}^{D} \boldsymbol{w}(\boldsymbol{x_i} - \boldsymbol{y_i})^2$ where $\boldsymbol{w} = [\frac{1}{2}, \frac{1}{3}, \frac{1}{6}]$ and D is equal to the number of points.

The reason behind this choice is to give more importance to prediction errors in the lower energy ranges that contain most of the signal. An *early stopping* strategy was considered during training, but the *validation loss* was already flat after five epochs for all models. Hence, the models' weights have been written on disk after five epochs of training. Figure 3.4 shows the training and validation losses for the autoencoder model with recurrent layers trained in the short-exposure scenario. More results can be found in Appendix 7.

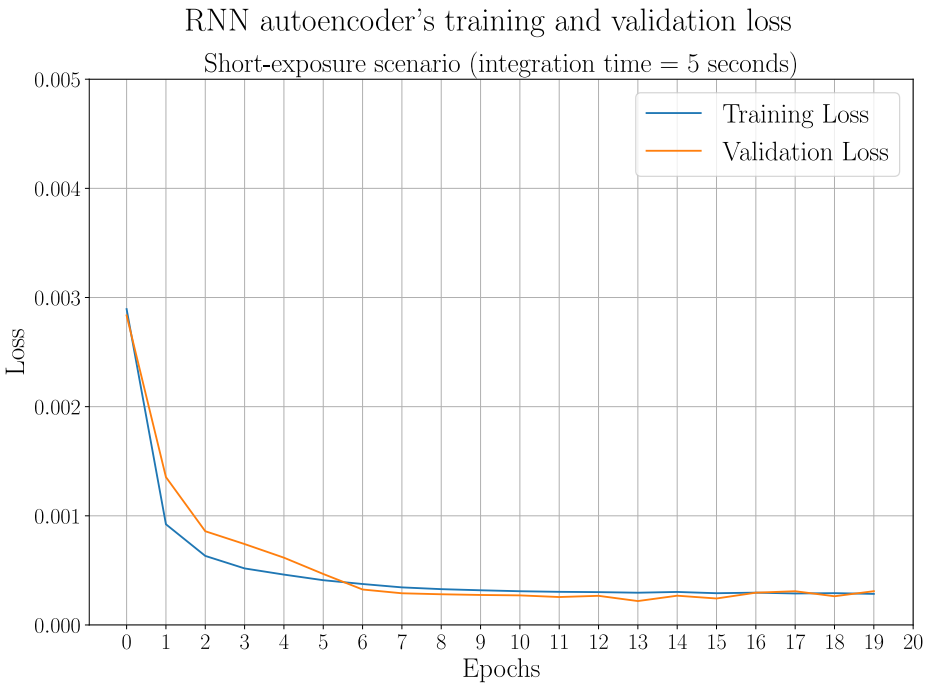


Figure 3.4: An example of train loss and validation loss for the autoencoder model with recurrent layers in the short-exposure scenario (integration time = 5 seconds).

## 3.2 P-VALUE ANALYSIS

As outlined in Section 1.2.4.1, the Science Alert Generation (SAG) system must be able to generate candidate science alerts, meaning that the positive classification produced by the proposed system must be associated with a Gaussian $\sigma$ level of statistical significance. A statistical pipeline based on hypothesis testing has been developed to achieve that. This analysis is used to obtain the threshold $\tau$ to classify the samples as anomalies with a certain $\sigma$ level ([Par+21]) or, on the contrary, to associate each anomaly score with a statistical confidence level (of a positive classification). In hypothesis testing, the initial assumption is that there is no correlation

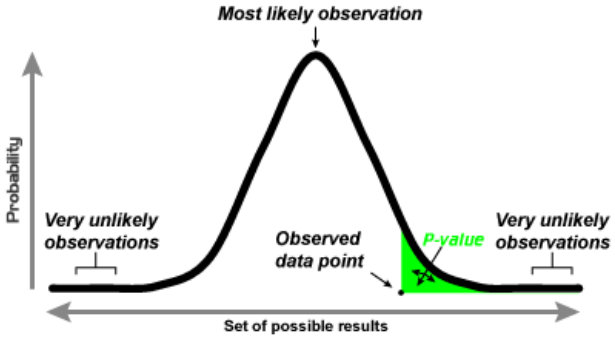


Figure 3.5: A one-tailed test, showing the p-value as the size of one tail. Credits to https://en.wikipedia.org/wiki/One-_and_two-tailed_tests

between the predictor and outcome variables in the population. Through statistical tests utilizing collected data, a determination is made as to whether there is enough evidence to reject the null hypothesis in favor of the alternative hypothesis. A two-tailed hypothesis test is designed to show whether the measure is significantly greater or less than a population's mean. The two-tailed test gets its name from testing the area under both tails (sides) of a normal distribution. On the other hand, a one-tailed hypothesis test is set up to show that the measure would be higher or lower than the population average. Figure 3.5 illustrates a one-tailed test, showing the p-value as the size of one tail. The level of statistical significance, represented by p-values, serves as the benchmark for these tests. The p-value is the likelihood of obtaining the study results by chance if the null hypothesis is true.

**Definition 3.2.1.** The p-value is the inverse of the cumulative distribution function of the test statistic. it defines the probability of accepting the alternate hypothesis with a given level of statistical confidence where in fact the null hypothesis is true:

$$p(ts \geq h) = \int_h^\infty \phi(x)\, dx$$

where $ts$ is the test statistic of the experiment, $\phi$ is the distribution of the test statistic of the observed experiments, and $h$ is a threshold.

In other words, the p-value is the likelihood of committing a type I error (false-positive) if one rejects a null hypothesis that is actually true. On the contrary, a type II error (false-negative) occurs if one fails to reject a null hypothesis that is actually false. The null hypothesis is rejected in favor of the alternative hypothesis if the p-value is less than the established level of statistical significance. For example, we can reject the null hypothesis if the p-value is less than $3x10^{-7}$, meaning $> 5\sigma$ confidence in the alternative hypothesis. Results with p-values lower than $3x10^{-7}$ can still be false positives every once in $3x10^7$ measures ($\pm$ statistical fluctuations). A metric that considers the number of false positives (or false alarms) per the total number of times the event didn't happen is called *false alarm rate* (FAR). A false

alarm rate is also known as the probability of false detection. To not be confused with the false alarm ratio (also abbreviated as FAR), which is the number of false alarms per the total number of alarms. Regarding hypothesis testing, if P is the probability of committing a type I error (rejecting the null hypothesis when it is actually true) and Q is the probability of making a type II error (failing to reject the null hypothesis when it is actually false), in this study, we prioritize minimizing Q over P to minimize the false positive rate and prevent the system from issuing false science alerts to the scientific community.

The null hypothesis in this context is represented by the absence of the GRB event in the data. The anomaly score is the test statistic considered in the p-value analysis. To evaluate the p-values, a data set composed of background-only data samples is fed to the autoencoder, and the distribution $\phi$ of the corresponding anomaly scores is computed. Then, the inverse of the cumulative distribution function is evaluated, obtaining a mapping between the test statistics and the p-values. As stated before, a p-value defines the probability of obtaining statistical confidence greater or equal to a threshold $h$ when the null hypothesis is true.

To reach the desired $5\sigma$ level, about $1e^8$ trials must be simulated and transformed into sub-sequences fed to the autoencoder to obtain the anomaly scores. The same simulation settings are used to generate the training test for the autoencoder models. Hence, more details about the data set generation for the p-value analysis will be provided in Section 3.1.3. There're only two differences concerning the simulation parameters used to generate the training set: the observation time is limited to 100 seconds, and the number of simulated trials is $1e^8$. This process is very computing intensive and has to be repeated several times each time a new model is deployed in production. Since a huge number of trials must be simulated to reach the desired $5\sigma$ level, the simulation script provided by [DP22] has been improved and optimized. As outlined in Section 3.1.1.1, starting from this data set, it is possible to perform different photometry analyses and extract sub-sequences with different lengths. For this reason, a script exploiting batch parallelization with Slurm has been developed to load a trained autoencoder, apply the photometry analysis, generate the sub-sequences, and perform inferences. In particular, each Slurm job takes a batch of photon lists, integrates the photon counts, computes the flux, extracts the time series, scales the data, and performs inference, measuring the anomaly score. To guarantee the statistical independence of these measures, from each photon list, only one sub-sequence is generated and given to the autoencoder. This is why the observation time was limited to 100 seconds. The time took to process $1e^7$ trials (about $3.5TB$ of data), using 100 jobs on 60 Intel(R) Xeon(R) Gold 6240 CPU cores, was about 48 hours. A computational bottleneck was found due to the network filesystem that limited the data transfer from a SATA hard drive.

To find the corresponding p-value for a given sigma level, we first defined the two-tailed probability corresponding to the sigma level of choice. For example, if $sigma = 3$, the corresponding two-tailed probability is $p = 99.73\%$. The survival function gives the p-value: $(1 - p) = 0.0027$. This p-value describes the probability of having a measure outside the $\pm 3\sigma$. Since this use case is constructed on half of a symmetrical distribution, the one-tailed p-value of that probability must be computed. Hence, the previous p-value must be halved, obtaining $(1 - p)/2 = 0.00135$. The p-value is then mapped to a threshold value. On the contrary, if we want to know the sigma level of a particular threshold, we take the corresponding p-value and find the inverse cumulative distribution function relative to that p-value. Again, since the normal distribution is symmetrical, the absolute value is the number of standard deviations rather than $\pm n\sigma$. If the threshold value is absent in the table, it can be interpolated within the closest two values to find the corresponding p-values interpolation.

## 3.3 SYSTEM CONFIGURATION

Figure 3.6 shows the interactions of a subset of the software components configuring the system from the observation metadata. A configuration phase starts when the system receives a new scheduling block (which contains the observation target and metadata). The system automatically caches any trained model to be reused with the same settings. In particular, each model is associated with a particular IRF, which in turn is chosen by considering the following observation metadata:

- the pointing (to define the zenith)
- the observing conditions (e.g., the night sky background)
- the observation duration

Each model is also associated with the shape of the data it was trained with:

- the integration time;
- the length of the single subsequence.

These two parameters are chosen according to the particular physical process we want to detect, but in principle, an ensemble of models trained in different settings can perform multiple predictions. The system will start the training using simulated data if no trained models are available. When the telescope array data-taking starts, a blind-search analysis is applied in the whole field of view to localize candidate sources. The tool will determine the aperture photometry on-region, and starting from it, the off-regions are computed (which number depends on the number of rings to use). The aperture photometry regions will be given to the Data Engine module, completing the system configuration.

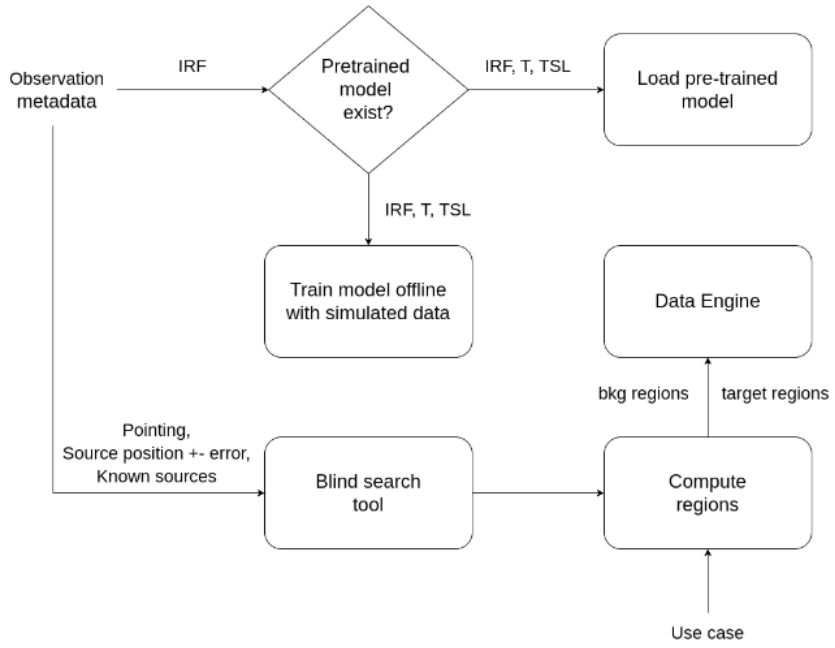


Figure 3.6: A diagram showing the interaction of a subset of the software components developed in this work, highlighting the initial phase that allows configuring the system from the observation metadata.

## 3.4 INFERENCE

Firstly, a bootstrap time is required to obtain the sequence's initial sample, consisting of *sws* points collected over a fixed integration time $T$. The time needed to acquire the first sample is determined by multiplying the number of points *sws* by the integration time $T$. As time progresses, subsequent samples are obtained by waiting for $T$ seconds and allowing the sequence to slide, effectively discarding the oldest point and incorporating the newest one. Once a new sample is obtained, it is fed into an autoencoder, which reconstructs the sequence using its learned representations. The reconstructed samples are then evaluated using an anomaly scoring mechanism, which computes the weighted mean squared error (MSE) defined by 3.1.1. The resulting anomaly score is then compared with a pre-defined classification threshold computed with the p-value analysis (described in Section 3.2) to determine whether the sample is normal or anomalous. Figure 3.7 shows the inference process. Each column is a different sub-sequence sample obtained with a stride equal to 1. Each row represents the energetic bin. The autoencoder reconstructions are the red lines, while in blue are the original samples. Each subplot reports the MSE for each energy bin, and on top of the subplots of the first row, we have the anomaly score (MSE weighted for each energy bin). Each column reports the label of the correct or wrong classification (made using a classification threshold that corresponds to $5\sigma$).

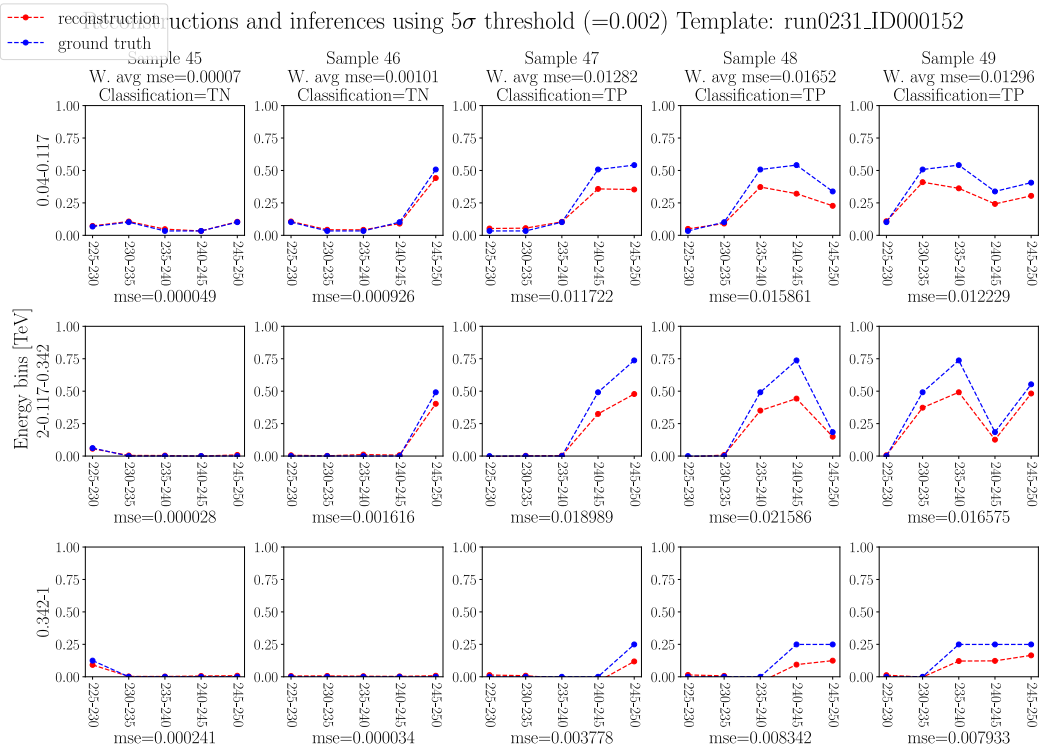


Figure 3.7: Each plot is a sample the autoencoder reconstructs. The red series are the autoencoder's reconstructions, while the original samples have a blue color. Each column holds a different sub-sequence sample obtained with a stride equal to 1. Each row represents the energetic bin. Each subplot reports the MSE for each energy bin, and on top of the subplots of the first row, we have the anomaly score (MSE weighted for each energy bin 3.1.1). Each column reports the label of the correct or wrong classification (made using a classification threshold that corresponds to $5\sigma$).

The anomaly scores obtained in each temporal bin can be translated to the corresponding $\sigma$ levels of significance and plotted. Figure 3.8 shows how the significance varies during a light curve analysis. The x-axis holds the temporal bins of the analysis (a temporal bin corresponds to one column of Figure 3.7). In this particular example, we can see that both anomaly detection and Li&Ma manage to perform a detection (reaching the $5\sigma$ statistical significance). However, the anomaly detection reaches the $5\sigma$ level before Li&Ma, obtaining a faster detection.

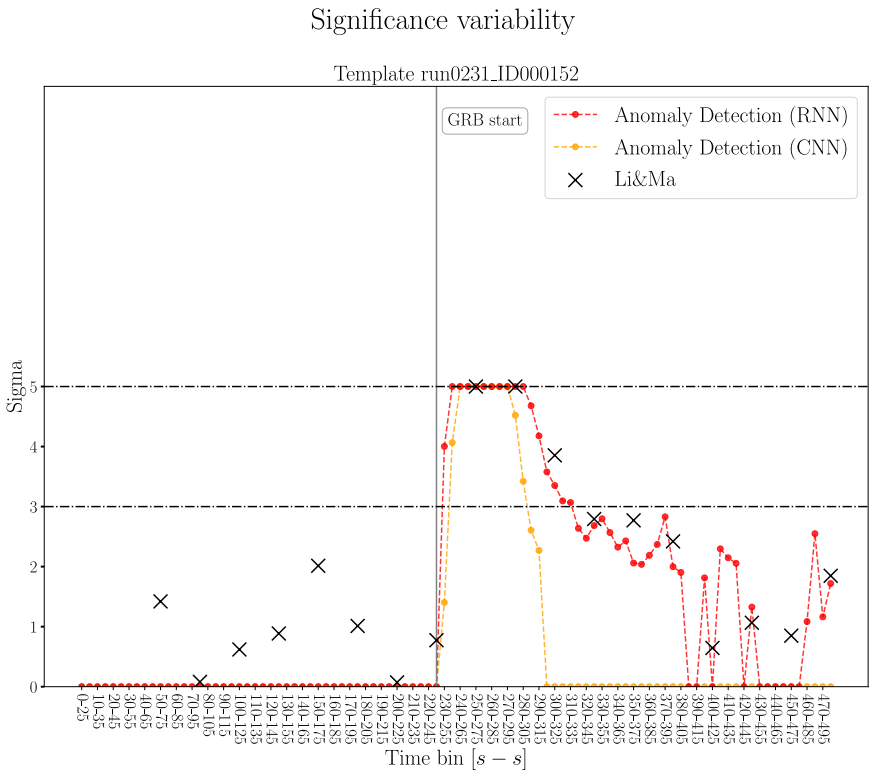


Figure 3.8: The significance variability of both the anomaly detection and Li&Ma analyses applied to a simulated light curve from template run0231_ID000152.

## 3.5 NON-STATIONARY SETTINGS

During inference, the model will process the real-time data generated by the telescopes during an observation. As explained in Section 2.1, the stationarity property of the stochastic process that generates the data can be invalidated by concept drifts. This can be a significant issue as it can negatively impact the performance of a machine learning model trained under a certain statistical distribution of input data. Neural networks are particularly vulnerable to concept drift, as they rely on the assumption that the data distribution remains constant during training and inference. In the presence of a concept drift, the model makes predictions based on outdated information, leading to decreased accuracy. In this context serval concept drifts can happen: when the telescopes repoint to a sky region with a different background (change point), weather changes can degrade the instrument response function, and hardware or software malfunctions can arise.

One approach is to update the model with the current samples. Online learning is a type of machine learning that allows for the analysis and modeling in a streaming fashion, as opposed to batch processing. Without any explicit change detection mechanism, online learning systems can adapt to evolving data. The ad-

vantage of explicit change detection is providing information about the dynamics of the process generating data. However, as time passes, the newly arrived data tend to erase the prior patterns. Learning is inevitably connected with forgetting in models such as artificial neural networks. The ability to continuously learn from a stream of examples while preserving previously learned knowledge is known as the stability plasticity dilemma [CG93]. It is a dilemma because there needs to be a balance between being stable to handle noise and being able to learn new patterns. Some artificial neural networks completely forget past patterns when exposed to new patterns. This phenomenon is known as catastrophic forgetting [Par+19]. A simpler approach is continuously monitoring the model's performance and detecting when a drift has occurred, using techniques such as drift detection methods (DDMs). DDMs can be classified into two categories: instance-based DDMs and feature-based DDMs. Instance-based DDMs monitor the model's error rate on a sliding window of the most recent instances, while feature-based DDMs monitor the distribution of the input features [Gem+20].

In the context of this study, we do not take into account this issue since we rely on the assumption that some software components within the Array Control and Data Acquisition System (ACADA), such as the Data Quality system [Bar+22] and the Monitoring system, will communicate to the Science Alert Generation (SAG) any updates regarding hardware failures, changing in the weather conditions and telescopes repointing to different sky regions.

## 3.6 SYSTEM DESIGN

#### 3.6.0.1 *Technical requirements and problem analysis*

Beyond the requirements of the SAG system, outlined in Section 1.2.4.1, the proposed system must also satisfy the following technical requirements:

- The system must be controlled by the Science Alert Generation (SAG) subsystem of ACADA;
- the system must configure itself using the observation metadata;
- the system must be capable of obtaining the real-time photons list data required for the analysis, and it must provide a real-time data stream to the machine learning models to perform inference;
- the system must provide the operator with the results of the analysis in real-time.

As shown in Figure 1.10, the SAG subsystem of ACADA consists of several components responsible for different use cases. The SAG-SCI software package

supervises the operations of the high-level analysis pipelines. The *RTApipe* framework [Par+22] was designed to simplify the development of real-time scientific analysis pipelines, providing developers with an easy way to integrate new science tools. Hence, integrating the proposed system inside the control flow of ACADA can be addressed by adding it to the set of available science tools in the *RTAPipe* framework. The second requirement imposes that the *scheduling block* data must be provided to the system. As outlined in Section 1.2.4, the scheduling block object is interchanged by the ACADA systems and reaches the SAG system. Hence, the *RTAPipe* framework can access the observation metadata and start the proposed system with the information it needs to auto-configure itself. To satisfy the third requirement, the system must have access to the input data: the photons list data is produced by the analysis pipeline supervised by the SAG-RECO package. These pipelines apply a Random Forest model to discriminate which detected events have been produced by gamma photons and to reconstruct their energy and arrival direction. The output is then written on file in HDF5 format [The00] in the file system location defined by the *SAG Pipeline Sub-Array Supervisor* Alma Common Software component of the SAG-SUP software package. Finally, the last requirement can be fulfilled by exploiting the *RTAPipe* framework that provides a database interface to store the analysis result.

## 3.7 SUMMARY

The previous chapter proposed a method to address the real-time source detection problem. The contribution described the anomaly detection technique, including the data pipeline for input generation, the investigated deep learning architectures, and the training process. The evaluation of the models was addressed in the experiment setup section. The p-value analysis was described to associate positive classifications with Gaussian statistical significance. The potential problems of non-stationary settings during telescope observations and their impact on the proposed system were investigated.

# 4

# RESULTS

This chapter presents the results of the p-value analyses and performance benchmarks. It is organized as follows: Section 4.1 reintroduces the scientific use cases and the assumptions made in these scenarios and then describes the test set generation process. Section 4.2 presents the results of the p-value analysis. Section 4.3 shows a comparison between the two investigated autoencoder architectures. Section 4.4 outlines the performances of the proposed anomaly detection method against the Li&Ma standard technique. The key performance indicators used for the comparison are introduced. The results for both use cases of serendipitous discoveries and follow-up observations are presented in the short-exposure and very short-exposure scenarios.

## 4.1 EXPERIMENTS SETUP

This chapter presents the results of a study aimed at developing an anomaly detection technique for addressing the real-time source detection problem in the context of the Cherenkov Telescope Array Observatory (CTAO). As outlined in Section 3.1, the proposed anomaly detection technique is based on an autoencoder model. Two autoencoder architectures based on convolutional and recurrent layers have been investigated. The comparison of their performances evaluated with standard machine learning metrics will be presented in this chapter. A p-value statistical analysis was performed to determine the classification thresholds the models can use to associate $5\sigma$ statistical significances to positive detections. This chapter will also present the results of these analyses. This chapter focuses on verifying if the proposed technique overcomes the limitations of the standard analyses presented in Section 1.3.2.1 and Section 1.3.3 obtains better performances in practice. Key performance indicators (KPIs) have been defined in the context of two scientific use cases: *serendipitous discovery* and *follow-up observations*, introduced in Section 1.2.5.

The results have been compared against the performances of the Li&Ma analysis technique. The KPIs were first evaluated in the short-exposure scenario, using an integration time of 5 seconds, meaning that each sub-sequence point aggregates 5 seconds of data. Furthermore, the same evaluations were repeated in the very short-exposure scenario, using an integration time of 1 second. The next section will re-introduce the scientific use cases of serendipitous discovery and follow-up observations, highlighting the assumptions and the experiments' setup.

### 4.1.1 *Scientific use cases and assumptions*

The serendipitous discoveries use case applies in the scenario where the telescopes observe a certain sky region and an unexpected event is seen in the field of view. This kind of discovery is possible even though the probability of a serendipitous GRB event appearing in the field of view during an observation is very low. Indeed, telescopes can systematically observe large sky regions, for example, during surveys, increasing the probability of serendipitous discoveries. Such events are scientifically crucial because the overall evolution of the GRB event can be observed from the beginning. To maximize the scientific return, detecting the serendipitous source should be addressed as soon as possible to let other observatories acknowledge the discovery through a science alert broadcasted to the scientific community. By doing so, the same events can be studied with other instruments, exploring different wavelengths or messengers. As explained in Section 1.3.2, the aperture photometry method defines an aperture, a closed region centered on the source, to count the on-source photons. The issue is that we don't know in which region the GRB event will appear. One way to address the problem is to use several regions, as shown by Figure 3.1. The problem with this approach is the significant increase in the number of analysis trials, which must be considered as a post-trials probability. In the context of this work, it is assumed a blind-search analysis is applied in the whole field of view to localize candidate sources in the least possible amount of time. CTools provide one example of blind-search implementation [Kno+16], executing a peak-detection algorithm on smoothed count maps with a localization acceptance threshold above a given significance. It is assumed that the blind-search analysis will always find the region with the GRB event in its center for each event considered in the performance analysis. It is also assumed that blind search will always take a fixed time of 10 seconds to find the region.

The observatory must not only be the sender of scientific alerts but also be capable of receiving and reacting to external ones. The follow-up observation use case models the scenario in which the observatory receives a science alert and changes the pointing of its telescope to a new sky region to detect the event. Several considerations should be made. The time from the event's start and the taking of the first

photon counts are variable. It depends on two factors: the time to receive the scientific alert and the time the telescopes take to change the pointing. This delay has been considered with four values: 25, 50, 75, and 100 seconds. After this delay, the analysis techniques can perform classifications once the required data is obtained. Unlike the serendipitous discovery scenario, the evolution of the GRB event cannot be observed from the beginning. The luminosity of a GRB event tends to decrease over time, increasing the difficulty of detection. Another factor to consider is the localization error on the position of the source provided by the received scientific alert. This error can be smaller than the field of view or much larger, as in the case of a gravitational-wave alert. The latter case would require multiple observations performed with a tiling strategy [SA+19], but this is outside this work's scope. If multiple aperture photometry regions must cover the localization error within the field of view, the blind-search approach used in the serendipitous scenario will also be considered here.

### 4.1.2 *Test set generation*

The test set used for the performance evaluation of the proposed anomaly detection method is a supervised data set containing two classes of samples. The first class of samples has been generated considering the background model in the simulations. The second class of samples has been generated considering the background and source models in the simulations. The Gamma-Ray Bursts (GRBs) simulation templates were taken from the POSyTIVE catalog [Ber+19]. The mock GRB population used by POSyTIVE is calibrated using a 40-year data set of multi-wavelength GRB observations, providing about 20 thousand GRB templates of known events. As stated in Section 1.4, GRBs are characterized by several key properties, including duration and peak flux. The latter measures the maximum energy level emitted by the event. It is usually measured in units of $erg/s/cm^2$ and is one of the most important parameters for characterizing the luminosity of a GRB. GRBs can have a wide range of peak fluxes, from as low as $10^-9 erg/s/cm^2$ to as high as $10^-5 erg/s/cm^2$, depending on the distance and their intrinsic properties. If the luminosity is too low, the GRB event will be indistinguishable from the background noise. For this reason and computational resource limitations, only a subset of the catalog has been simulated. The mean of the background level has been considered a threshold to limit the number of simulations. The test set has been generated simulating all the templates whose peak flux is greater or equal to the background level mean minus $1\sigma$ to account for the background fluctuations. Figure 4.1 shows the distribution of the peak flux of each GRB template and highlights the selected set. As explained in Section 3.1.1.1, the simulation tool generates a photons list, i.e., a list of photons whose energy and arrival direction

are reconstructed. Only the GRB afterglow model has been simulated since the prompt models were not available at the time of writing. The simulation tool has been used to generate one simulated trial for each selected template. The resulting test set contains 419 trials with a peak flux in the interval $[2.6785e^{-10}, 1.1688e^{-04}]$. The simulation time is limited to 500 seconds because we want to detect the event

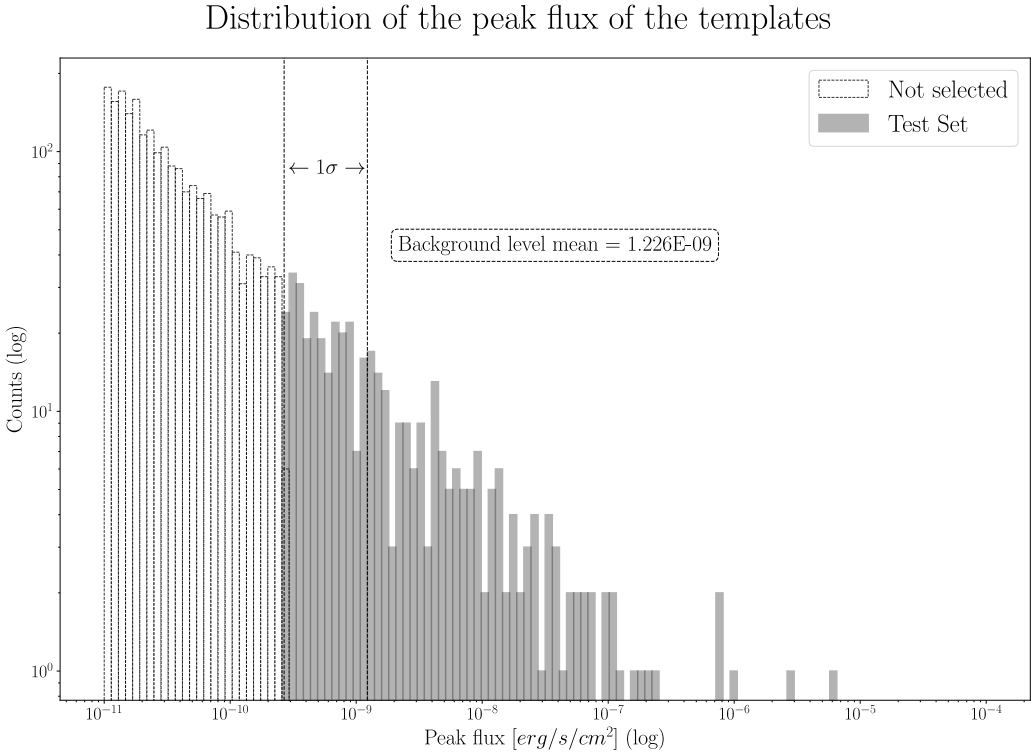


Figure 4.1: The distribution of the peak flux for each GRB template. The number of selected templates is 419, and their peak fluxes are within the $[2.6785e^{-10}, 1.1688e^{-04}]$. range.

as soon as possible. The trigger time that defines the start of the GRB event is fixed and equal to 250 seconds. The instrument response function (IRF) models the response of a sub-array configuration of 4 LST-1 telescopes located in the northern hemisphere site, observing for 5 hours at 40° zenith angle (the same IRF used for generating the training set). As described in Section 3.1.1.2, the photons lists must be integrated in space, time, and energy to generate the multi-variate time series of flux measurements. The integration time is set to 5 seconds to evaluate the performances of the techniques in the short-exposure scenario and 1 second for the very short-exposure scenario. The integration process generates time series of 100 (short-exposure scenario) and 500 (very short-exposure scenario) flux measurements. Finally, as described in Section 3.1.1.3, the time series extractor module extracts sub-sequences from the generated time series. The length of the sub-sequences is fixed and set to 5 points, the same setting used to generate the

| Simulation parameters | |
|---|---|
| trials | 1 |
| simtype | grb |
| runid | [List of GRB templates] |
| scalefluxfactor | 1.0 |
| caldb | prod5-v0.1 |
| emin | 0.04 |
| emax | 1 |
| irf | North_z40_5h_LST |
| offset | 0.5° |
| roi | 2.5° |
| tobs | 500 |

Table 4.1: The parameters used to customize the photon lists simulation for the test set generation. The list of templates (*runid*) has been omitted due to limited space.

| Test test (short-exposure) | |
|---|---|
| integration_time | 5 |
| number_of_energy_bins | 3 |
| normalize | True |
| sub_window_size | 5 |
| stride | 1 |
| Normal samples | 40224 |
| Anomalous samples | 20531 |

| Test test (very short-exposure) | |
|---|---|
| integration_time | 1 |
| number_of_energy_bins | 3 |
| normalize | True |
| sub_window_size | 5 |
| stride | 1 |
| Normal samples | 207824 |
| Anomalous samples | 104331 |

Table 4.2: The parameters used to configure the *DataManager* class to extract sub-sequences with photometry and to generate the test set. The left panel shows the parameters of the short-exposure scenario, while the right panel shows the parameters of the very short-exposure scenario.

data to train the auto-encoder. The stride parameter is set to 1 because the anomaly detection technique can use overlapping temporal bins, drastically reducing the time the model waits for data during the online inference. In contrast, the Li&Ma technique must use temporal bins that are statistically independent. Finally, each sub-sequence is associated with a label. A sub-sequence is labeled as anomalous if at least one of its points exceeds the trigger time threshold that defines the start of the GRB event. For the short-exposure scenario (integration time = 5 seconds), the number of generated test samples is 40224, of which 20531 are labeled anomalous. For the very short-exposure scenario (integration time = 1 second), the number of test samples is 207824, of which 104331 are labeled anomalous. These configuration parameters are reported in Table 4.1 and Table 4.2.

## 4.2 p-value analysis results

As outlined in Section 3.2, the p-value analysis is used to obtain the threshold $\tau$ to classify the samples as anomalies with a certain $\sigma$ level or, on the contrary, to associate with each anomaly score a statistical confidence level (of a positive classification). Four p-value analyses have been performed, one for each autoencoder model (convolutional and recurrent) in the short-exposure and very short-exposure settings. The models performed inferences on $1e^8$ background-only samples, and the distributions of the anomaly score (weighted mean squared error) have been computed. Then, the inverse cumulative of the distribution function was evaluated, and the mapping between the y-axis (p-values) and the x-axis (anomaly scores) was written on disk.

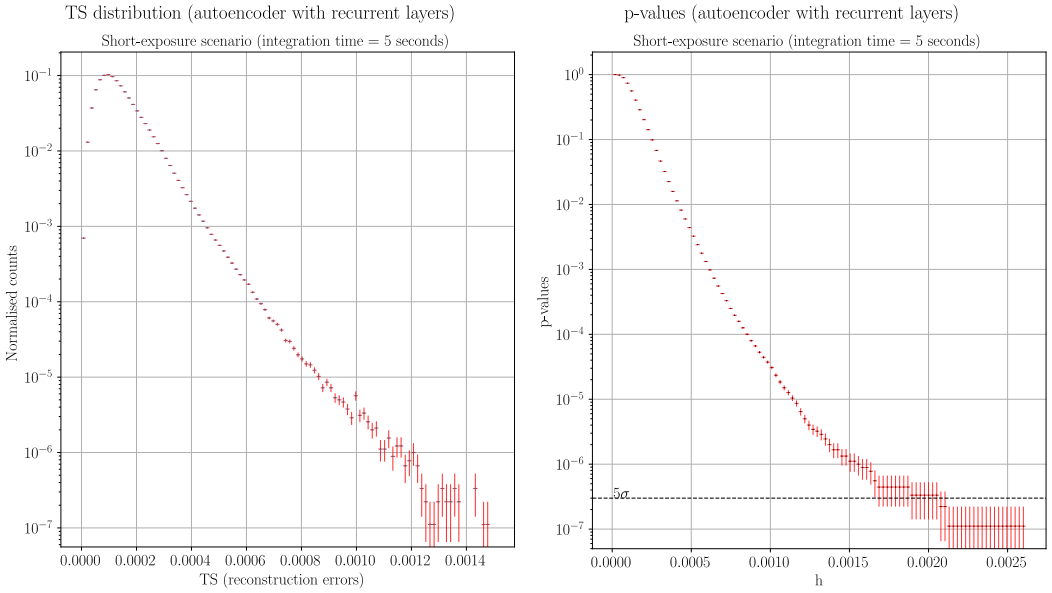


Figure 4.2: TS distribution and p-values for the autoencoder model with recurrent layers in the short-exposure scenario (integration time = 5 seconds).

Figure 4.2 shows the p-value analysis results for the autoencoder model with recurrent layers in the short-exposure scenario. The left panel shows the test statistic distribution, and the right panel shows the corresponding p-values. Table 4.3 shows the actual values of the anomaly score - sigma level mapping. The number of rows of the table was limited to only certain levels of sigma. Chapter 8 lists the p-value analysis results for each model and analysis setting.

| p-values | | | |
|---|---|---|---|
| Threshold | p-value | ± error | Sigma |
| 0.003171 | 1.283972e-01 | 1.201785e-04 | 1.134 |
| 0.003751 | 6.115264e-02 | 8.293861e-05 | 1.545 |
| 0.004476 | 2.277199e-02 | 5.061155e-05 | 2.000 |
| 0.005492 | 5.323285e-03 | 2.447028e-05 | 2.554 |
| 0.006507 | 1.228234e-03 | 1.175411e-05 | 3.029 |
| 0.007667 | 2.312711e-04 | 5.100465e-06 | 3.502 |
| 0.009118 | 3.149606e-05 | 1.882250e-06 | 4.001 |
| 0.010859 | 3.262092e-06 | 6.057553e-07 | 4.509 |
| 0.013760 | 4.499438e-07 | 2.249719e-07 | 4.912 |
| 0.013905 | 2.249719e-07 | 1.590791e-07 | 5.047 |

Table 4.3: An example of p-value analysis for the autoencoder model with recurrent layers in the short-exposure scenario (integration time = 5 seconds). The table shows a subset of all the rows. Only the threshold values corresponding to predefined sigma levels are shown.

## 4.3 COMPARISON OF THE AUTOENCODER ARCHITECTURES

As described in Section Section 3.1, the proposed anomaly detection technique utilizes an autoencoder model. Both convolutional and recurrent layer-based architectures for autoencoders have been examined. The chapter will compare their performances using standard machine learning metrics. The accuracy, precision, recall, and false positive rate metrics express the percentage of the time series correctly classified, what proportion of positive identifications was correct, what proportion of actual positives was identified correctly, and what proportion of the actual negative events were wrongly categorized as positive. While precision measures the probability of a sample classified as positive to be an actual positive sample, the false positive rate measures the ratio of false positives within the negative samples. The accuracy, precision, and recall values computed with different thresholds can help understand the evolution of trade-offs between the number of false positive and false negative classifications. Figure 4.3 shows the autoencoder model's accuracy, precision, and recall curves with recurrent layers in the short-exposure settings (integration time = 5). Table 4.4 and Table 4.5 show the actual values of accuracy, precision, recall, and false positive rate computed using the $5\sigma$ threshold, for both architectures and short-exposure settings. The autoencoder with recurrent layers obtained better accuracy and recall in both scenarios. At the same time, the autoencoder with convolutional layers is less precise but has a better false positive rate. Appendix 7 presents the same metrics for each model and analysis setting.

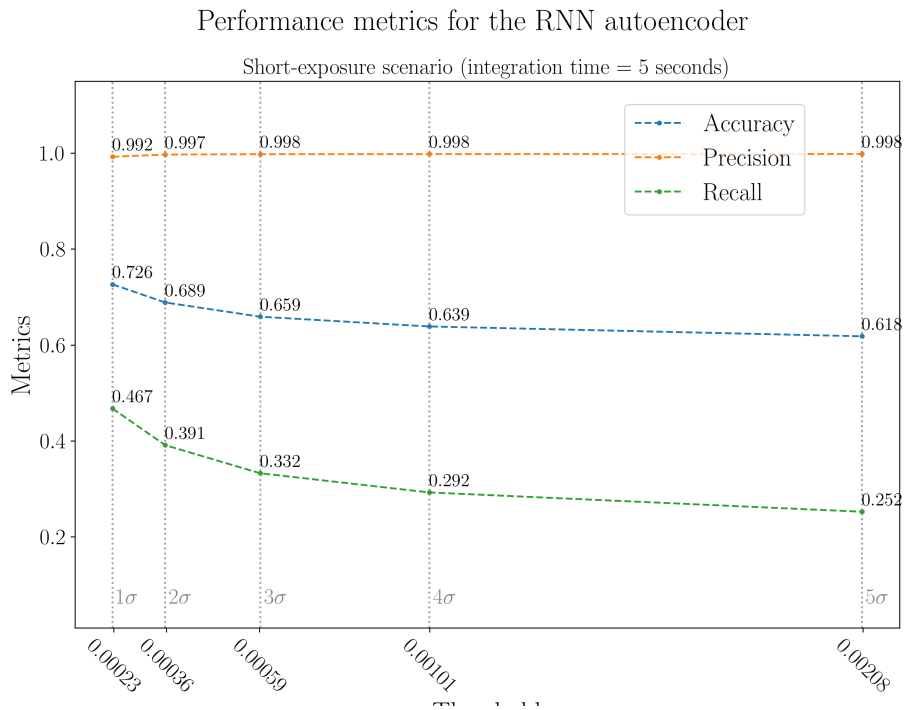


Figure 4.3: Accuracy, precision, and recall curves for the autoencoder model with recurrent layers in the short-exposure settings (integration time = 5).

| Model | Accuracy | Precision | Recall | False Positive Rate |
|---|---|---|---|---|
| RNN | 61.81% | 99.83% | 25.22% | 0.17% |
| CNN | 59.96% | 99.86% | 21.58% | 0.13% |

Table 4.4: Standard metrics in the short-exposure settings (integration time = 5), using the $5\sigma$ threshold. The number of test samples is 40224, of which 20531 are anomalous.

| Model | Accuracy | Precision | Recall | False Positive Rate |
|---|---|---|---|---|
| RNN | 57.05% | 99.97% | 14.46% | 0.033% |
| CNN | 55.58% | 99.98% | 11.52% | 0.016% |

Table 4.5: Standard metrics in the very short-exposure settings (integration time = 1), using the $5\sigma$ threshold. The number of test samples is 207824, of which 104331 are anomalous.

## 4.4 EVALUATION OF THE PROPOSED TECHNIQUE AGAINST LI&MA

The performances of the proposed anomaly detection method have been evaluated against Li&Ma, the standard technique used to address the source detection problem in the ground-based gamma-ray astronomy field.

### 4.4.1 *Key performance indicators*

The first key performance indicator is the cumulative number of $5\sigma$ detections. This metric evaluates the robustness of the techniques. As outlined in Section 4.1.2, the data set for testing is composed of 419 simulated GRB events. The number of $5\sigma$ detections is evaluated for each time bin whose time span depends on the integration time and the length of the sub-sequences. This number is cumulated for the following time bins.

For the reasons stated in the previous section, evaluating how fast the technique is to issue $5\sigma$ detections is crucial. Hence the second key performance indicator represents the average time to perform detections. This metric is called *Detection Delay* (DD) and is evaluated as follows. For each time bin, the number of seconds between the start of the GRB event ($t_{grb} = 250s$) and the detection time is computed. This delay ($d_{tb}$) is the same for each detection performed in the same time bin. For example, if the autoencoder model performs a detection in time bin $[230 - 255]$, the delay ($d_{[230-255]}$) is equal to 5 seconds ($255 - 250$). The delay is computed for each time bin. Then, a weighted average is evaluated, considering the number of the detections in common to the two techniques within the time bins, from $[245 - 250]$ to $tb_{max}$. Considering the short-exposure scenario, the detection delay (DD) is defined as:

$$DD = \frac{1}{tcd} \sum_{tb=[245-250]}^{tb_{max}} w_{tb} * d_{tb}$$

Where $tcd$ is the total number of detections in common to the two methods, $w_{tb}$ represents the number of detections made in a particular time bin, and $d_i$ is the delay expressed as the difference between $tb$ and the time bin of the start of the GRB event ($t_{grb} = 250s$).

### 4.4.2 *Serendipitous discovery results*

In the serendipitous discovery scenario, while the telescopes observe a particular target, an unexpected event appears in the field of view. The issue is that we don't know in which region the GRB event will appear. This study assumes a blind-search analysis applied to the whole field of view to determine the aperture photometry regions. In particular, it is assumed that the blind-search analysis will always find the region that contains the GRB event in its center and will take a fixed time of 10 seconds to complete.

#### 4.4.2.1 *Short-exposure scenario*

Figure 4.4 shows the cumulative number of detections performed by the anomaly detection and Li&Ma techniques for each temporal bin in the short-exposure scenario (integration time = 5 seconds). The x-axis holds each temporal bin. The y-axis holds the cumulative number of detections among all the GRB events. The vertical dashed line corresponds to the start of the GRB event. The grey area represents the application of the blind-search algorithm, so even if the series starts at bin $[225 - 250]$, only the points outside the grey area must be considered. The

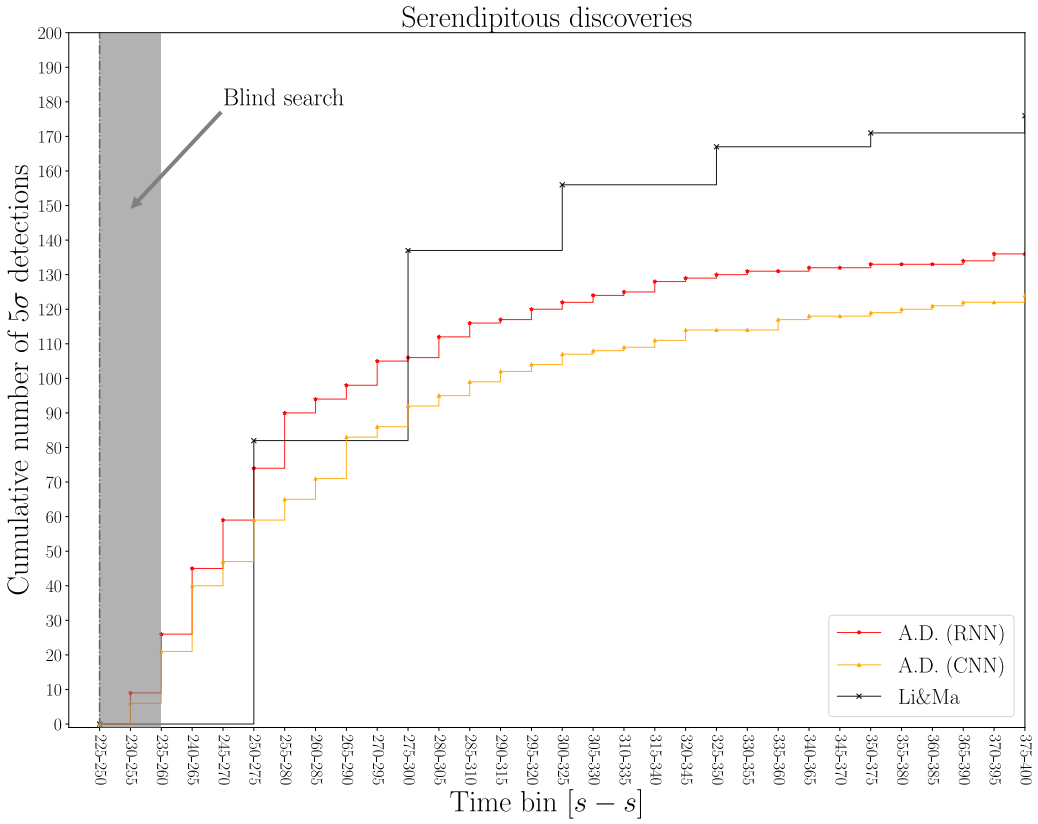


Figure 4.4: The cumulative number of detections of anomaly detection and Li&Ma techniques for each temporal bin in the short-exposure scenario (integration time = 5 seconds). The x-axis holds each temporal bin. The y-axis holds the cumulative number of detections among all the GRB events. The vertical dashed line corresponds to the start of the GRB event. The grey area represents the application of the blind-search algorithm.

figure shows the robustness of the Li&Ma technique, detecting at the end of the observation more GRBs than the anomaly detection technique. However, we are interested in the first part of the observation since detections must be performed as soon as possible. The anomaly detection technique can be applied after every $T = 5$ seconds of new data. In contrast, Li&Ma is limited by statistical indepen-

dence assumption and can be applied only for temporal bins that do not overlap. To quantify how fast the anomaly detection technique is to perform detections concerning Li&Ma, Table 4.6 shows the DD metric. The first temporal bin considered in the table starts when the blind-search algorithm ends. The next temporal bins are the ones in which Li&Ma was applied. Table 4.6 proves the ability of the anomaly detection technique to perform faster detections with respect to Li&Ma.

| **Detection delay** (common detections) | | | | | | |
|---|---|---|---|---|---|---|
| Temporal bins | CNN | | RNN | | Li&Ma | |
| 250-275 | 6.86 | $\pm$5.41 | 6.37 | $\pm$5.57 | 15.00 | $\pm$0 |
| 275-300 | 10.78 | $\pm$9.25 | 12.04 | $\pm$11.56 | 21.28 | $\pm$10.87 |
| 300-325 | 14.28 | $\pm$13.69 | 15.98 | $\pm$15.58 | 24.87 | $\pm$14.6 |
| 325-350 | 17.02 | $\pm$17.31 | 18.87 | $\pm$18.75 | 27.64 | $\pm$17.9 |
| 350-375 | 18.98 | $\pm$19.88 | 21.31 | $\pm$21.79 | 29.59 | $\pm$20.25 |

Table 4.6: Detection delay (in seconds) for serendipitous discoveries in the short-exposure scenario (integration time = 5 seconds) with common detections.

#### 4.4.2.2 *Very short-exposure scenario*

The same metrics are evaluated again in the very short-exposure scenario, with an integration time equal to 1 second. In this scenario, the number of photon counts is further reduced, often exceeding the lower bounds of the Li&Ma technique (equal to 10 photon counts from the on and off regions), limiting its applicability. In contrast, it reduces from 5 to 1 second the time the anomaly detection technique needs to wait for new data since a new flux data measurement is available each second, and a new sub-sequence is ready for analysis. This time is also reduced for Li&Ma, from 25 to 5 seconds. Figure 4.5 shows that the number of detections performed by the anomaly detection technique with recurrent layers is always greater than Li&Ma's, proving greater robustness in this extreme scenario. As said before, the Li&Ma is not always applicable due to its limitation of requiring at least ten photon counts. The reduced number of photon counts also negatively affects the performance of the anomaly detection technique. It's also worth noticing that the anomaly detection technique with convolutional layers performs better with respect to Li&Ma from the beginning of the observation until the $[295 - 300]$ temporal bin. Concerning the detection delay, Table 4.6 and Table 4.7 show that the anomaly detection technique is faster with respect to Li&Ma, also in the very short-exposure scenario.

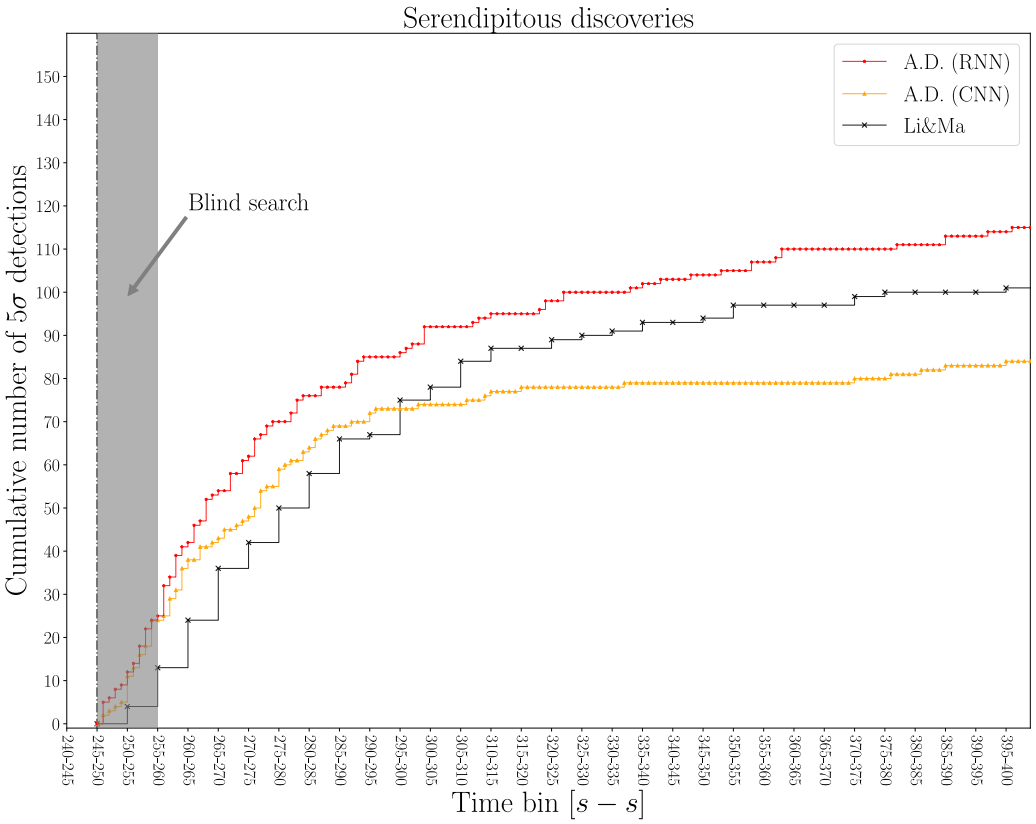


Figure 4.5: The cumulative number of detections of anomaly detection and Li&Ma techniques for each temporal bin in the very short-exposure scenario (integration time = 1 second). The x-axis holds each temporal bin. The y-axis holds the cumulative number of detections among all the GRB events. The vertical dashed line corresponds to the start of the GRB event. The grey area represents the application of the blind-search algorithm.

| **Detection delay** (common detections) | | | | | | |
|---|---|---|---|---|---|---|
| Temporal bins | RNN | | CNN | | Li&Ma | |
| 280-285 | 10.11 | ±7.46 | 11.04 | ±7.54 | 11.57 | ±8.46 |
| 305-310 | 13.77 | ±11.69 | 13.86 | ±9.86 | 17.01 | ±13.86 |
| 330-335 | 15.39 | ±13.56 | 15.93 | ±12.46 | 19.8 | ±16.39 |
| 355-360 | 17.24 | ±16.41 | 17.14 | ±13.95 | 22.31 | ±19.48 |
| 380-385 | 18.95 | ±19.27 | 18.65 | ±17.28 | 24.35 | ±22.23 |

Table 4.7: Detection delay (in seconds) for serendipitous discoveries in the very short-exposure scenario (integration time = 1 second) with common detections.

### 4.4.3 *Follow-up observation results*

In the follow-up observation scenario, the observatory receives a scientific alert and adjusts the pointing of its telescope to a new sky region to detect the event. This study considers the variable delay between the event's start and the taking of the first photon counts, which depends on the time to receive the scientific alert and the time the telescopes take to change the pointing. Four delay values are considered (25, 50, 75, and 100 seconds). As discussed in Section 4.1.1, a blind-search algorithm is assumed to start at soon as the telescopes take data. As in the serendipitous discovery scenario, the blind-search algorithm is always assumed to find the right region in a fixed time of 10 seconds. The cumulative detection plot is repeated four times for each delay value in different sub-plots. Each subplot shows a vertical line representing the follow-up observation's start. The next vertical line represents the time the techniques must wait to gather enough data to make the first inference.

#### 4.4.3.1 *Short-exposure scenario*

Due to the delay that postpones the observation, the evolution of the GRB event cannot be observed from the beginning. The luminosity of a GRB event tends to decrease over time, increasing the difficulty of detection. Figure 4.6 shows the superiority of the Li&Ma technique in the short-exposure scenario. The cumulative number of detections is greater for Li&Ma for every temporal bin, and as a consequence, its detection delay, shown in Table 4.8, is also better.

| **Detection delay** (common detection) | | | | | | |
|---|---|---|---|---|---|---|
| Temporal bins | RNN | | CNN | | Li&Ma | |
| 275-300 | 25 | $\pm 0$ | 25 | $\pm 0$ | 25 | $\pm 0$ |
| 300-325 | 35.94 | $\pm 7.79$ | 33.54 | $\pm 9.26$ | 28.68 | $\pm 8.89$ |
| 325-350 | 41.5 | $\pm 13.5$ | 38.37 | $\pm 13.59$ | 32.01 | $\pm 13.83$ |
| 350-375 | 45 | $\pm 17.08$ | 43.04 | $\pm 18.42$ | 34.32 | $\pm 16.95$ |
| 375-400 | 48.71 | $\pm 21.82$ | 48.25 | $\pm 24.32$ | 36.71 | $\pm 20.75$ |

Table 4.8: Detection delay (in seconds) for follow-up observations in the short-exposure scenario (integration time = 5 seconds) for common detection.

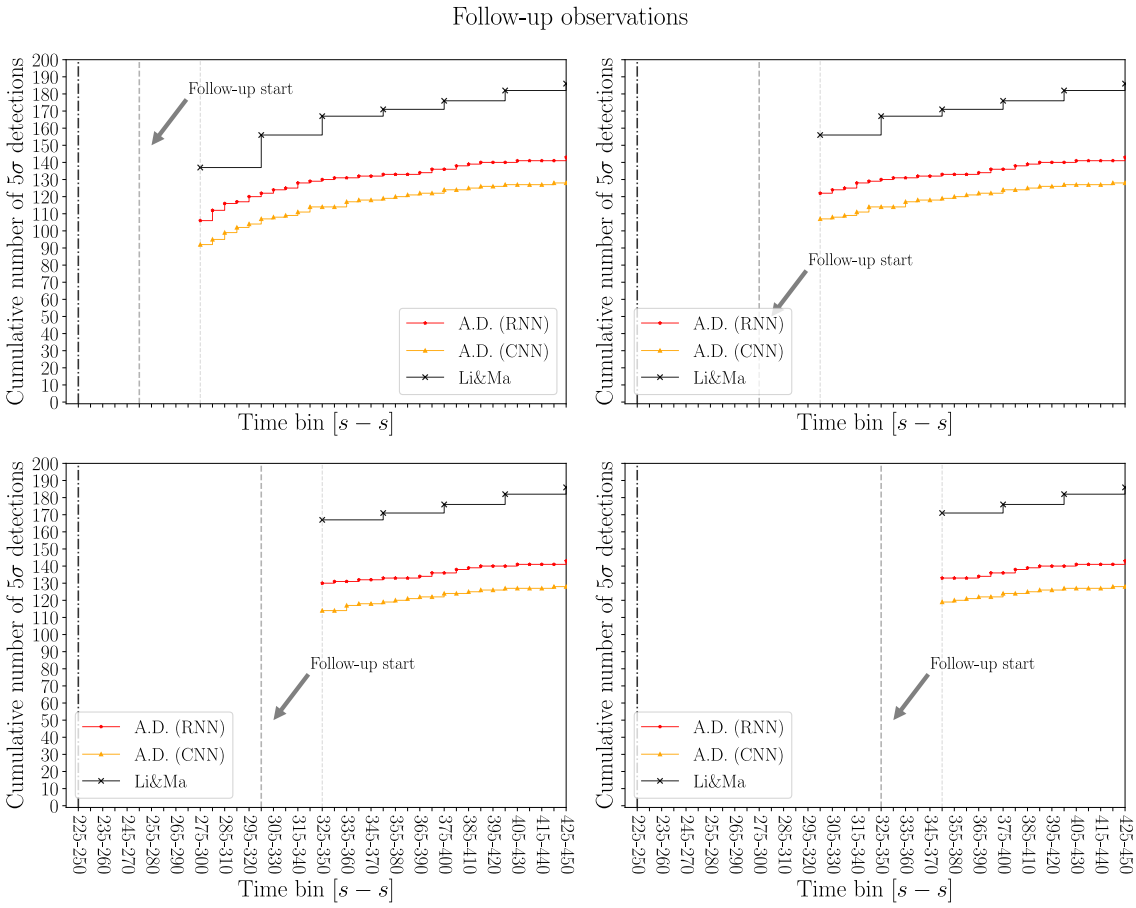


Figure 4.6: The cumulative number of detections of anomaly detection and Li&Ma techniques for each temporal bin in the short-exposure scenario (integration time = 5 seconds). The x-axis holds each temporal bin. The y-axis holds the cumulative number of detections among all the GRB events. The vertical dashed line corresponds to the start of the GRB event.

#### 4.4.3.2 *Very short-exposure scenario*

The same metrics were evaluated in a short-exposure scenario with 1 second of integration time. The same considerations for the serendipitous discovery use case can be made here: the short exposure leads to fewer photon counts and faster data availability. The reduced photon counts negatively affect the performance of the anomaly detection and Li&Ma techniques. The anomaly detection technique based on the autoencoder with recurrent layers showed greater robustness than Li&Ma and was faster in detecting anomalies. Figure 4.7 shows the cumulative number of detections. Also, in this case, the anomaly detection technique with convolutional layers performs better with respect to Li&Ma from the beginning of the observation until the $295 - 300$ temporal bin. The detection delay, shown in Table 4.9, is always lower for anomaly detection with recurrent and convolutional layers, despite the

moment the Li&Ma starts to perform better than the CNN model from bin 295 – 300.

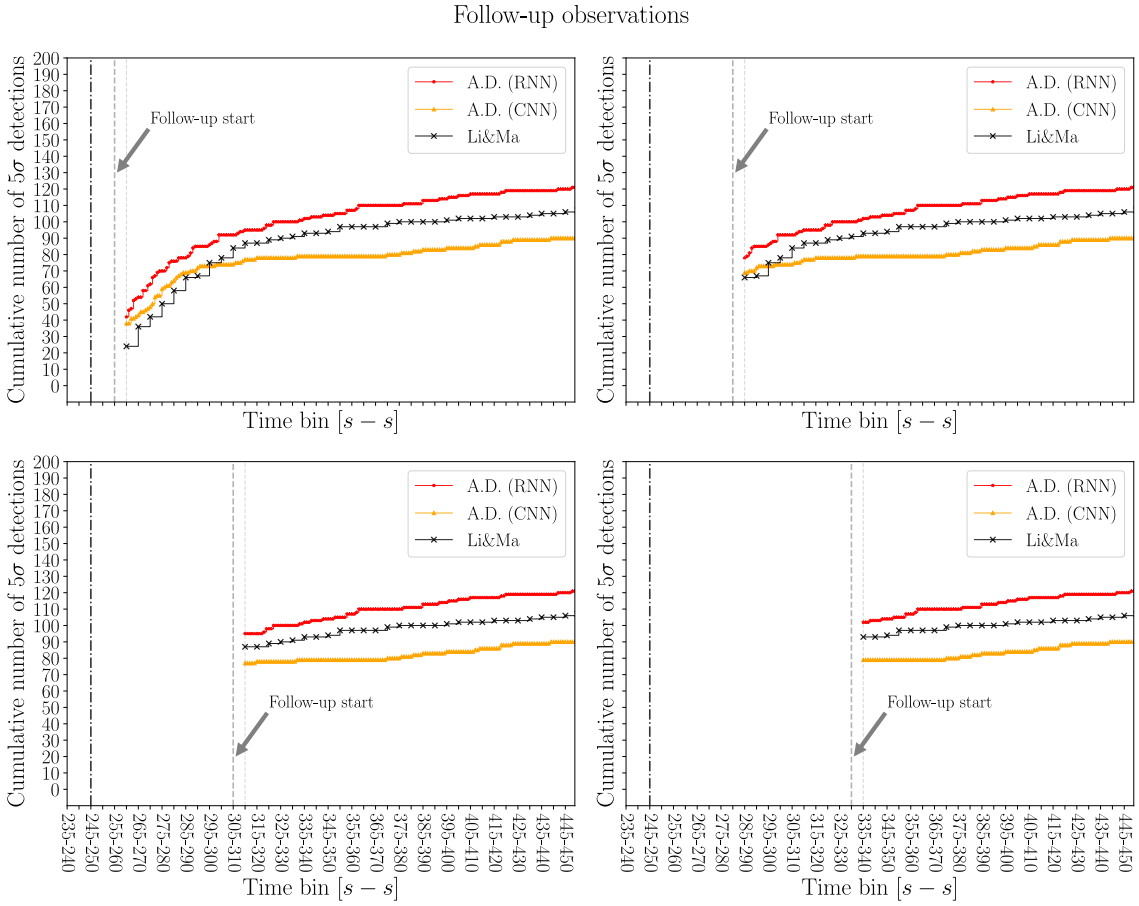


Figure 4.7: The cumulative number of detections of anomaly detection and Li&Ma techniques for each temporal bin in the very short-exposure scenario (integration time = 1 second). The x-axis holds each temporal bin. The y-axis holds the cumulative number of detections among all the GRB events. The vertical dashed line corresponds to the start of the GRB event.

| **Detection delay** (common detection) | | | | | | |
|---|---|---|---|---|---|---|
| Temporal bins | RNN | | CNN | | Li&Ma | |
| 260-265 | 5 | ±0 | 5 | ±0 | 5 | ±0 |
| 290-295 | 15.78 | ±7.71 | 16.74 | ±6.82 | 14.65 | ±7.82 |
| 315-320 | 18.74 | ±11.07 | 19.13 | ±11.26 | 19.62 | ±12.74 |
| 340-345 | 20.46 | ±13.2 | 20.31 | ±13.14 | 22.49 | ±15.72 |
| 365-370 | 22.86 | ±16.96 | 20.82 | ±13.83 | 24.78 | ±18.48 |

Table 4.9: Detection delay (in seconds) for follow-up observations in very the short-exposure scenario (integration time = 1 second) for common detections.

## 4.5 SUMMARY

This chapter presented the results of p-value analyses and performance benchmarks. It reintroduced the scientific use cases and the assumptions made in these scenarios and described the process of test set generation. The results of the p-value analysis were shown, followed by a comparison between the two investigated autoencoder architectures. The performance of the proposed anomaly detection was compared against Li&Ma, using key performance indicators, with results presented for both serendipitous discoveries and follow-up observation scenarios in the short-exposure and very short-exposure.

# 5

# CONCLUSIONS

This chapter summarizes the key outcomes from my study to reach final conclusions. It also highlights potential improvement areas, including further testing and feature developments. Lastly, the future outlook for this research will be discussed at the end of this chapter.

## 5.1 RESULTS AND CONTRIBUTIONS

The following summary will show the main results and contributions to conclude the dissertation.

- Chapter 3 addressed the problem of real-time source detection in gamma-ray astronomy using anomaly detection. I developed a deep learning-based method for detecting anomalous time series data from the Cherenkov Telescope Array observations to achieve this goal. Efficient data processing pipelines have also been developed to produce (simulated) data sets, utility scripts, and notebooks. The source code can be found in [L.23].
- A statistical analysis pipeline to associate the predictions of the deep learning models with a statistical Gaussian $\sigma - level$ has been developed. Results are shown in Section 4.2. Since this analysis requires a large sample of simulation data, the code has been developed to be as efficient as possible.
- Chapter 4 evaluates the performance of the proposed anomaly detection method in the Gamma-Ray Bursts (GRBs) detection scenario for the Cherenkov Telescope Array. The method is compared against the Li&Ma technique in two scenarios of serendipitous discoveries and follow-up observations in two settings with short exposure times (5 sec and 1 sec). The assumption of a blind-search analysis to localize candidate sources in a fixed time of 10 seconds has been made. In the

short exposure time of 5 seconds scenario, the Li&Ma technique proves greater robustness, detecting most GRBs as shown in Figure 4.4 and Figure 4.6. However, in the serendipitous discovery scenario, the anomaly detection method proves, on average, faster in detecting GRBs compared to Li&Ma, as shown by detection delay metrics in Table 4.6. With a very short exposure time of 1 second, both techniques face limitations with reduced photon counts. However, the anomaly detection method with recurrent layers is more robust overall than Li&Ma and still performs faster detections on average, as shown by Figure 4.5 and Table 4.7.

- The method does not rely on the assumptions that the background and source models accurately represent the data. This would limit its ability to detect sources that do not conform to these models, especially challenging in the real-time context in which pipelines perform under degraded conditions. The autoencoder's inference is fast and well-suited for real-time analysis. Finally, the proposed method is not limited by the statistical independence requirement of the analysis bins nor by a minimum number of photon counts to perform analysis. For these reasons, the proposed method overcomes the limitation of the Full Field of View Maximum Likelihood and Li&Ma techniques.
- The nature of the proposed method is flexible enough to allow different analysis settings, with different exposure times and temporal bin time spans, to allow real-time search for transient events on multiple time scales.
- This study significantly contributes to astrophysics by demonstrating the effectiveness of deep learning-based anomaly detection techniques for real-time source detection in gamma-ray astronomy.

## 5.2 OUTLOOK AND FUTURE WORK

The proposed method has shown promising results, but there is room for improvement. Additional implementation and examination could enhance the understanding of the estimated performances and refine the method.

- The method should be extensively tested, relaxing the assumptions made. Concerning the blind-search analysis, we can detect multiple candidates' regions of interest at different times with different significance levels in the real-world scenario. The time delays in the context of the follow-up observation use case have been assumed fixed. A more in-depth study with a randomized time delay (or something more similar to the current typical GW alert latency) would improve the estimate of the method's performance.
- Another future improvement is to evaluate the visibility of the GRB events. In this study, the information on the spatial localization of the events was not possi-

ble to infer since the trigger times, i.e., the timestamps associated with the start of each GRB event, were not available. Hence, the same sky coordinates have been considered for each GRB simulation. Consequently, a single instrument response function was sufficient because only one background level was considered. Access to the trigger times would allow visibility studies and require different IRFs, introducing changes in the observing condition.

- It is important to evaluate the post-trial probability to provide a more robust estimate of the method's performance.
- It is crucial to test the method on real data and address the non-stationary settings of online observations.
- Testing the proposed method on multiple time scales with more transient phenomena other than GRBs would further demonstrate the versatility and effectiveness of the method.
- The system can be integrated within the Science Alert Generation system and deployed on the onsite computing cluster at La Palma. This would provide valuable insights into the method's performance in a real-world setting and be another valuable tool for discovering new transient events in real time.

# 6

# APPENDIX A

## 6.1 DETECTIONS

This chapter shows several examples of time series generated from the photon lists simulations of different GRB templates in the short-exposure and very short-exposure scenarios. The corresponding *significance variability* plots are shown for each simulated trial. The following analysis results have been selected from the 419 trials that defined the test set. They have been divided into four groups, extracting four representatives for each group:

- The first category shows some examples in which the anomaly detection analysis performs a $5\sigma$ detection, while the Li&Ma technique couldn't. The trials under consideration have been simulated from the following templates: *"run*0050_*ID*000319", *"run*0276_*ID*000042", *"run*0481_*ID*000069", *"run*0522_*ID*000294".

- The second category shows some examples in which the anomaly detection analysis agrees well with the Li&Ma technique. The trials under consideration have been simulated from the following templates: *"run*0266_*ID*000397", *"run*0051_*ID*000166", *"run*0231_*ID*000152", *"run*0170_*ID*000418".

- The third category shows some examples in which the anomaly detection analysis performs worst than the Li&Ma technique and cannot manage to perform detection. The trials under consideration have been simulated from the following templates: *"run*0352_*ID*000076", *"run*0421_*ID*000208", *"run*0428_*ID*000157", *"run*0615_*ID*000245".

- The fourth category shows some examples of the analysis of very strong GRBs. The trials under consideration have been simulated from the following templates: *"run*0535_*ID*000181", *"run*0222_*ID*000318", *"run*0340_*ID*000060", *"run*0453_*ID*000359".

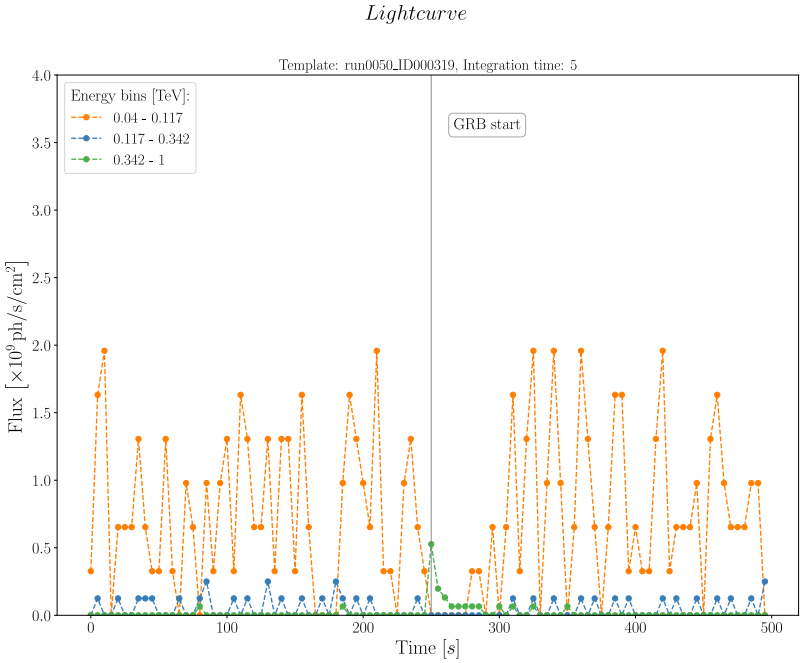


Figure 6.1: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

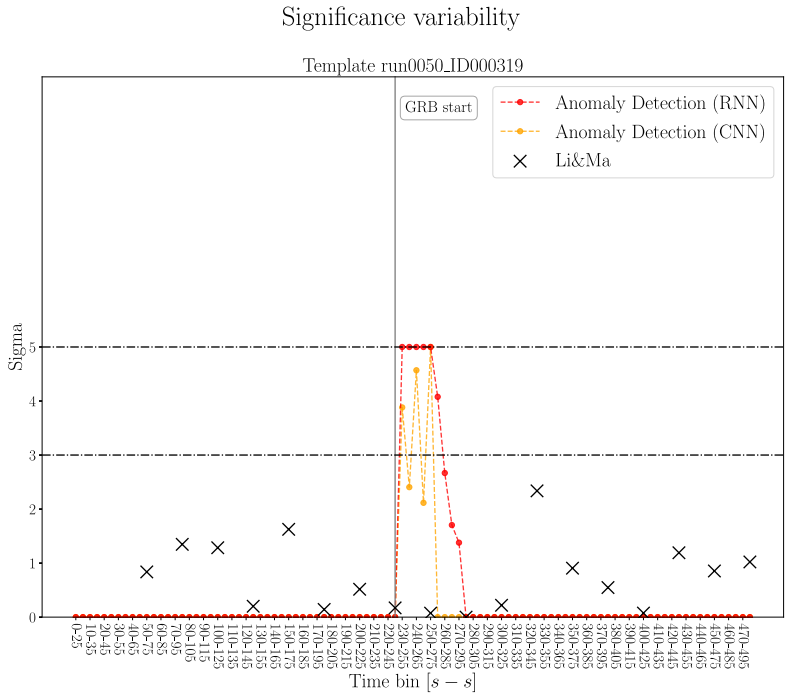


Figure 6.2: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

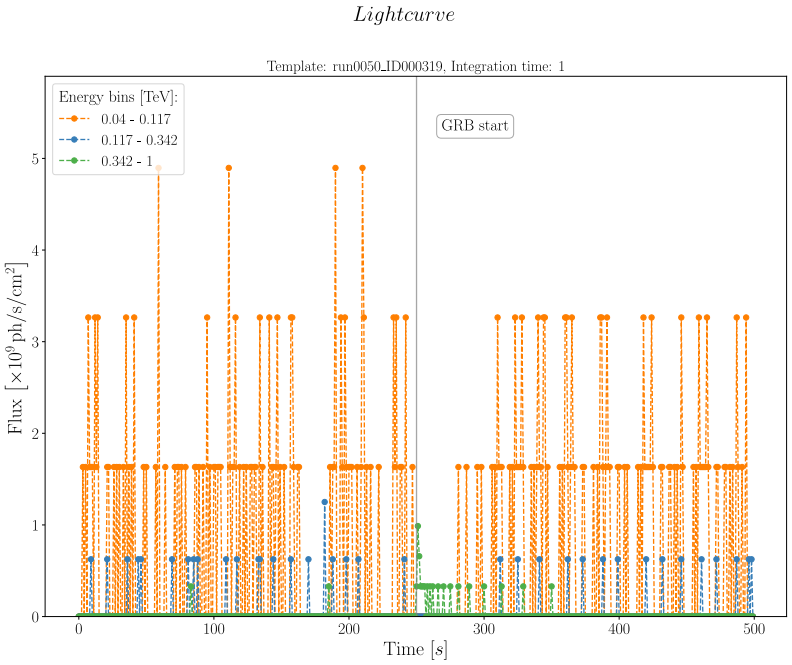


Figure 6.3: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

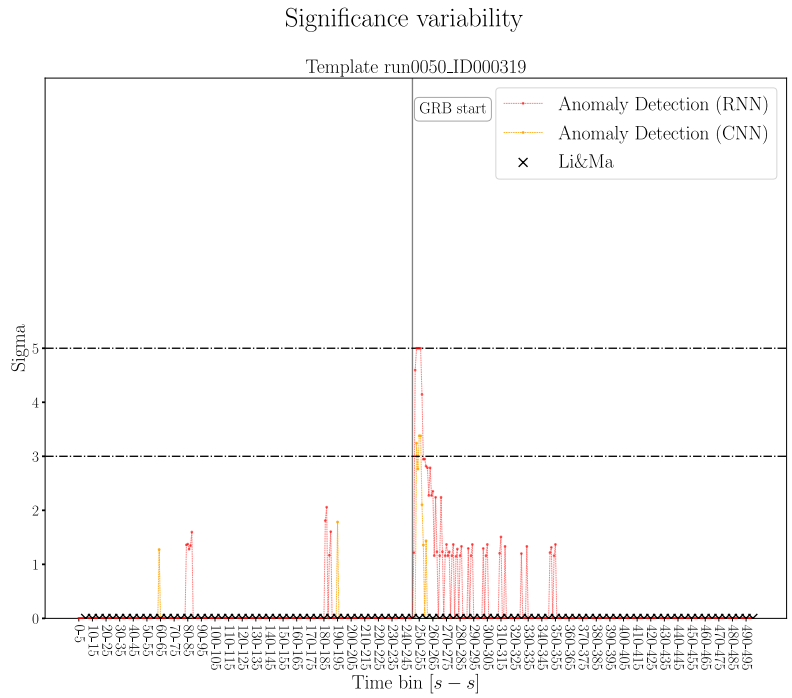


Figure 6.4: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

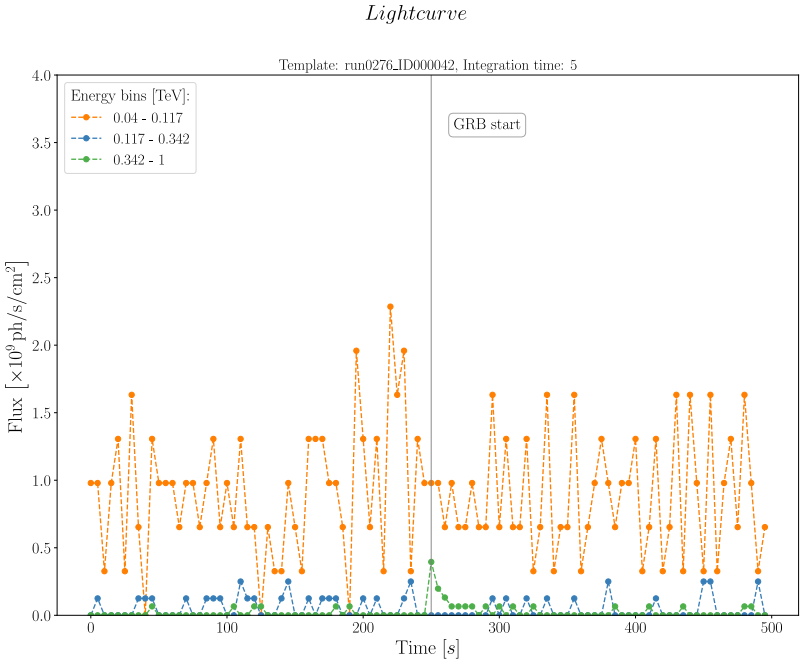


Figure 6.5: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

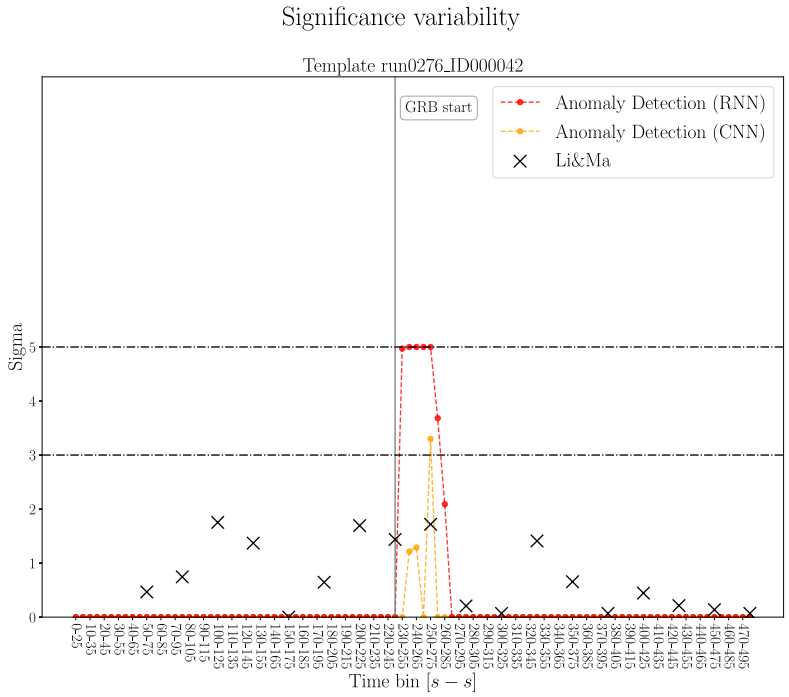


Figure 6.6: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

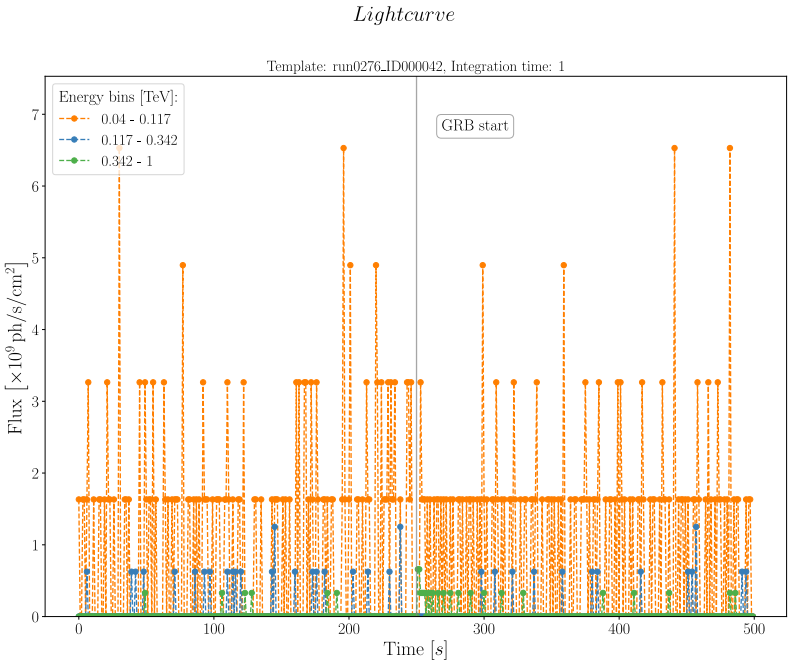


Figure 6.7: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

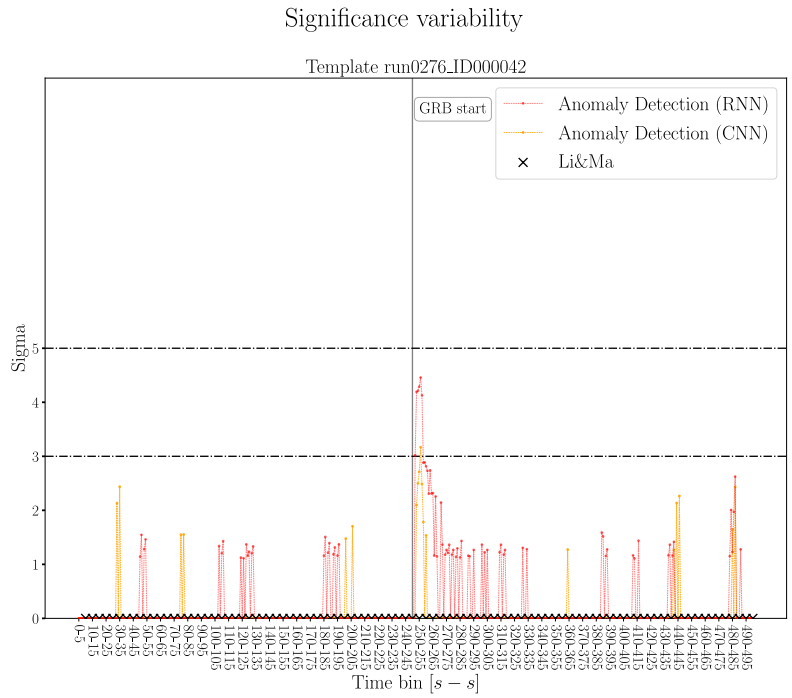


Figure 6.8: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

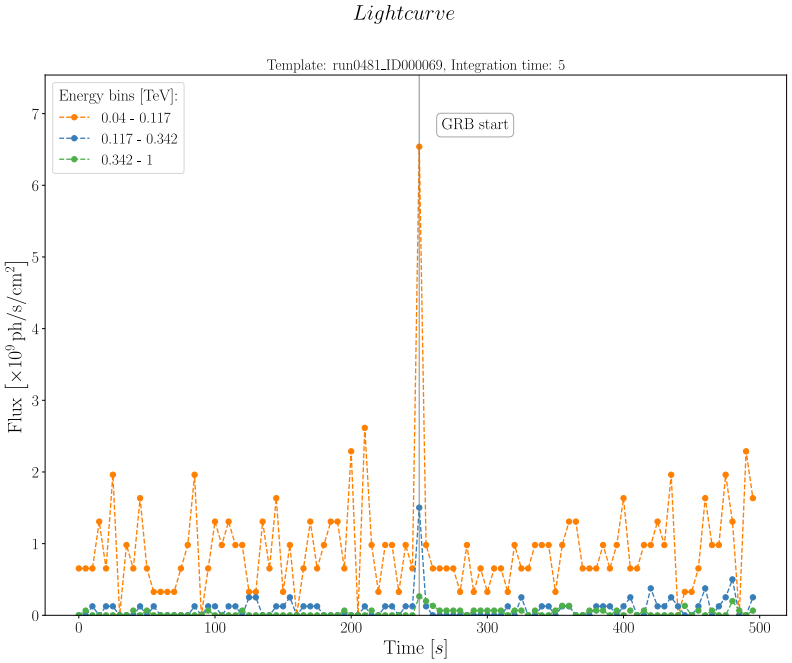


Figure 6.9: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

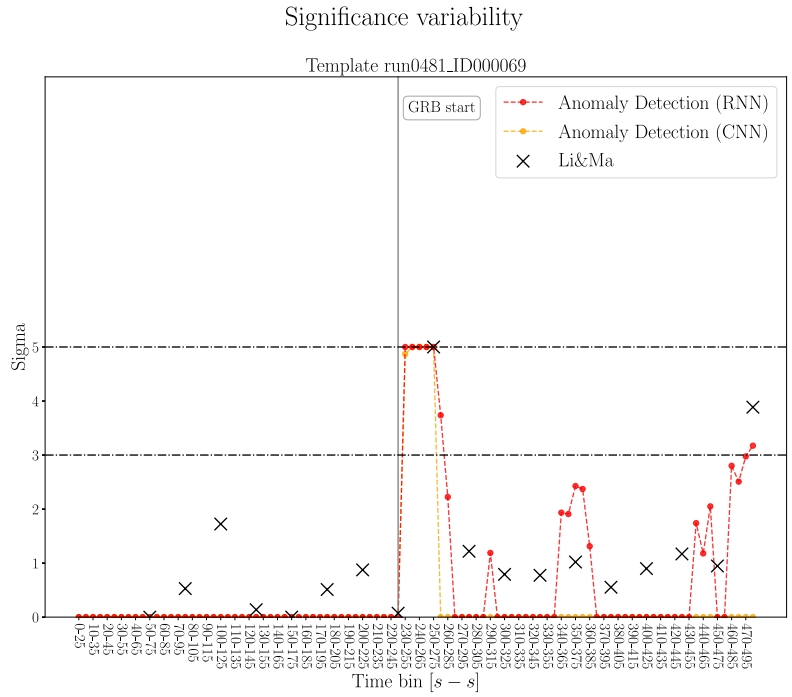


Figure 6.10: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

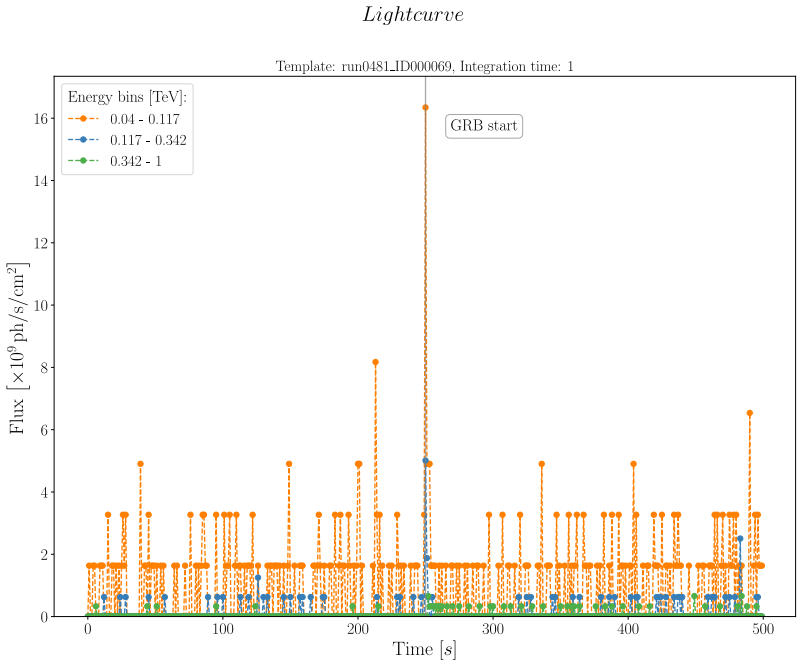


Figure 6.11: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

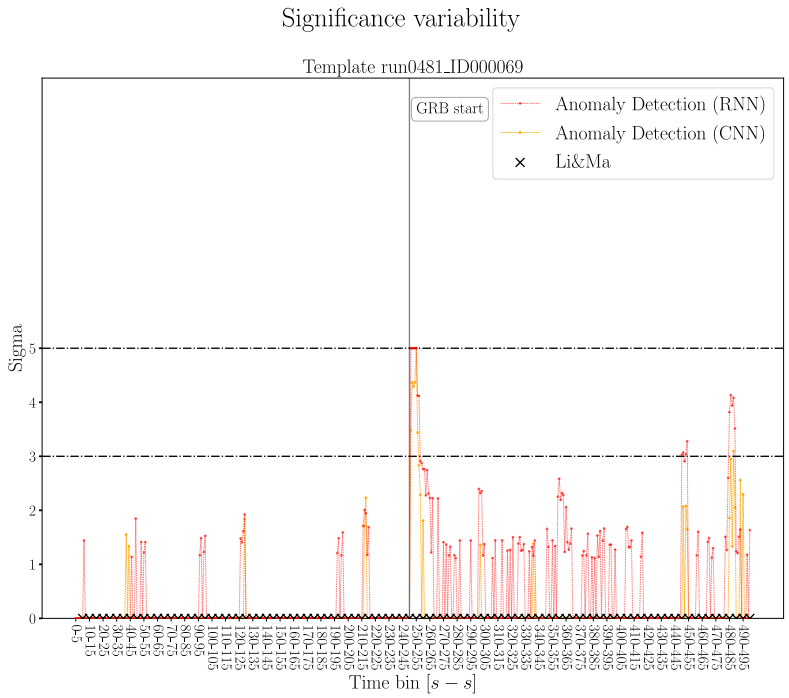


Figure 6.12: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

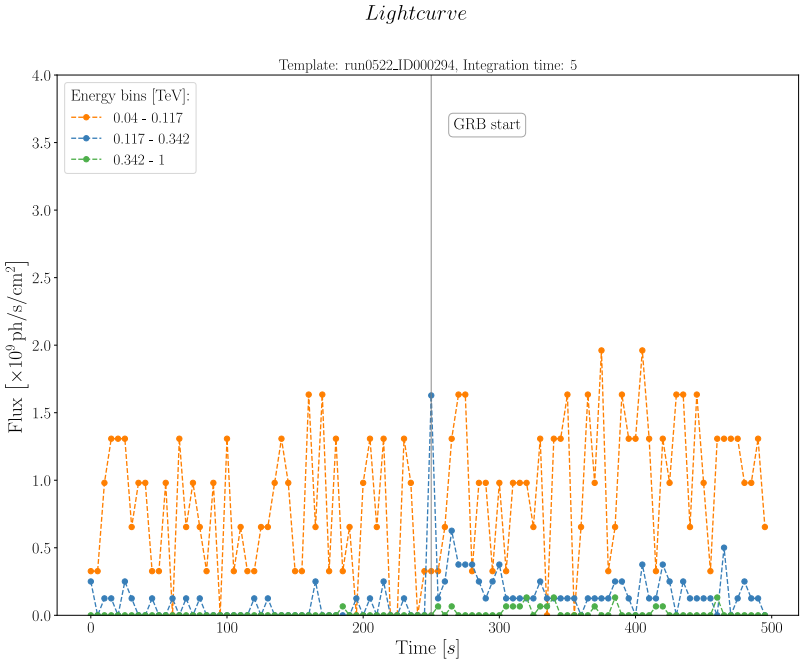


Figure 6.13: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

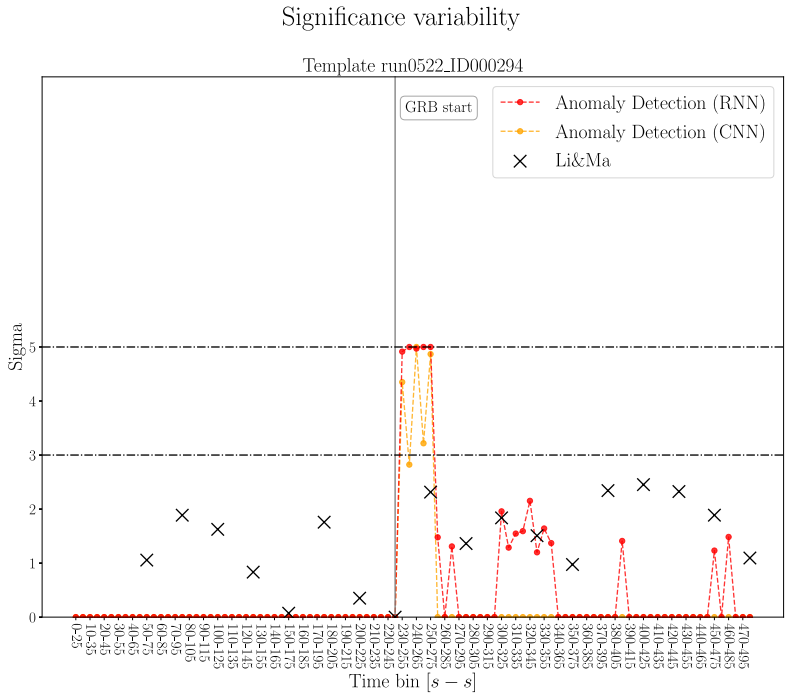


Figure 6.14: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

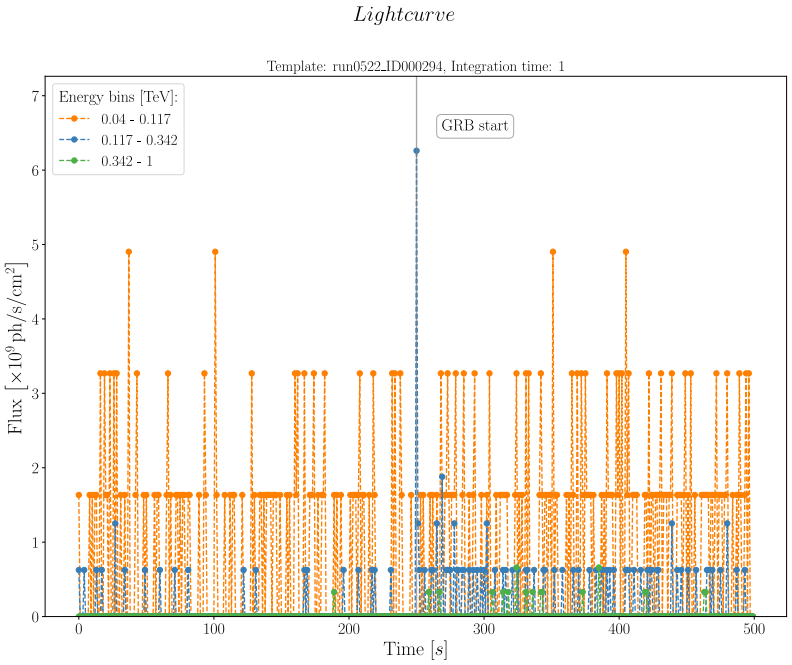


Figure 6.15: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

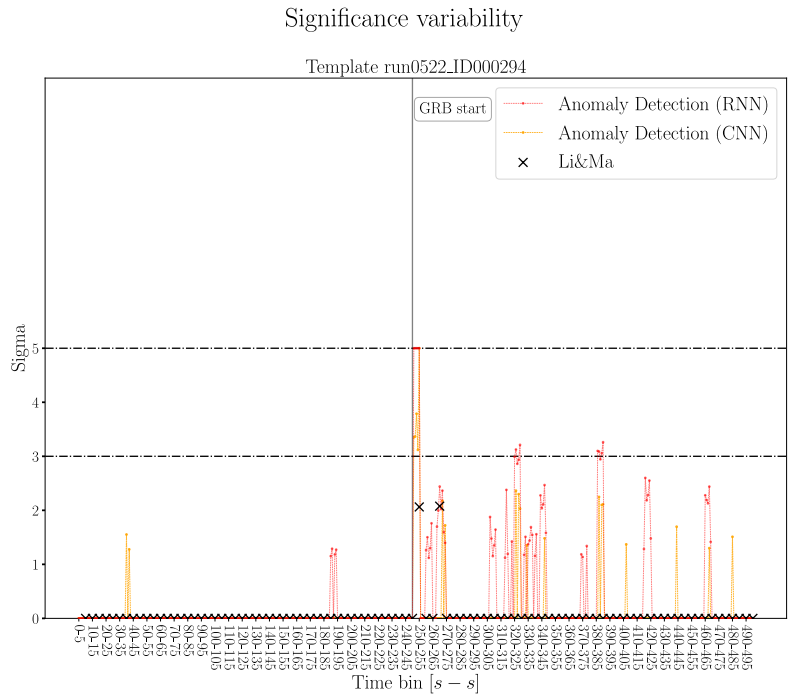


Figure 6.16: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

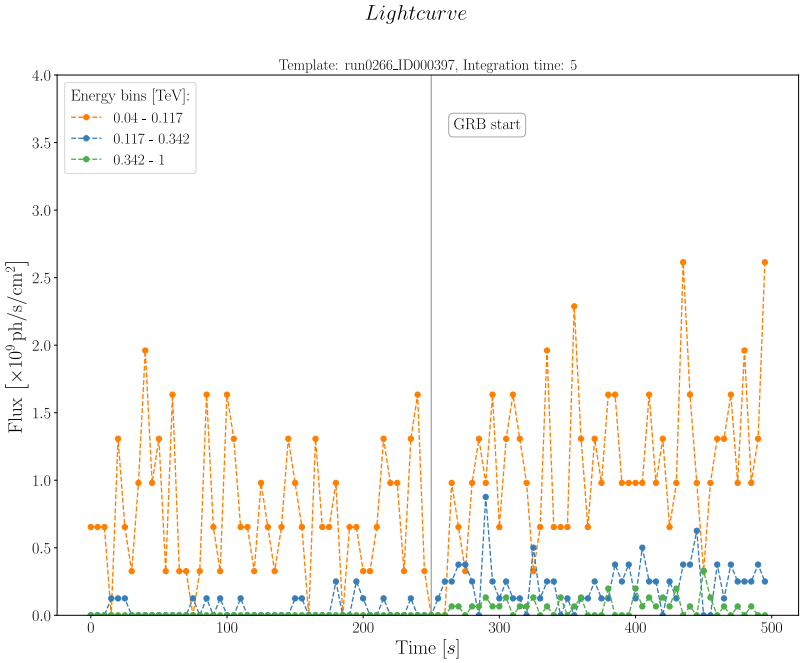


Figure 6.17: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

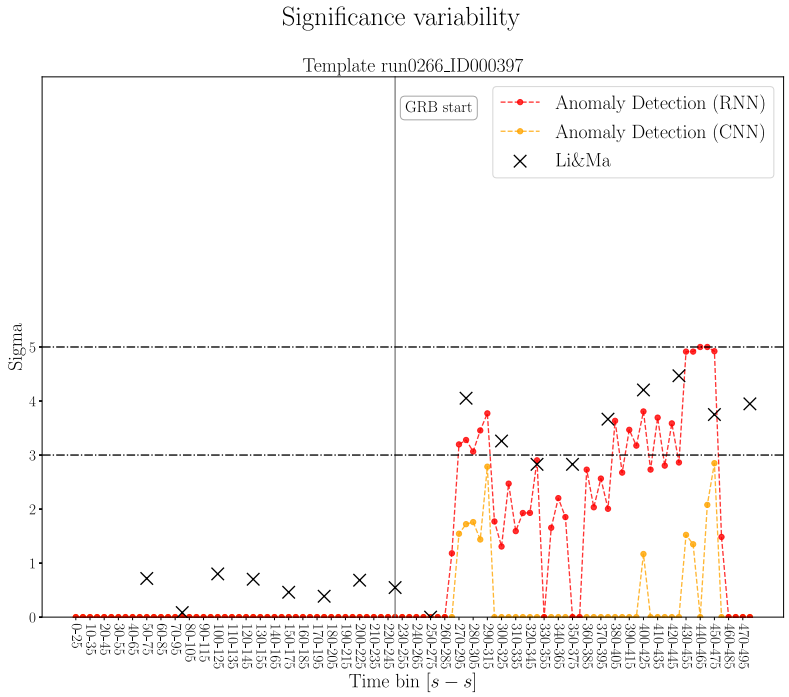


Figure 6.18: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

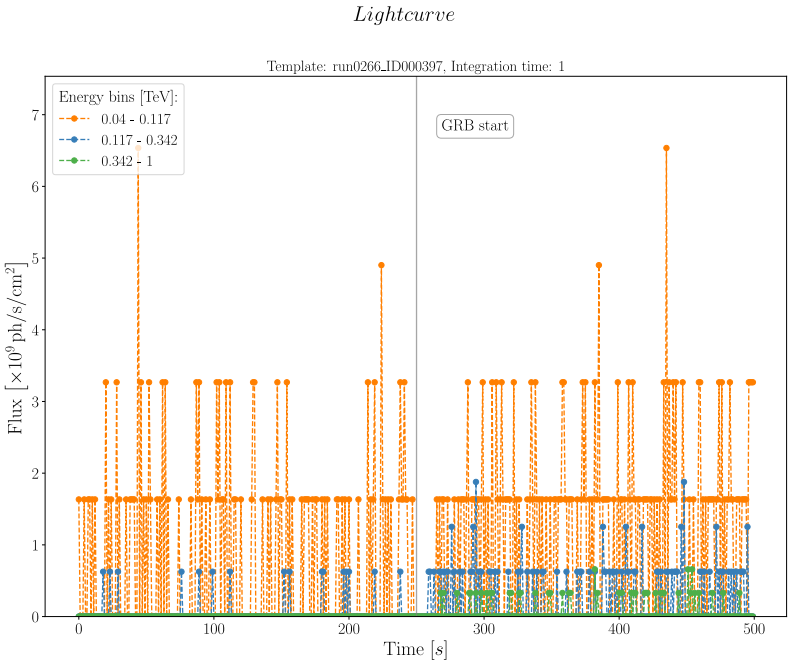


Figure 6.19: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).


Figure 6.20: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

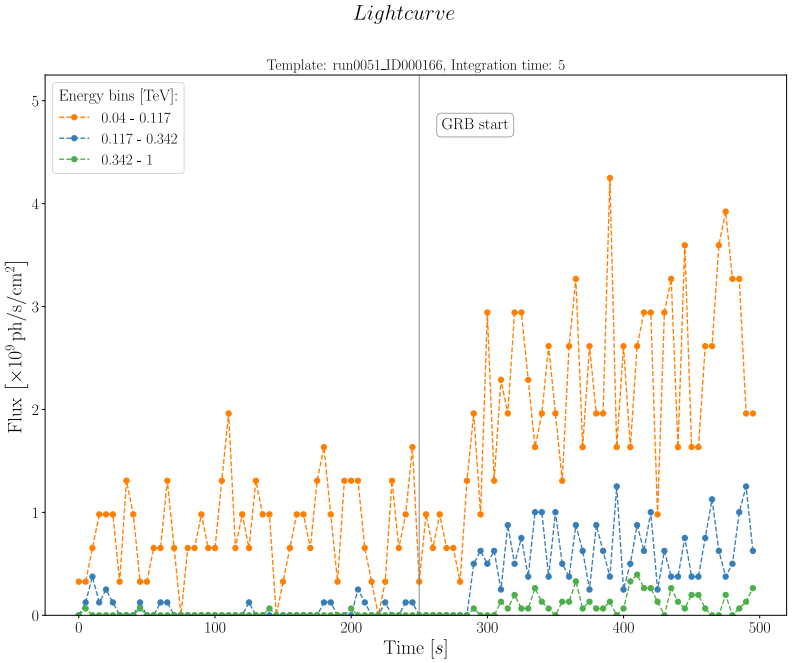


Figure 6.21: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

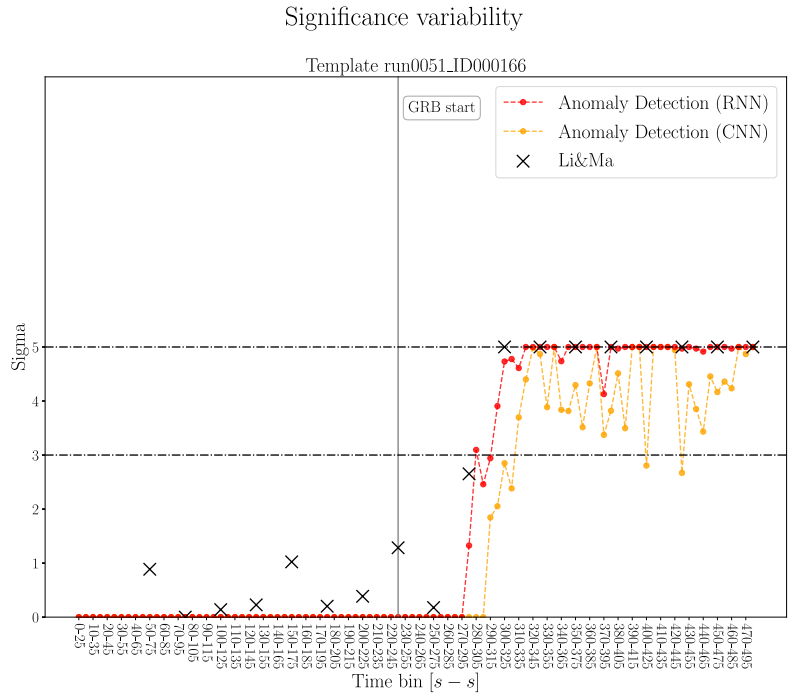


Figure 6.22: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

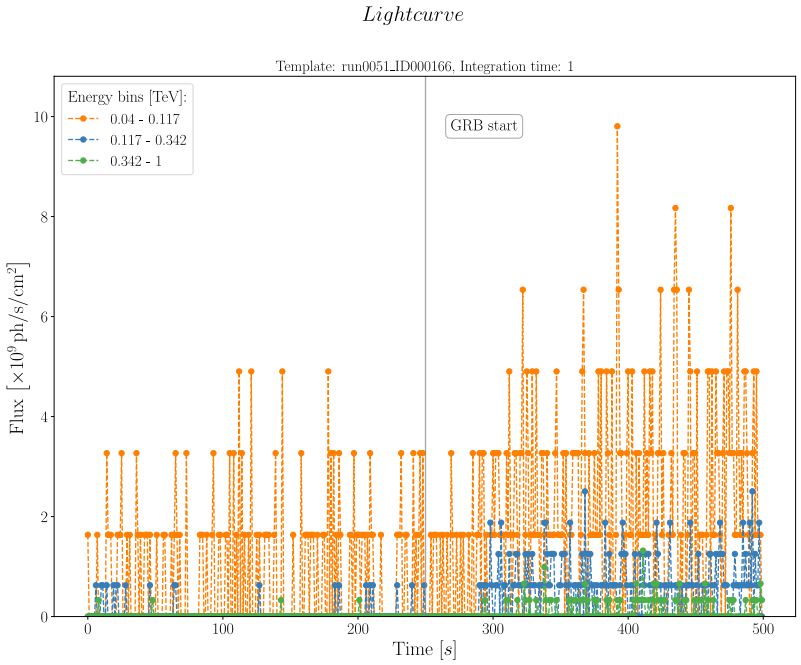


Figure 6.23: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

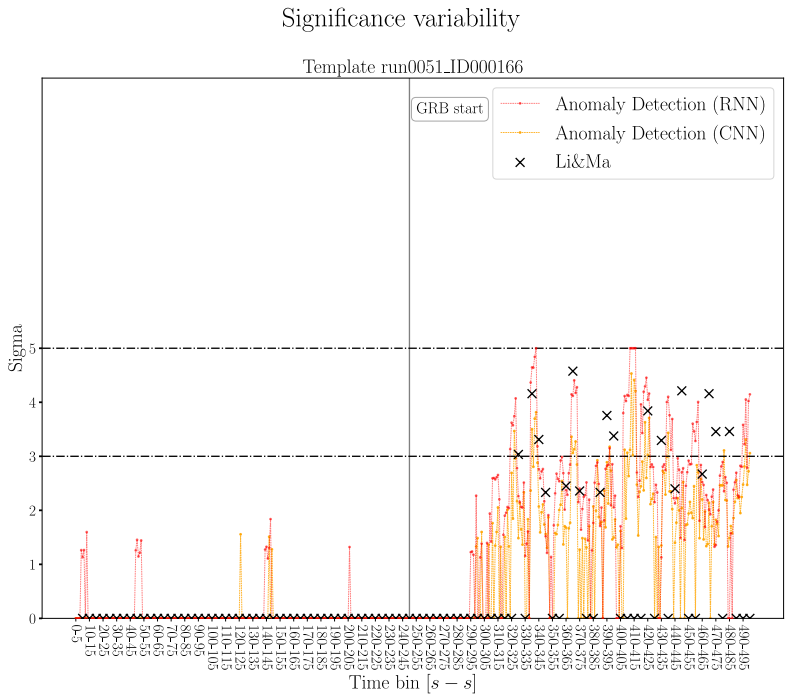


Figure 6.24: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

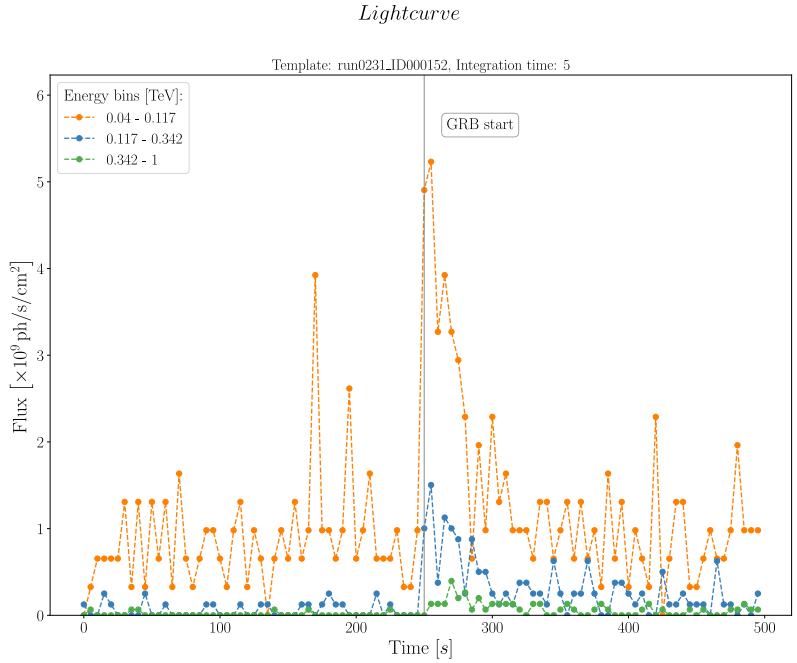


Figure 6.25: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

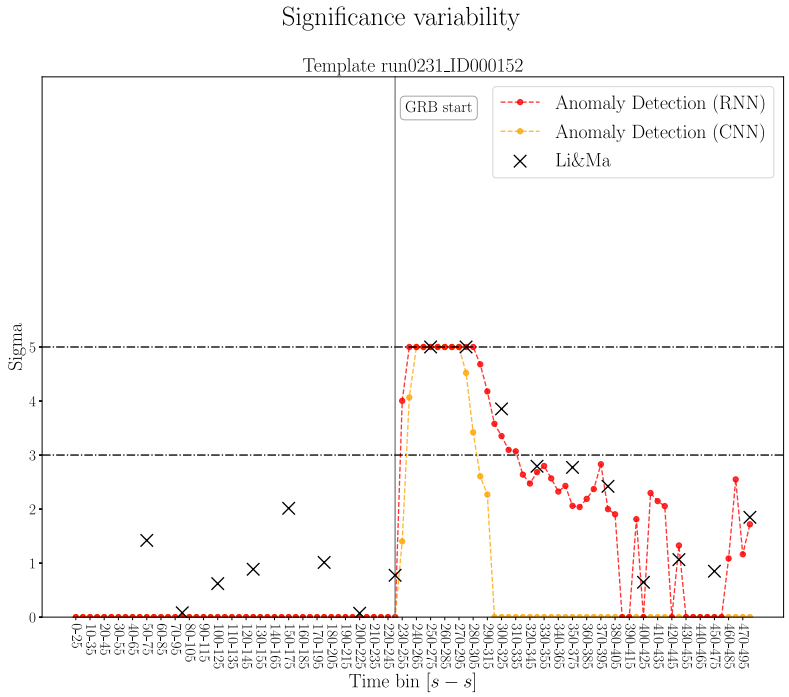


Figure 6.26: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

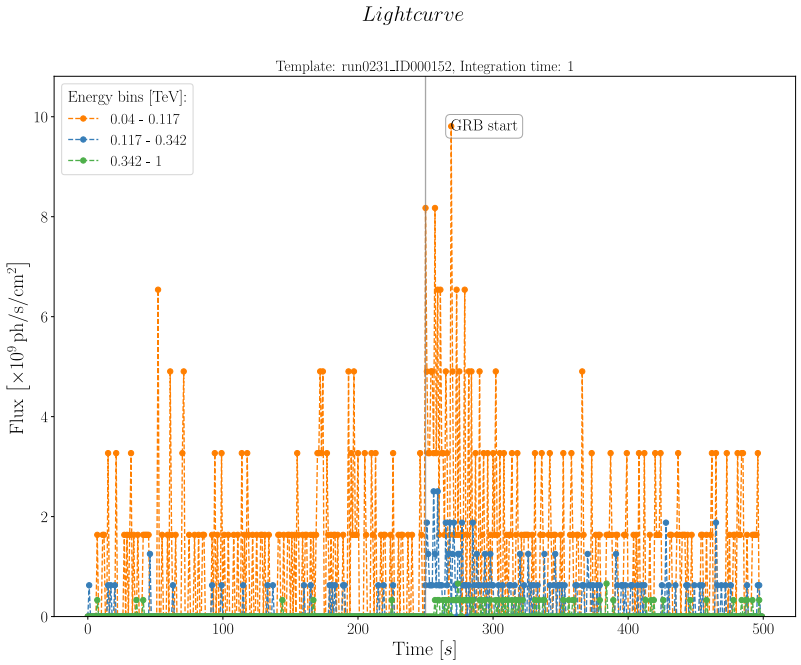


Figure 6.27: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

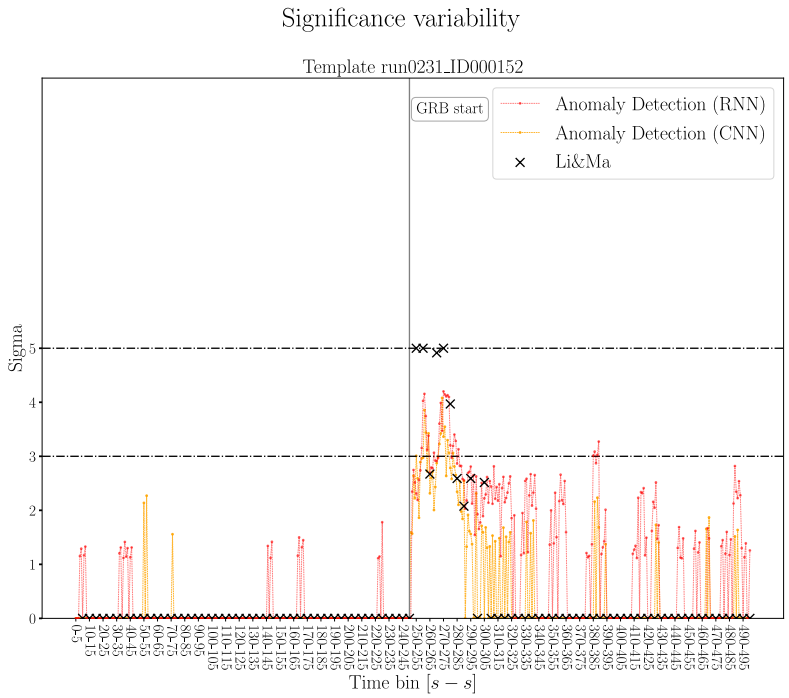


Figure 6.28: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

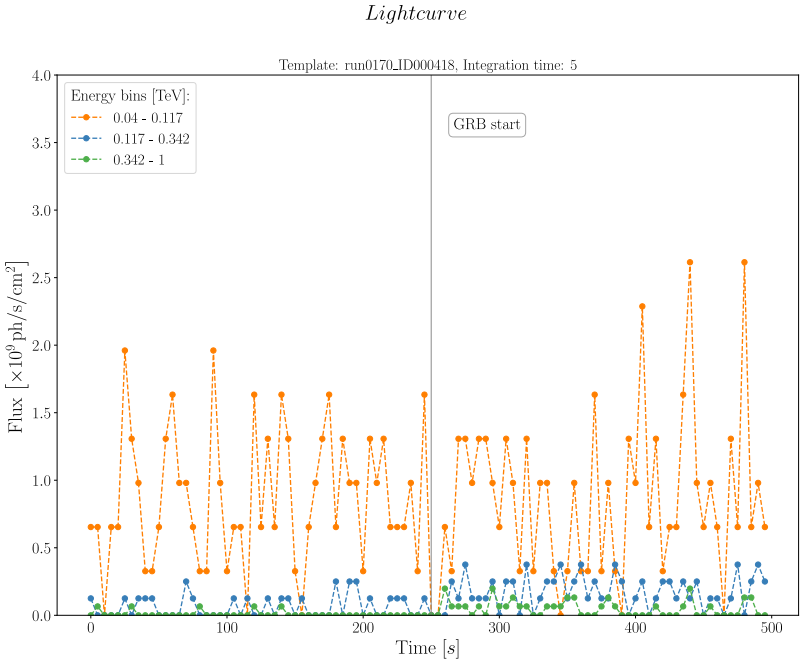


Figure 6.29: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

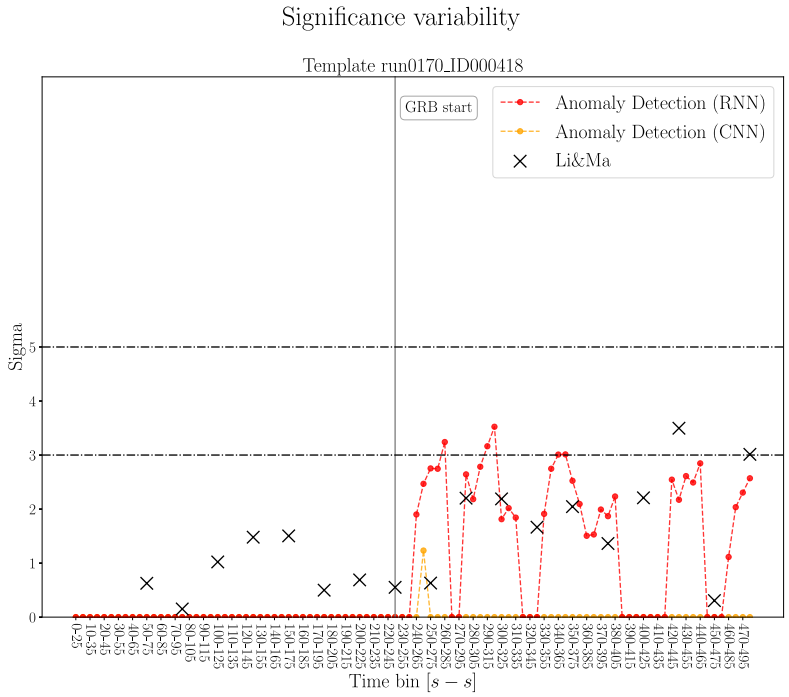


Figure 6.30: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

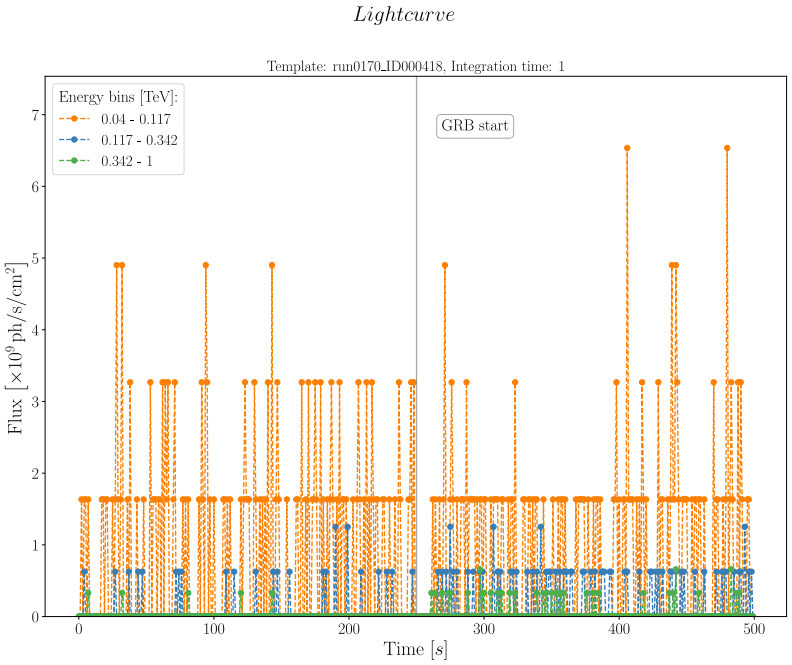


Figure 6.31: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

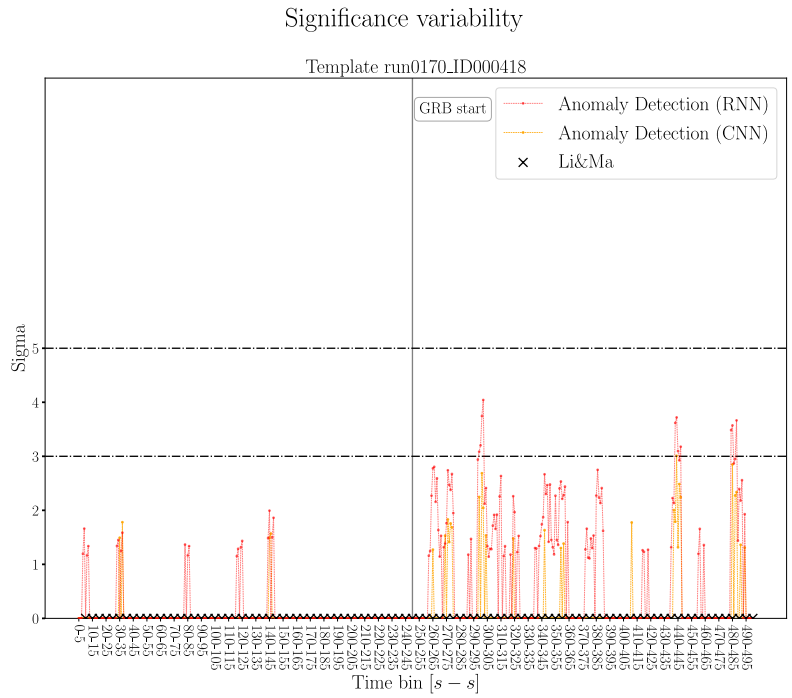


Figure 6.32: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

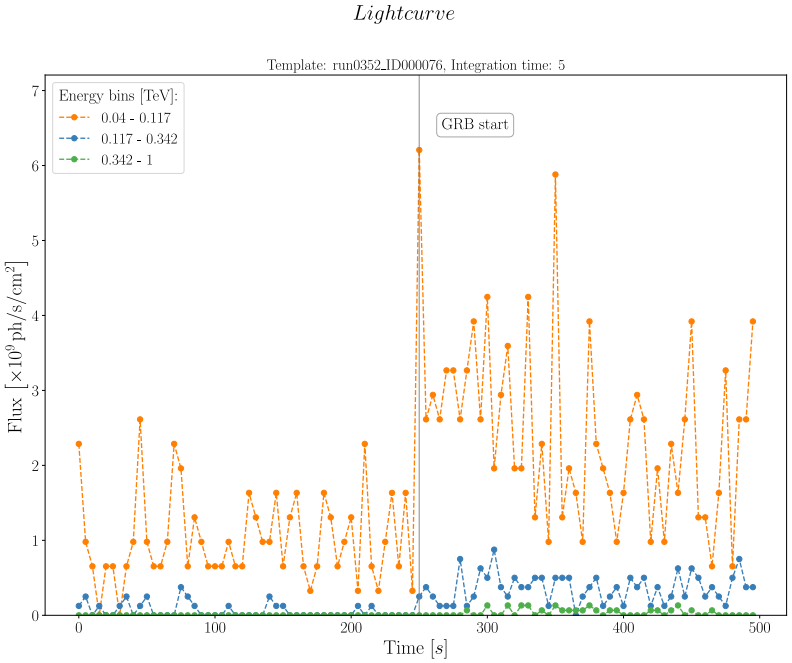


Figure 6.33: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

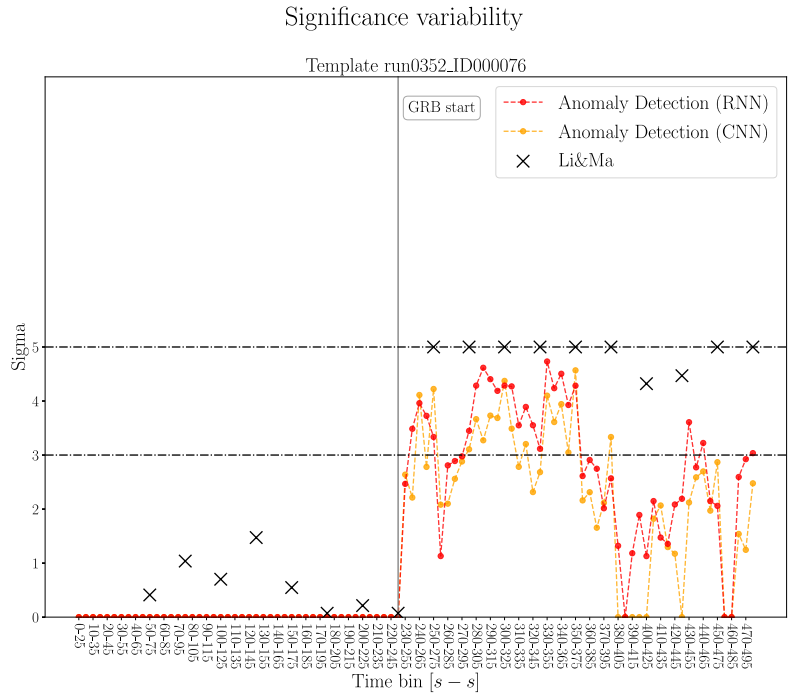


Figure 6.34: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

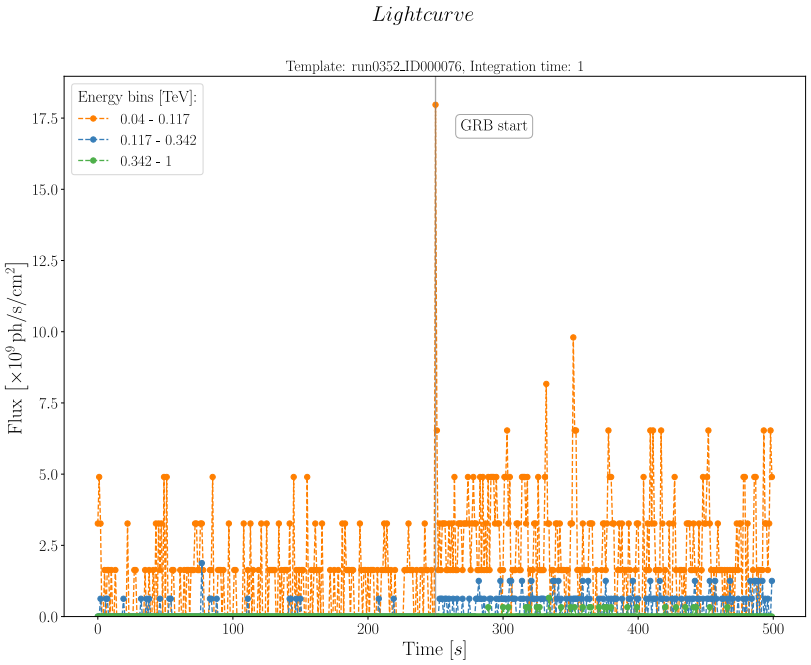


Figure 6.35: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

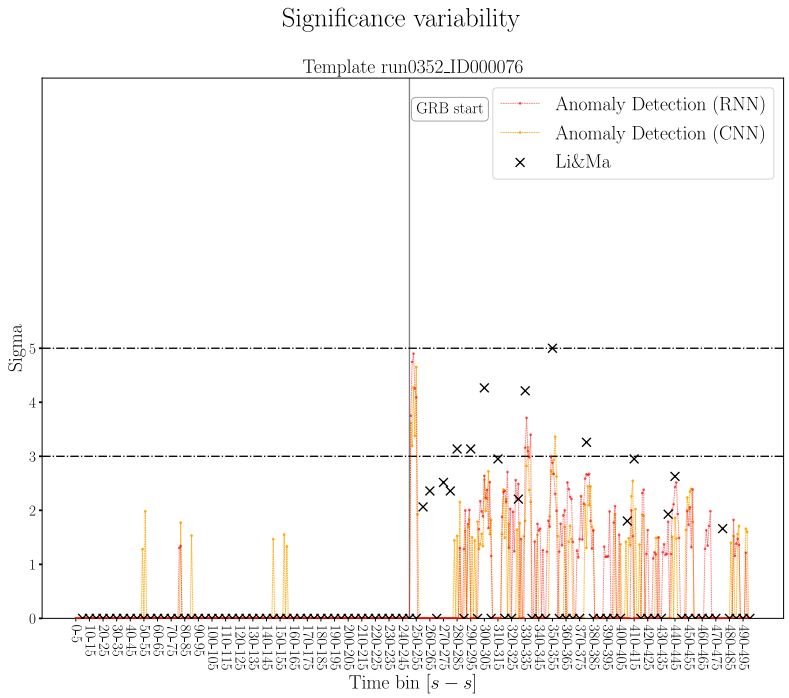


Figure 6.36: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

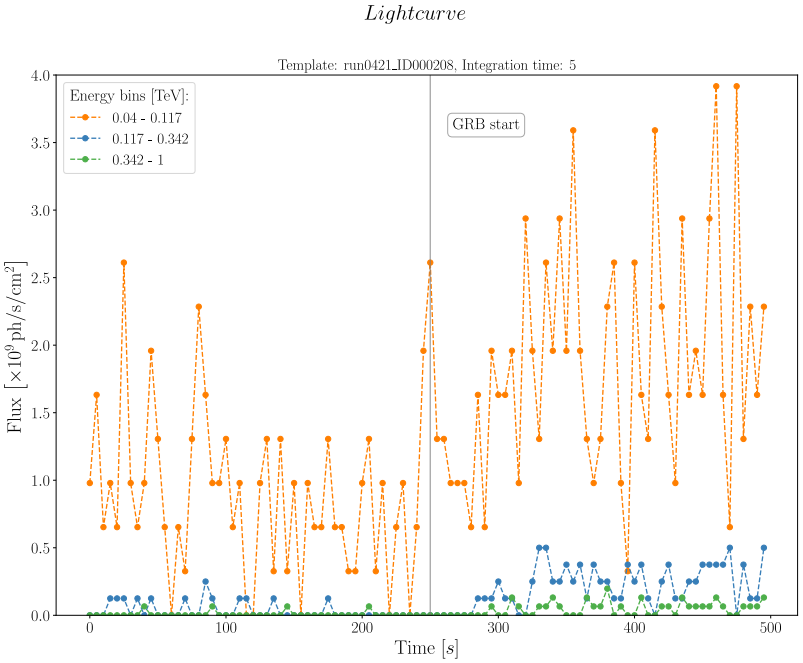


Figure 6.37: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

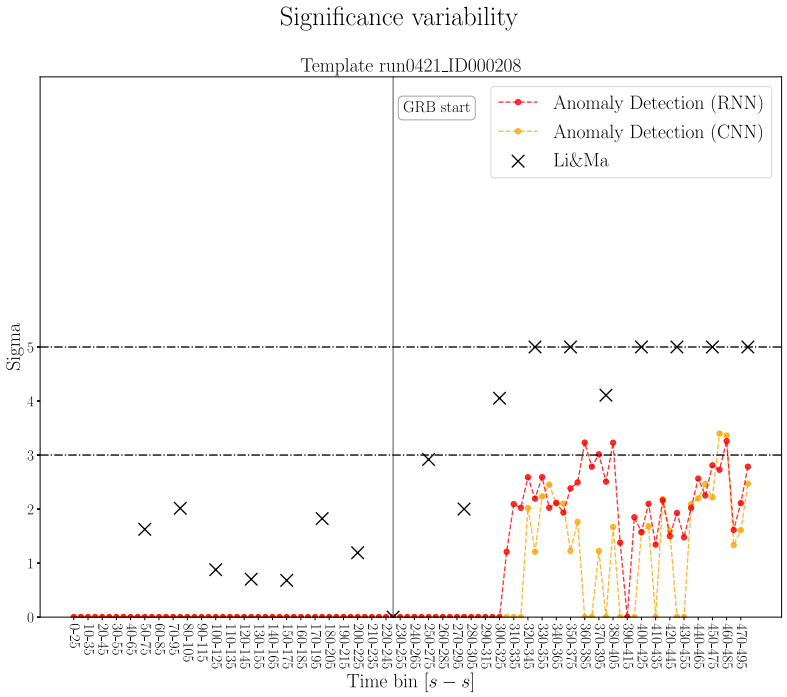


Figure 6.38: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

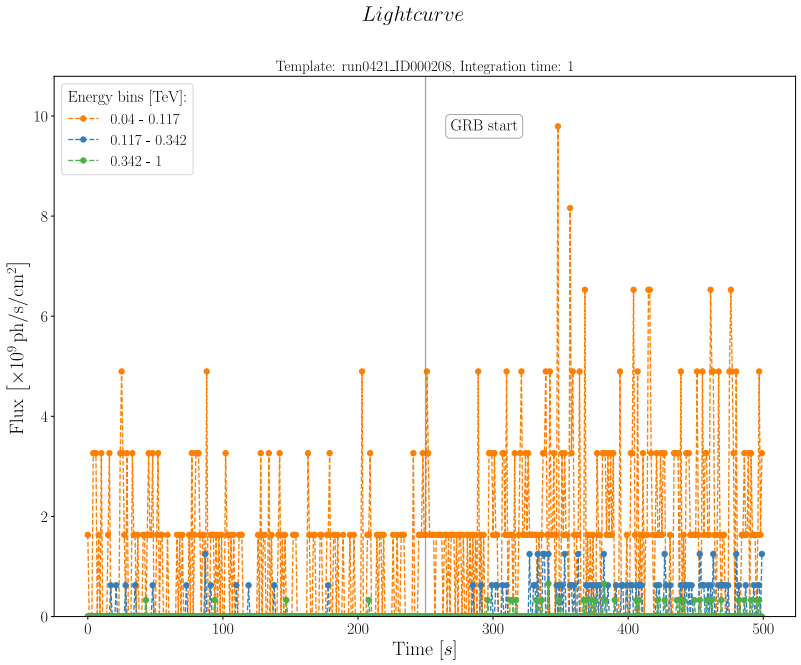


Figure 6.39: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

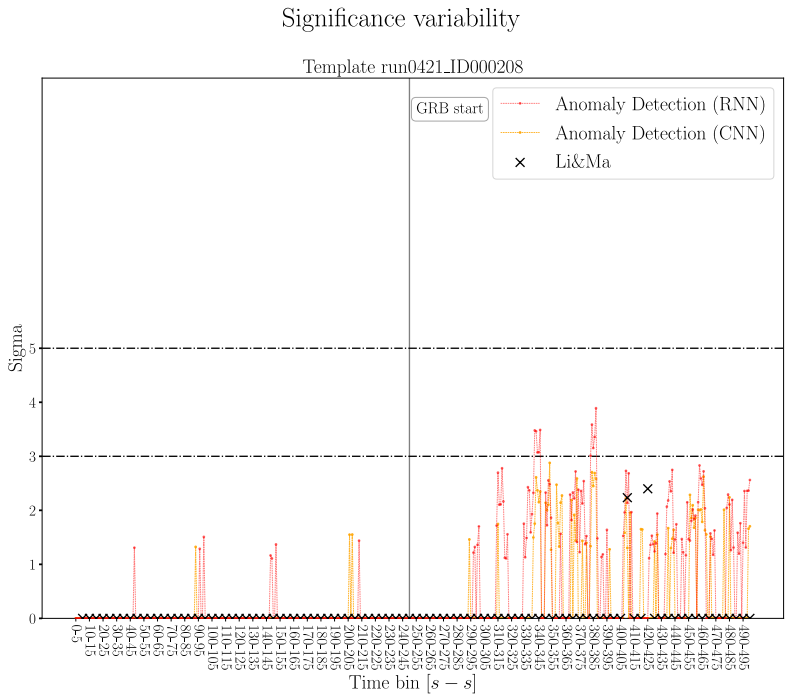


Figure 6.40: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

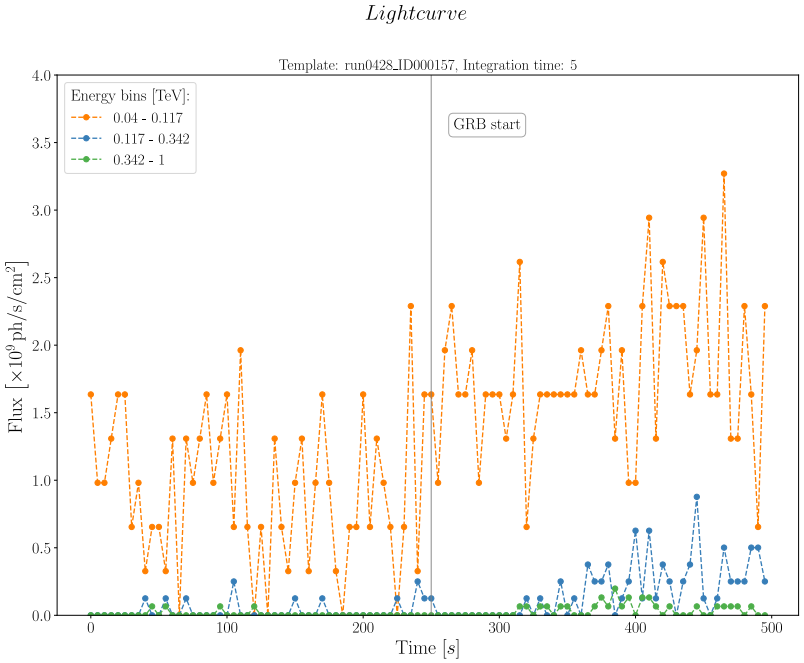


Figure 6.41: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

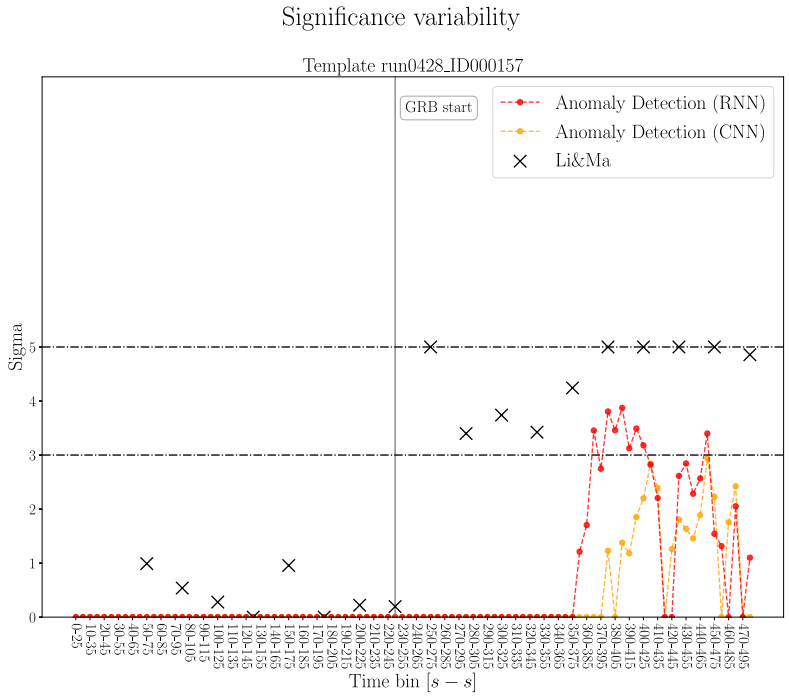


Figure 6.42: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

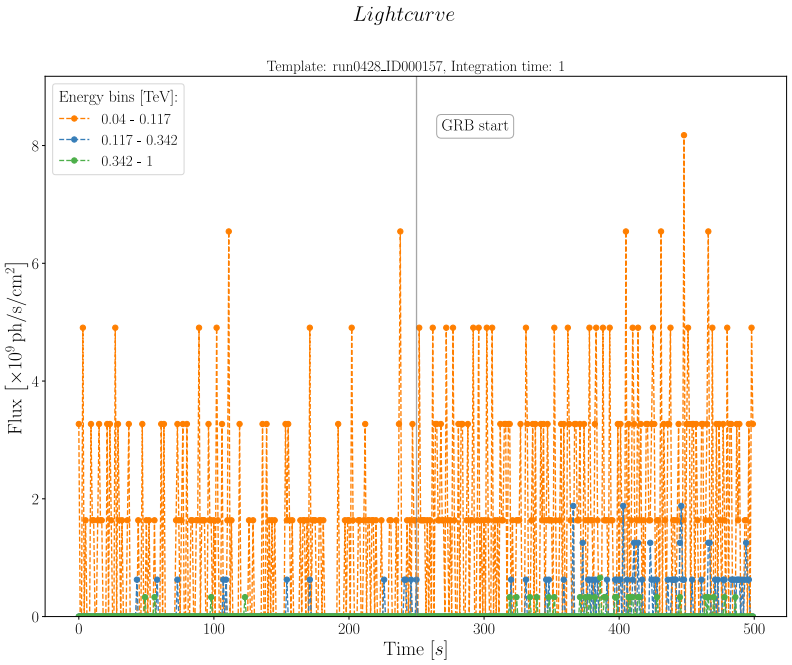


Figure 6.43: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

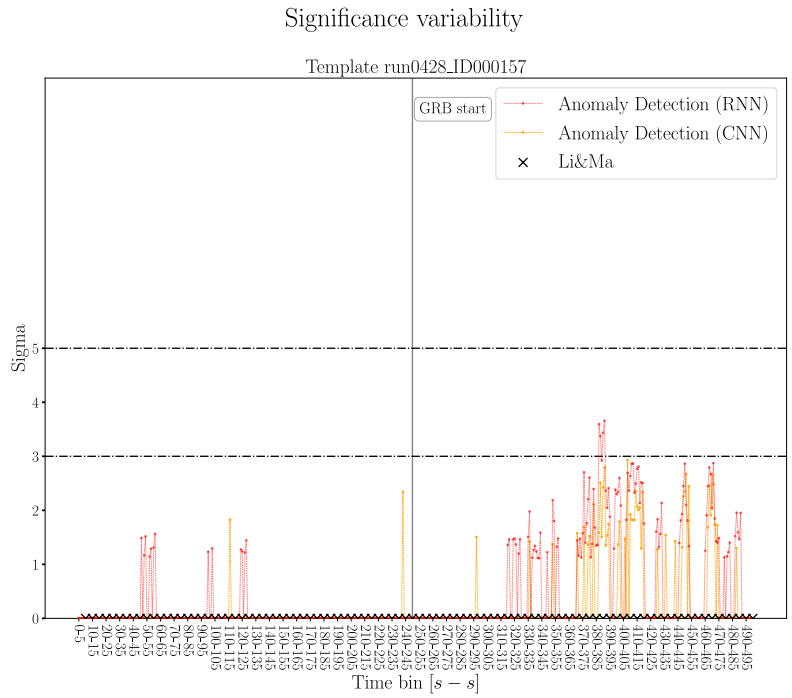


Figure 6.44: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

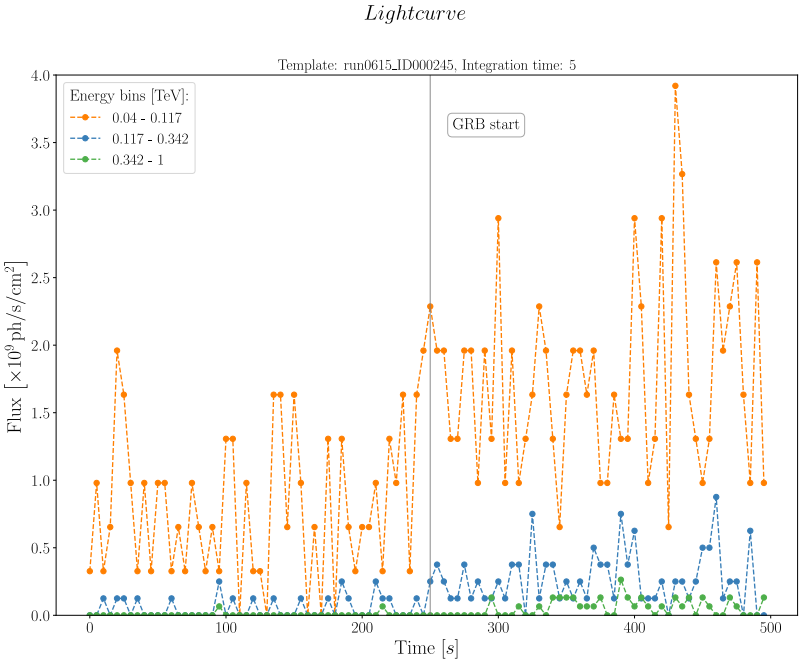


Figure 6.45: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

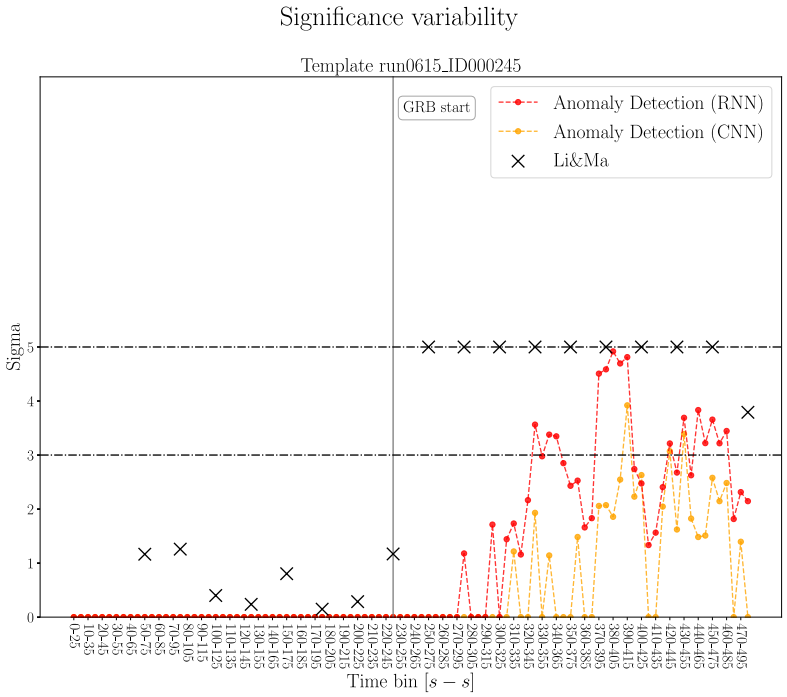


Figure 6.46: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

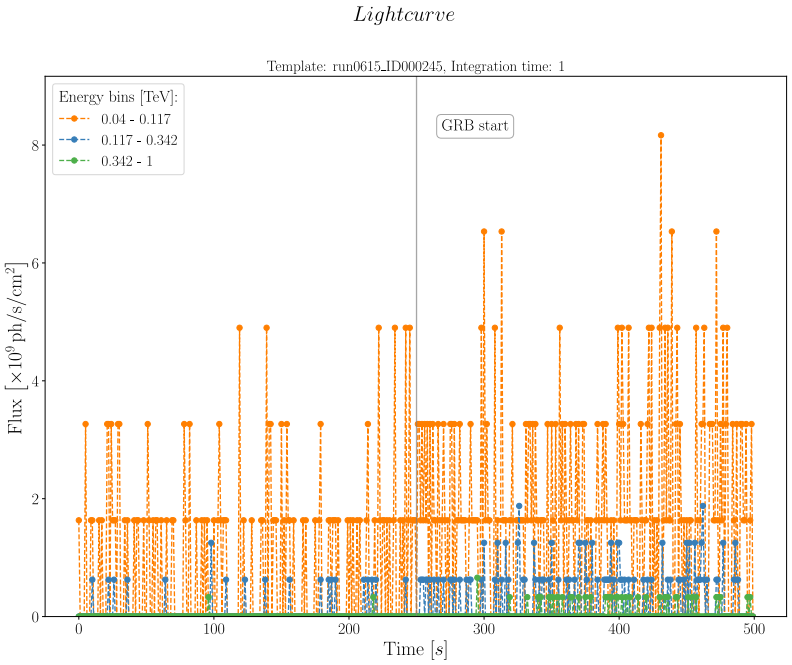


Figure 6.47: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

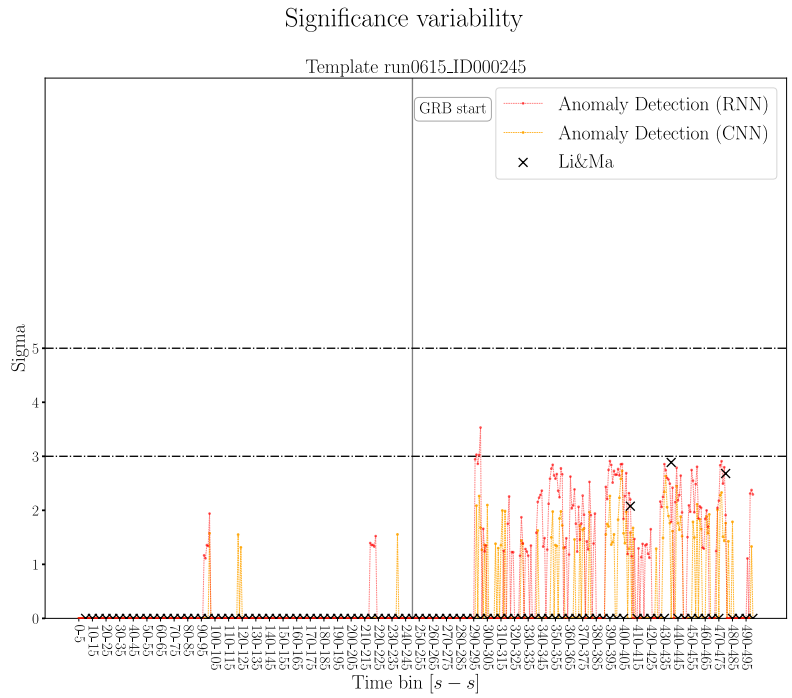


Figure 6.48: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

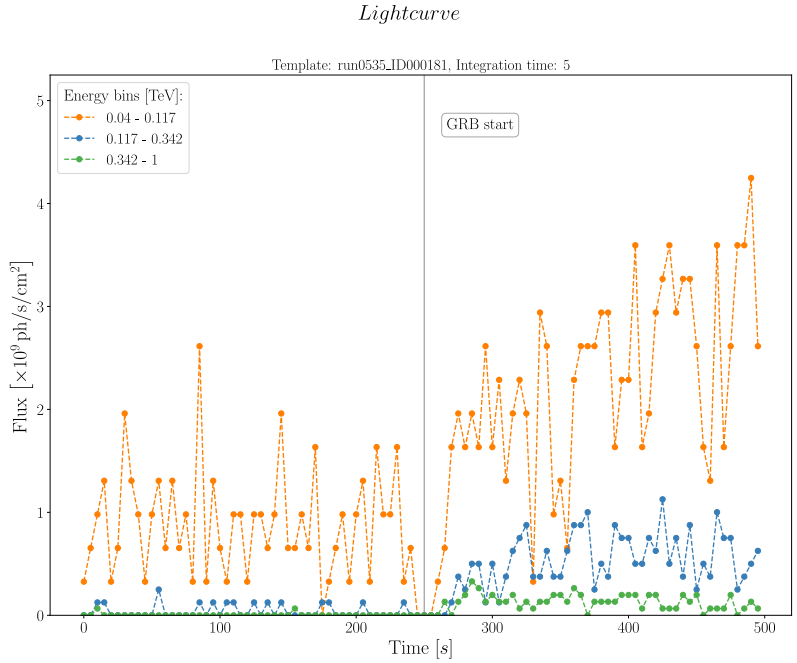


Figure 6.49: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).


Figure 6.50: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

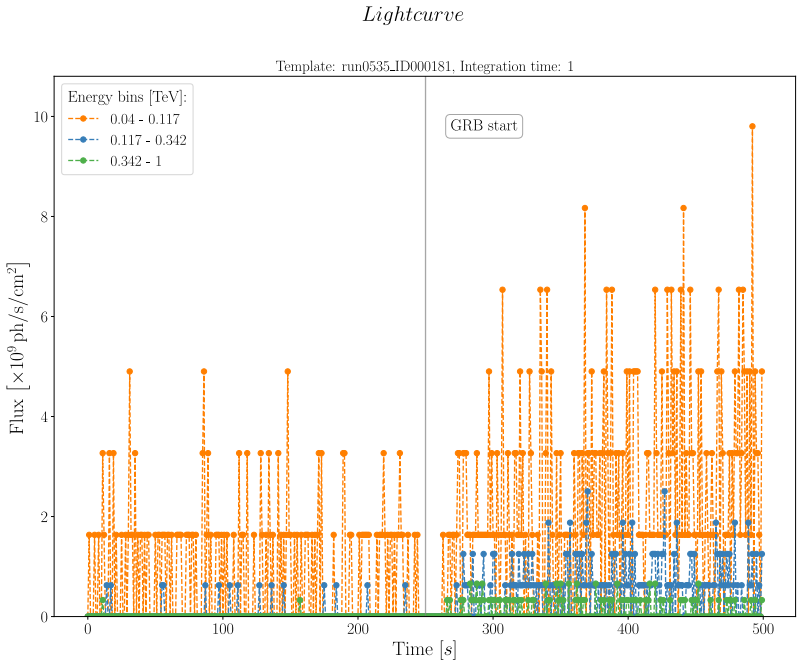


Figure 6.51: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

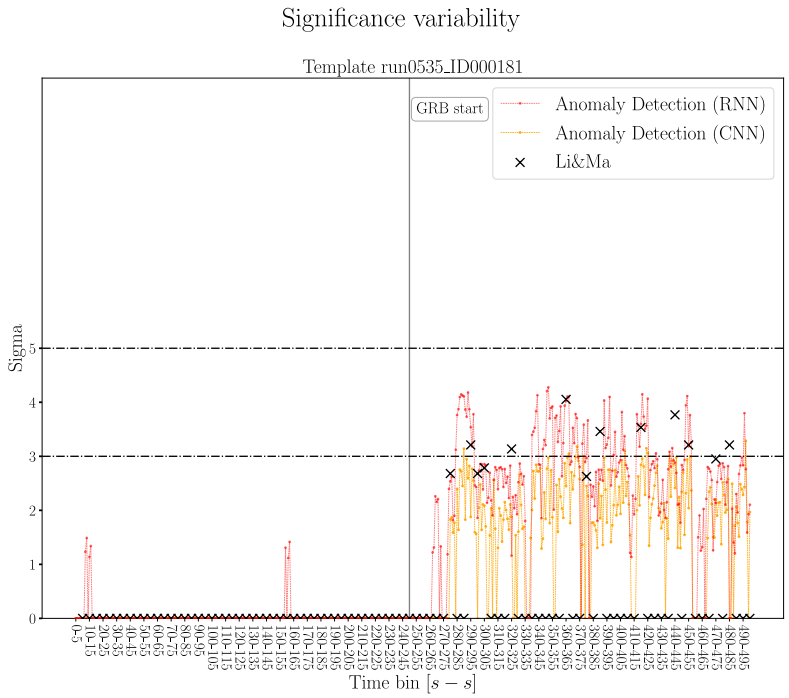


Figure 6.52: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

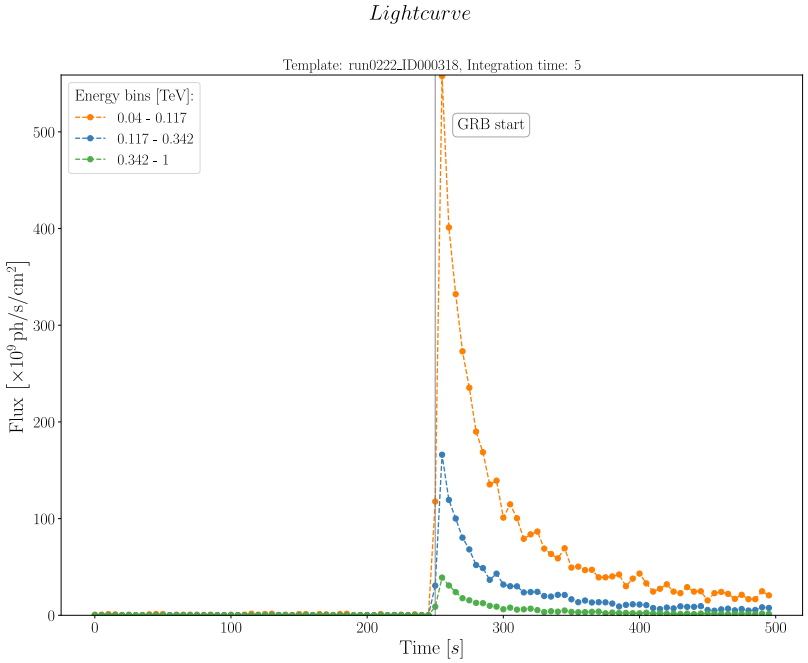


Figure 6.53: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

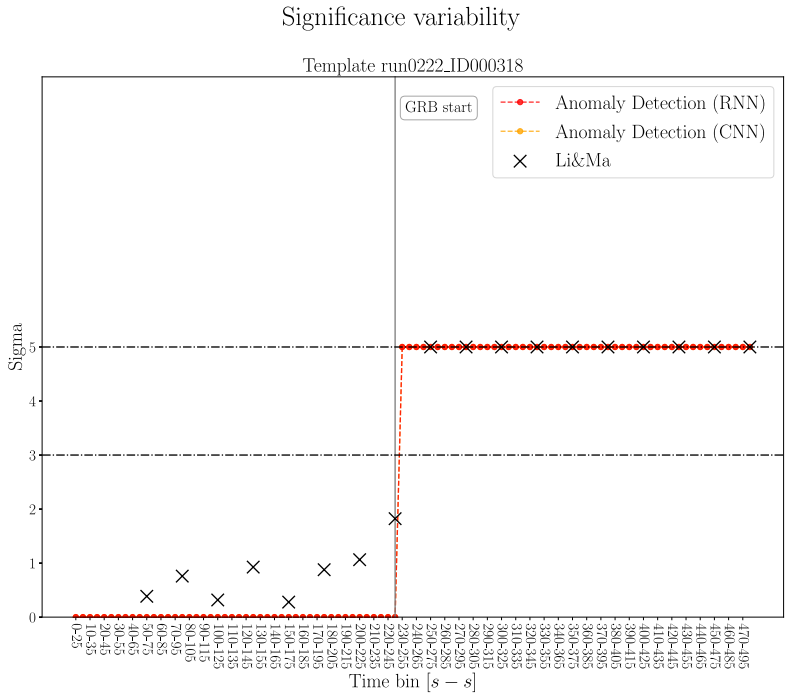


Figure 6.54: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

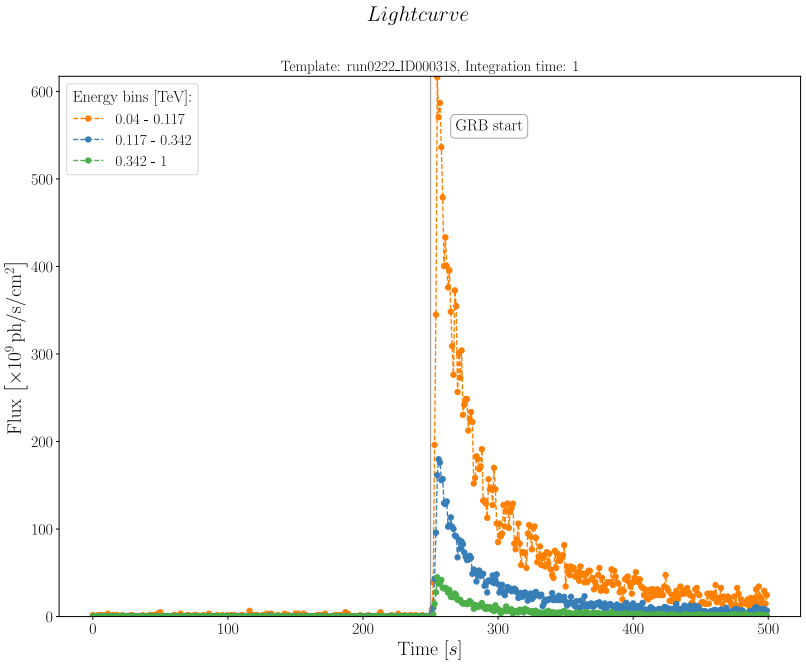


Figure 6.55: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

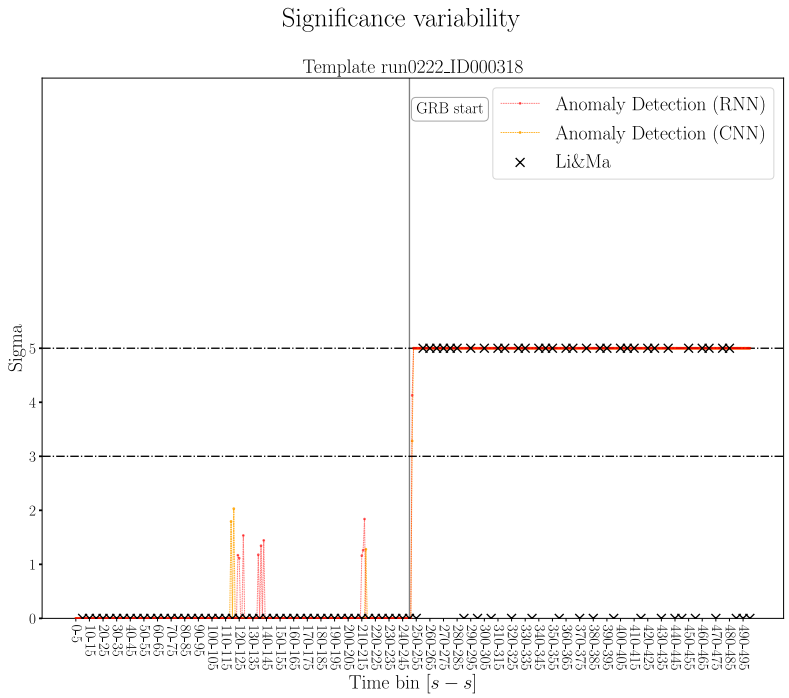


Figure 6.56: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

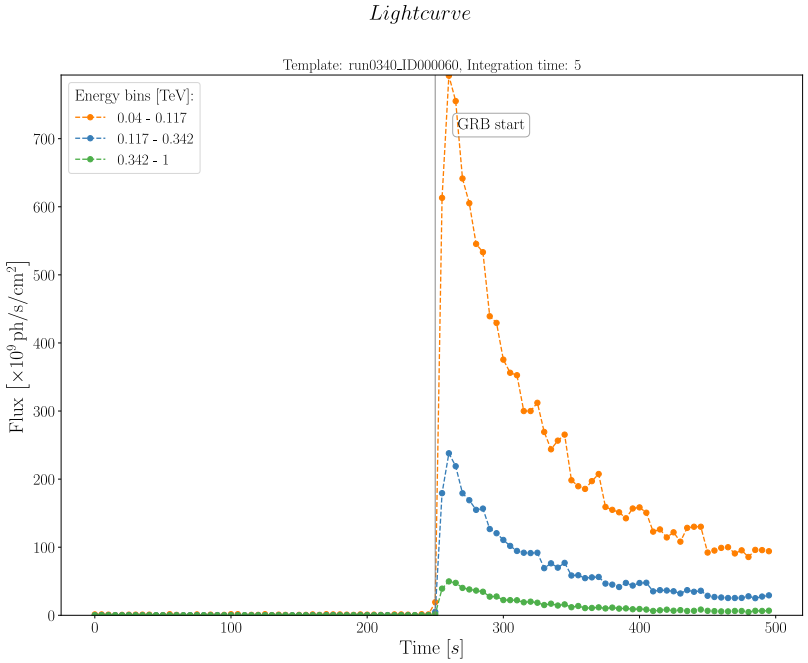


Figure 6.57: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

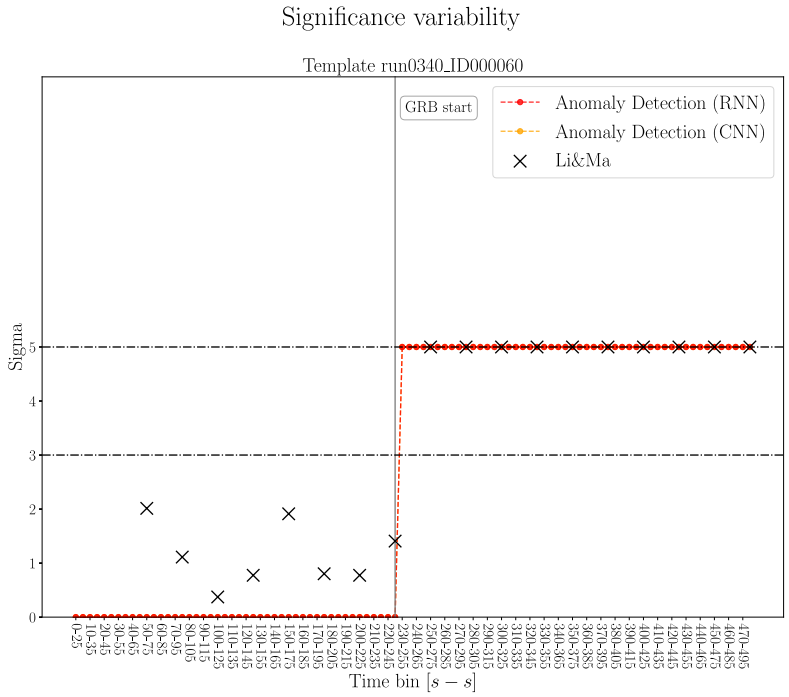


Figure 6.58: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

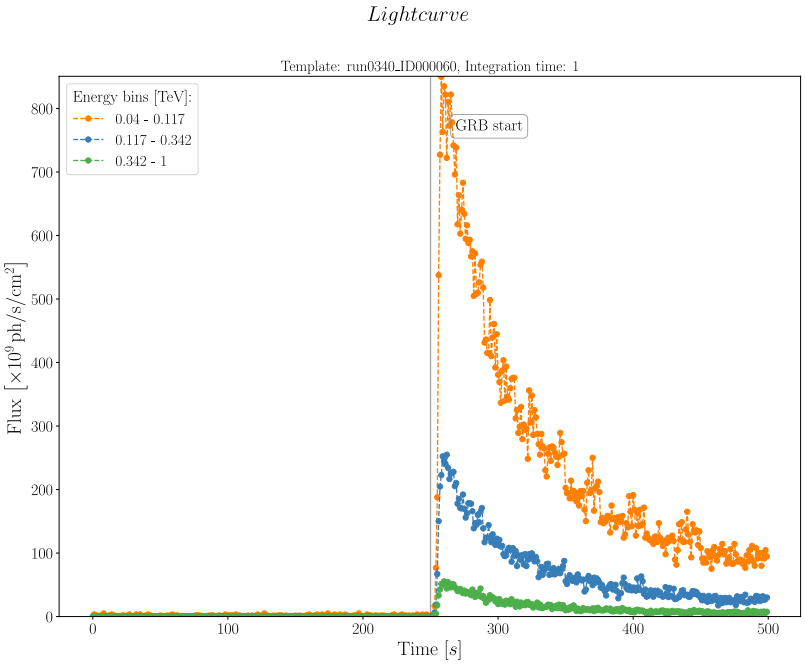


Figure 6.59: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

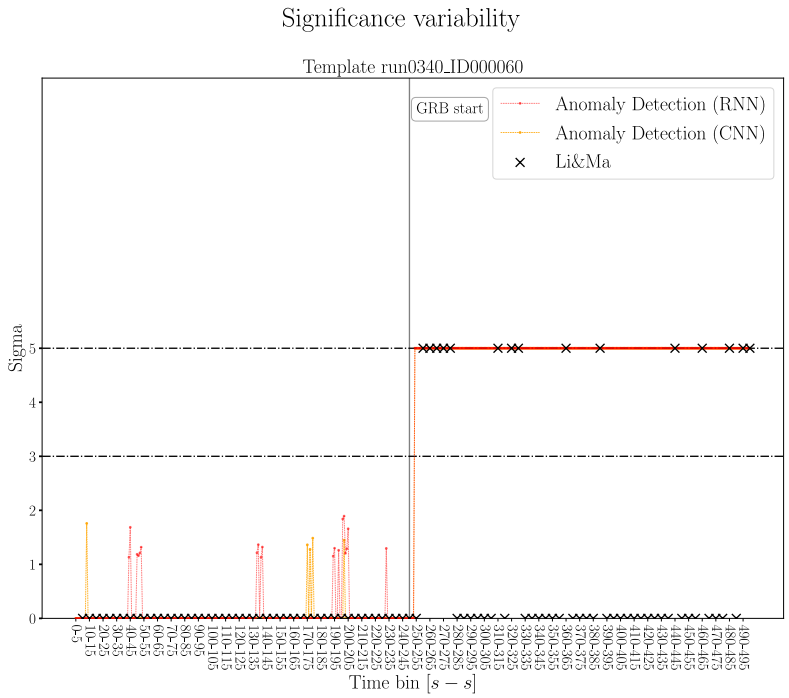


Figure 6.60: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

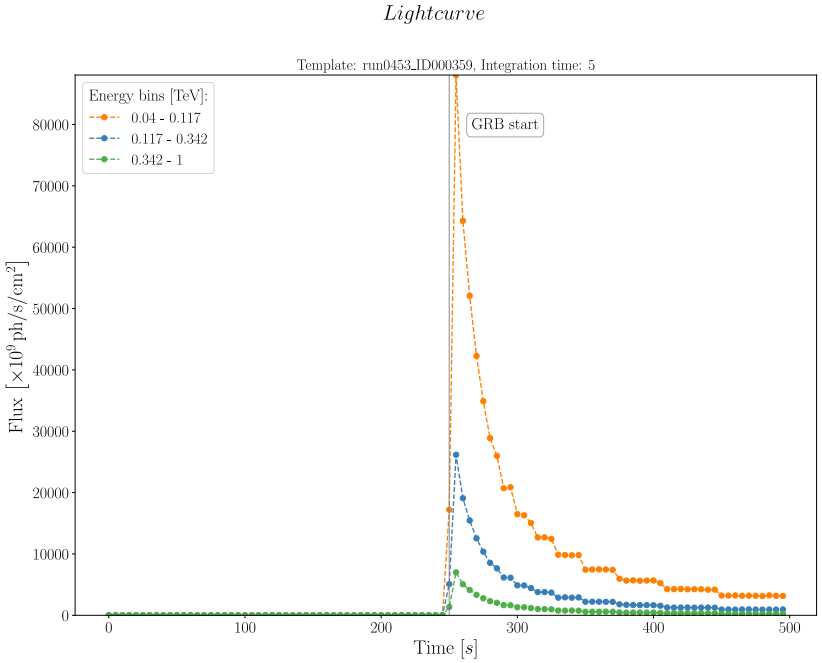


Figure 6.61: A GRB simulated trials in the short-exposure scenario (integration time = 5 seconds).

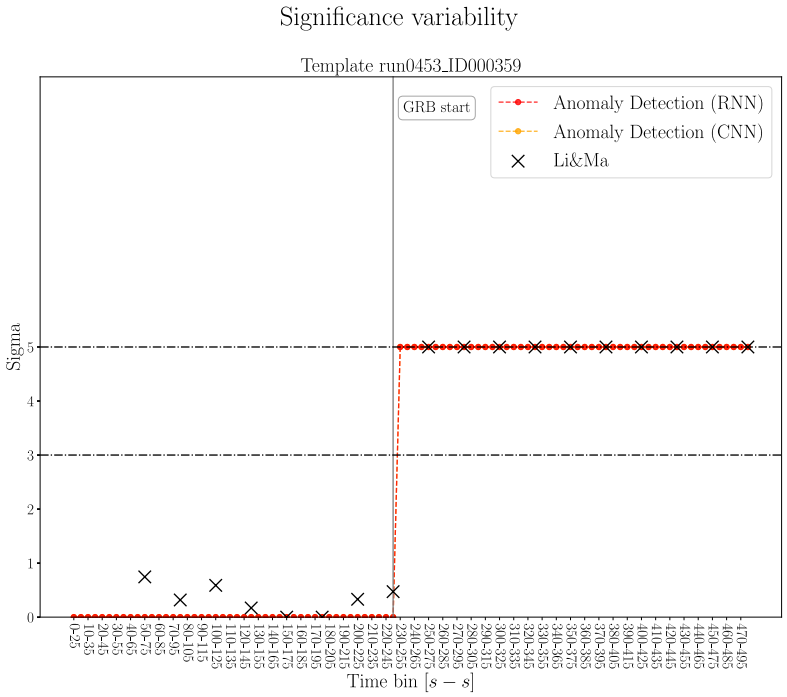


Figure 6.62: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the short-exposure scenario (integration time = 5 seconds).

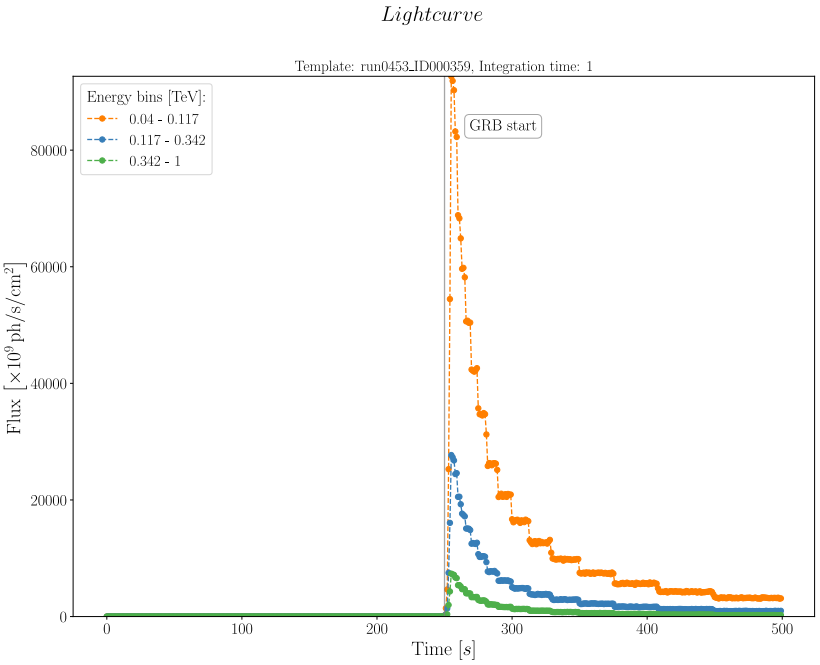


Figure 6.63: A GRB simulated trials in the very short-exposure scenario (integration time = 1 second).

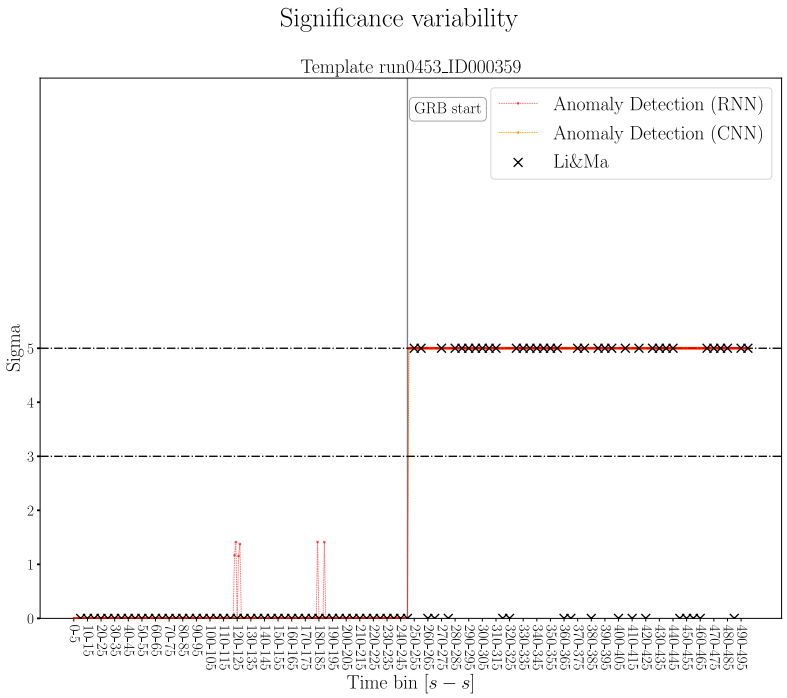


Figure 6.64: The *significance variability* plot shows the analysis result (sigma) of the RNN and CNN-based autoencoders and Li&Ma on the simulated trial in the very short-exposure scenario (integration time = 1 second).

# 7

# APPENDIX B

## 7.1 TRAINING AND INFERENCE (RECONSTRUCTIONS AND CLASSIFICATIONS)

### 7.1.1 *Losses during training*

Figures 7.1, 7.2, 7.3, and 7.4, show the evolution of the loss during training, for the RNN and CNN autoencoders, in both short-term and very short-term scenarios. The models were trained for five epochs.

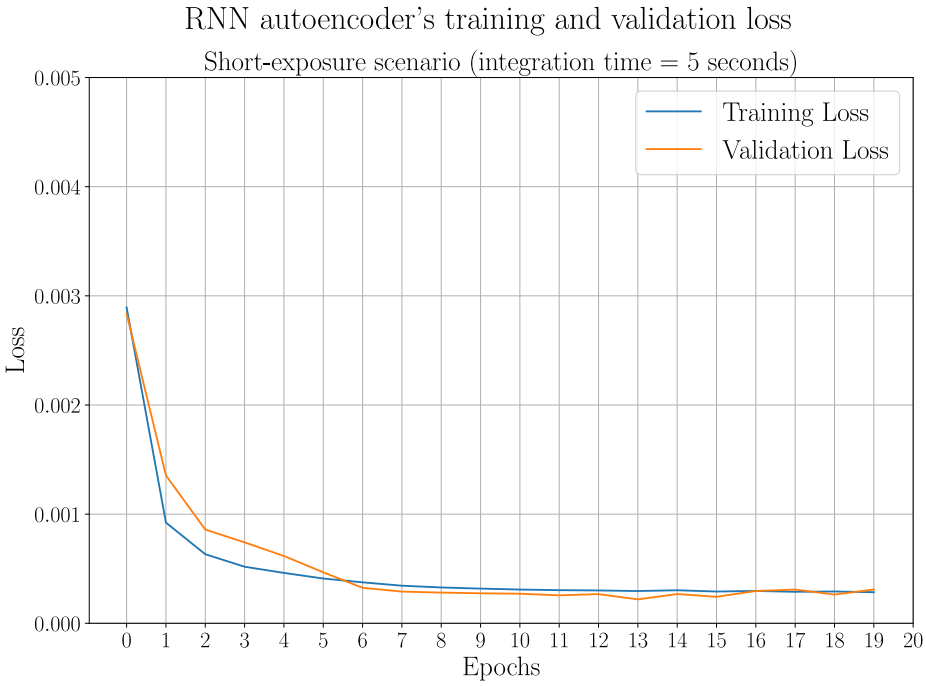


Figure 7.1: The training and validation losses during the training of the autoencoder with recurrent layers, in the short-exposure scenario (integration time = 5 seconds).

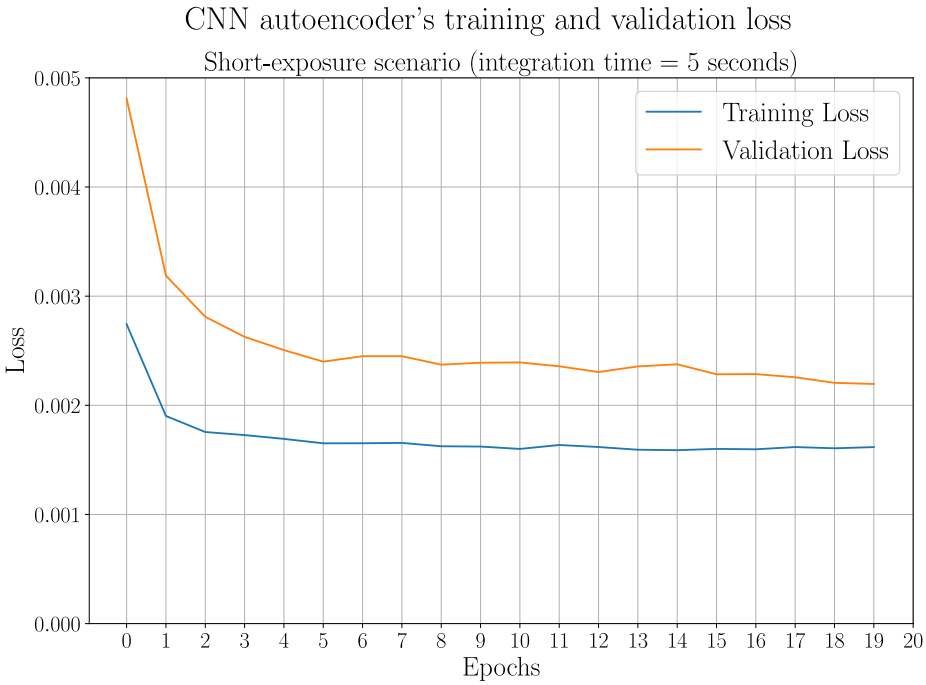


Figure 7.2: The training and validation losses during the training of the autoencoder with convolutional layers, in the short-exposure scenario (integration time = 5 seconds).

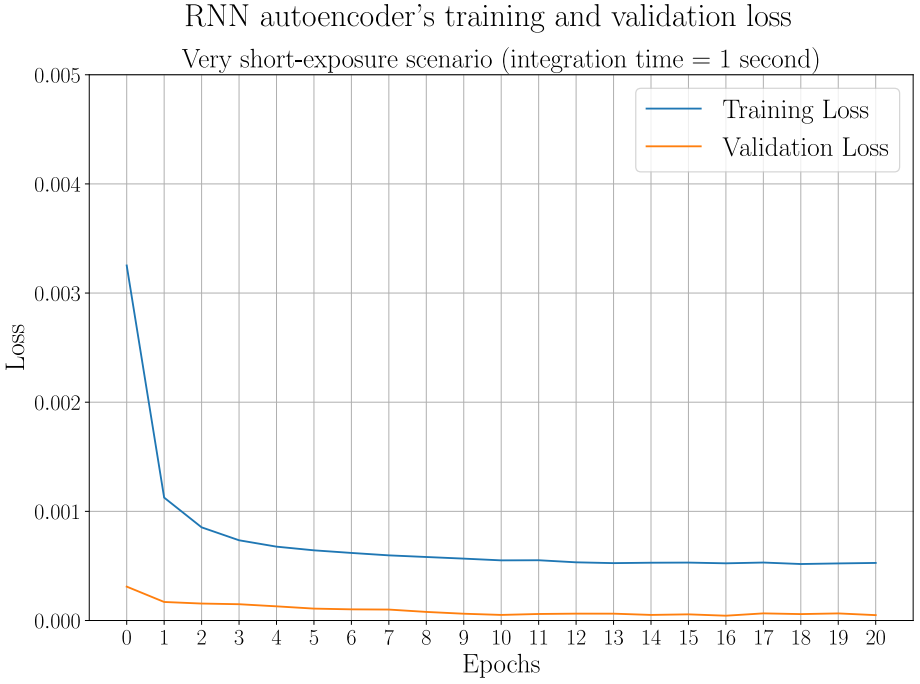


Figure 7.3: The training and validation losses during the training of the autoencoder with recurrent layers, in the very short-exposure scenario (integration time = 1 second).

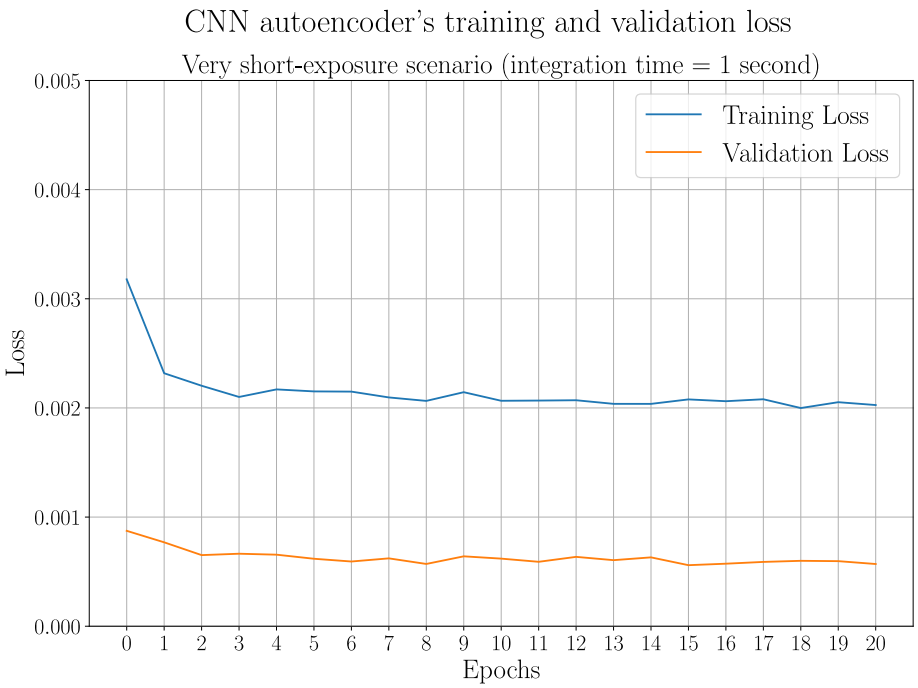


Figure 7.4: The training and validation losses during the training of the autoencoder with convolutional layers, in the very short-exposure scenario (integration time = 1 second).

### 7.1.2 *Performance metrics for classification*

Figures 7.5, 7.6, 7.7, and, 7.8 show how the accuracy, precision, and recall metrics vary as a function of the classification threshold, for the RNN and CNN autoencoders, in both short-term and very short-term scenarios. The classification threshold values correspond to discrete $\sigma$ levels up to 5.

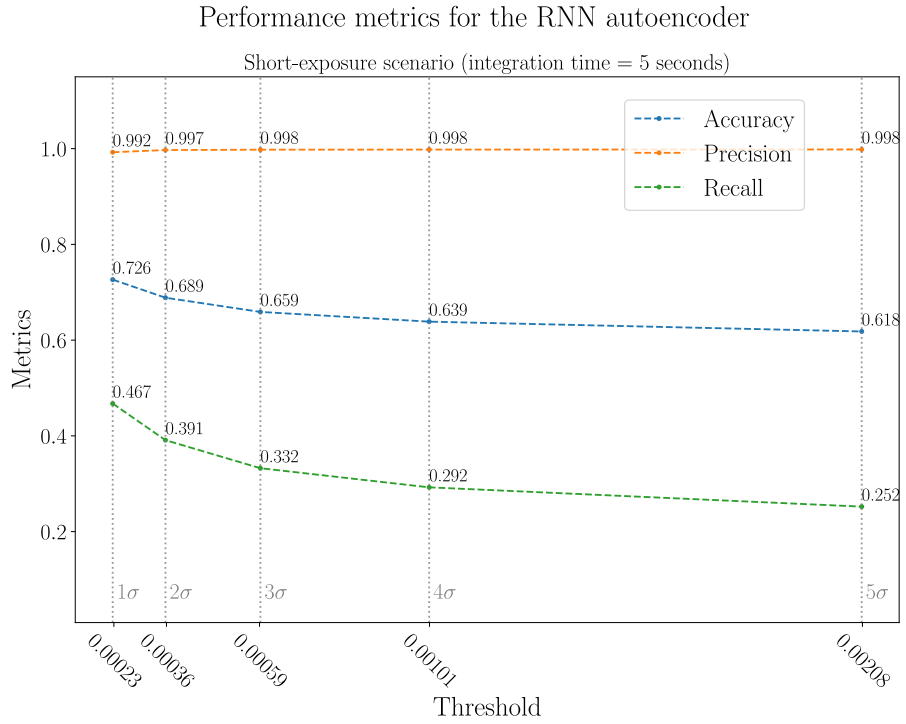


Figure 7.5: Accuracy, precision, and recall, as a function of the classification threshold for the RNN autoencoder in the short-term exposure scenario (integration time = 5 seconds).

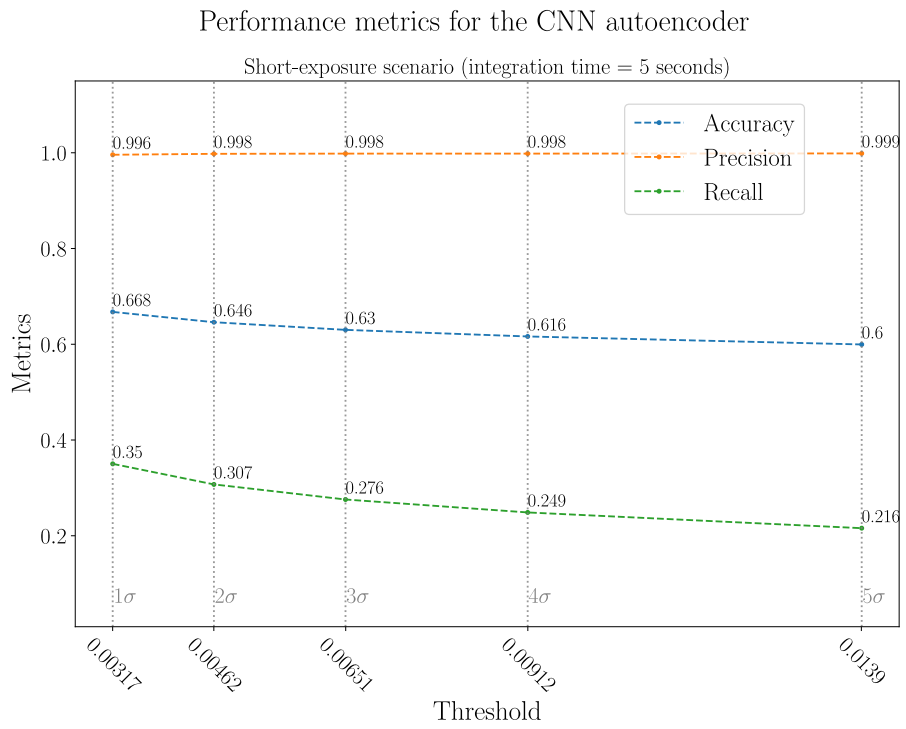


Figure 7.6: Accuracy, precision, and recall, as a function of the classification threshold for the CNN autoencoder in the short-term exposure scenario (integration time = 5 seconds).

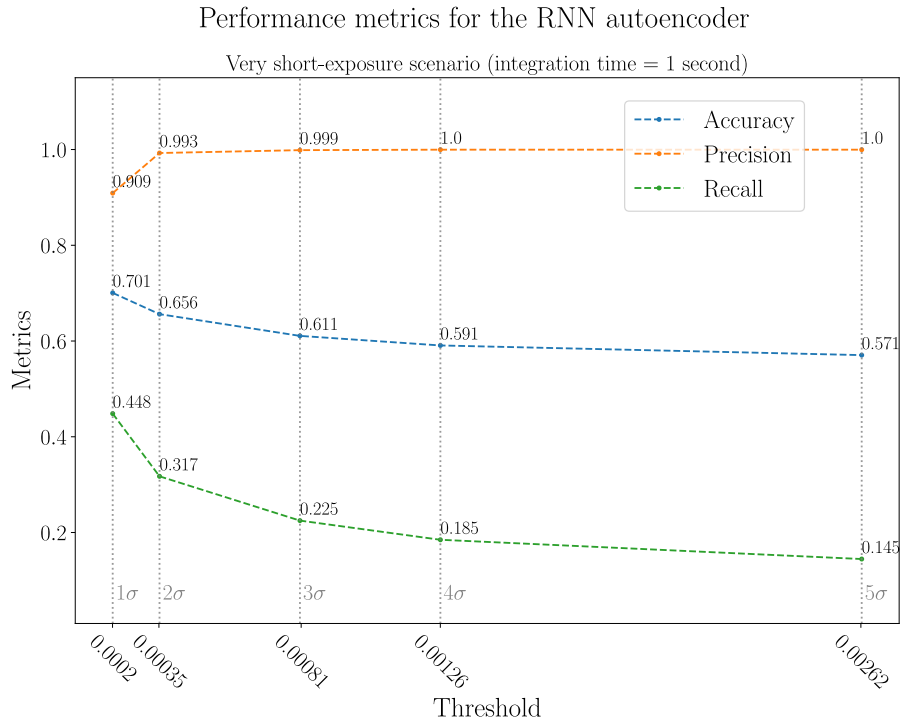


Figure 7.7: Accuracy, precision, and recall, as a function of the classification threshold for the RNN autoencoder in the very short-term exposure scenario (integration time = 1 second).

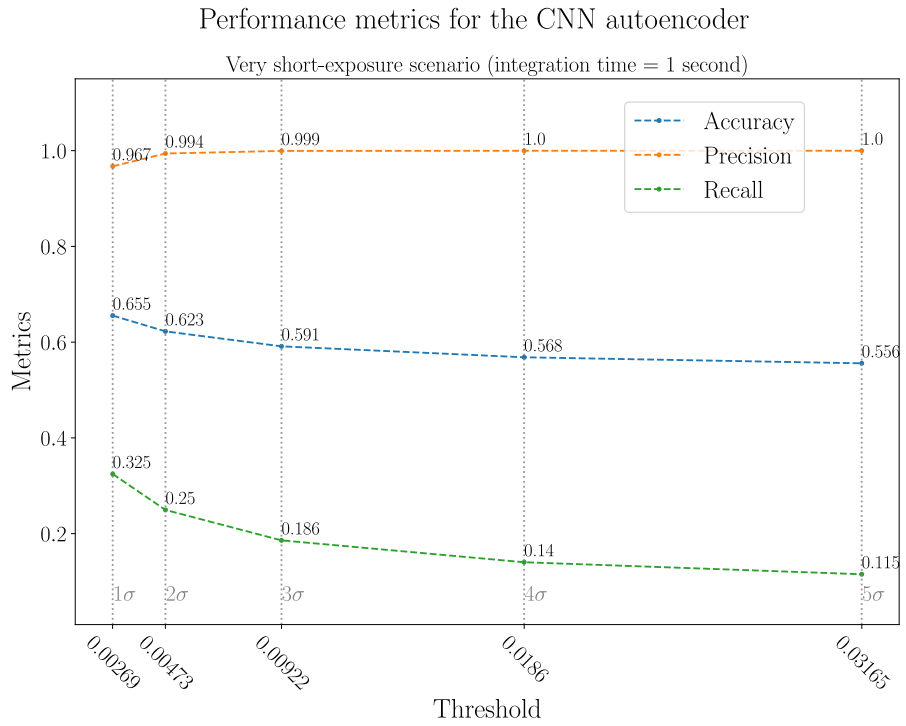


Figure 7.8: Accuracy, precision, and recall, as a function of the classification threshold for the CNN autoencoder in the very short-term exposure scenario (integration time = 1 second).

### 7.1.3 *Reconstructions and inferences*

The following plots show several reconstructions and classifications the autoencoder model with recurrent layers performs in the short-exposure scenario (integration time = 5 seconds). Four trials have been selected from the previous chapter: 6.1, 6.25, 6.37, 6.33.

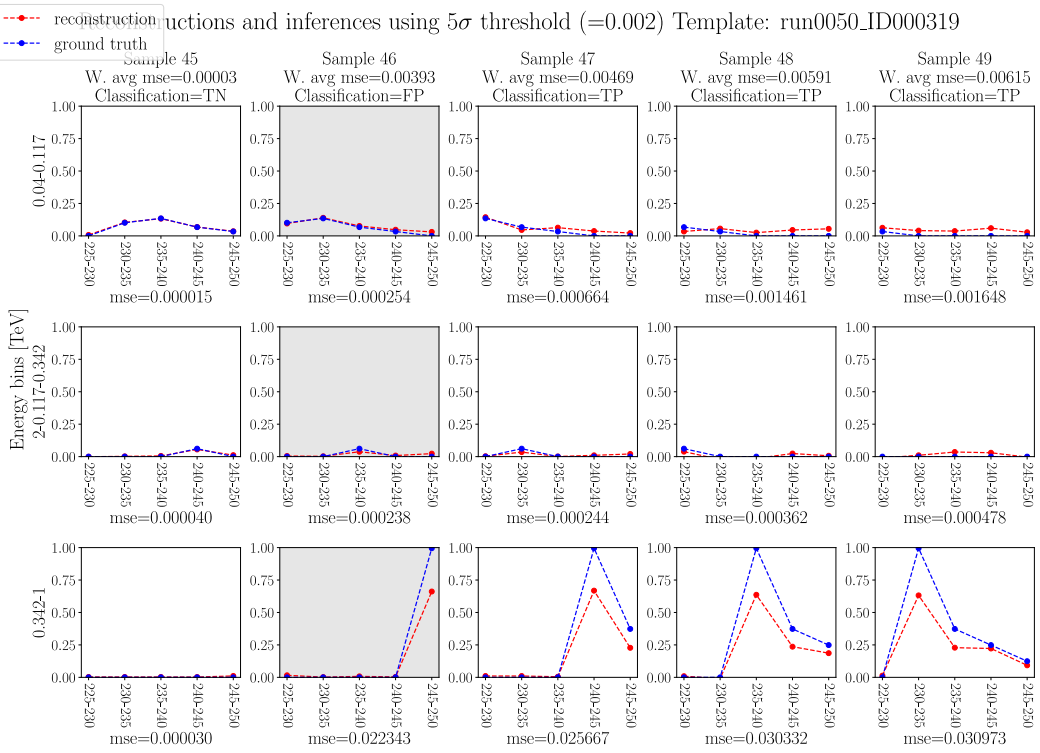


Figure 7.9: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

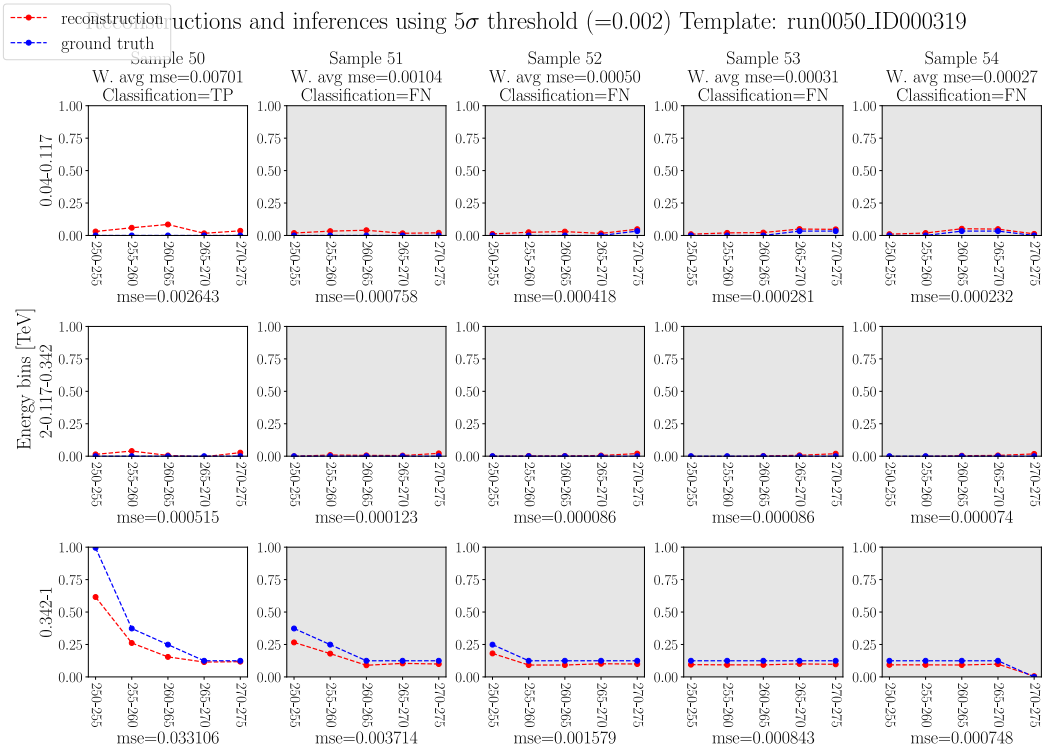


Figure 7.10: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

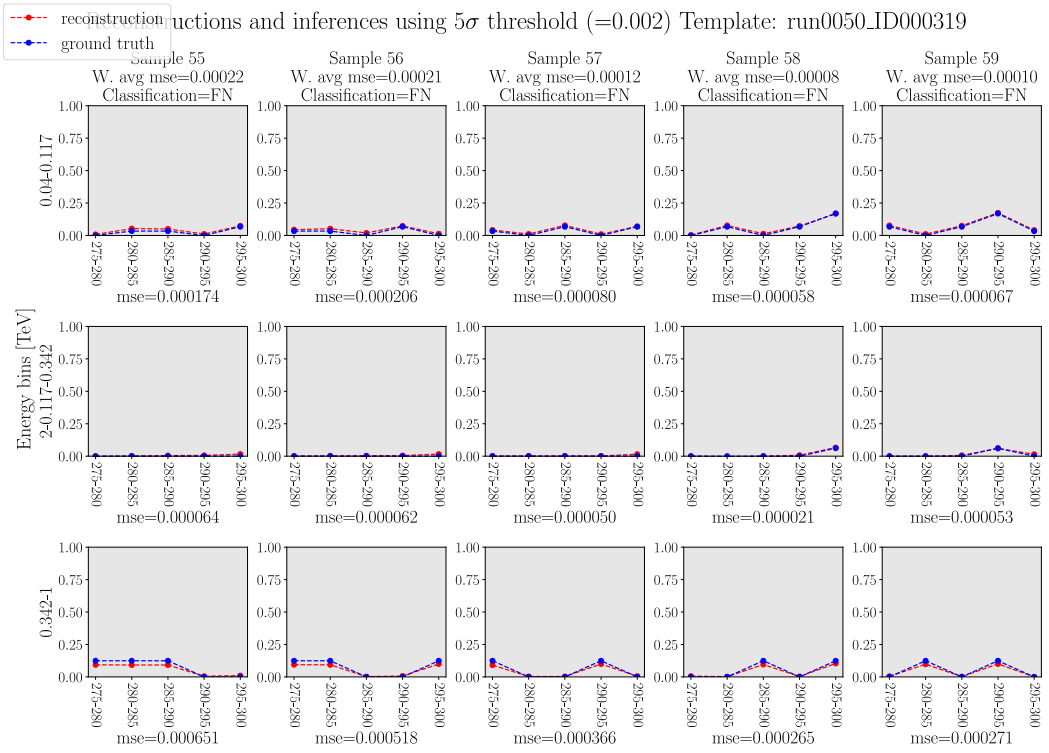


Figure 7.11: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

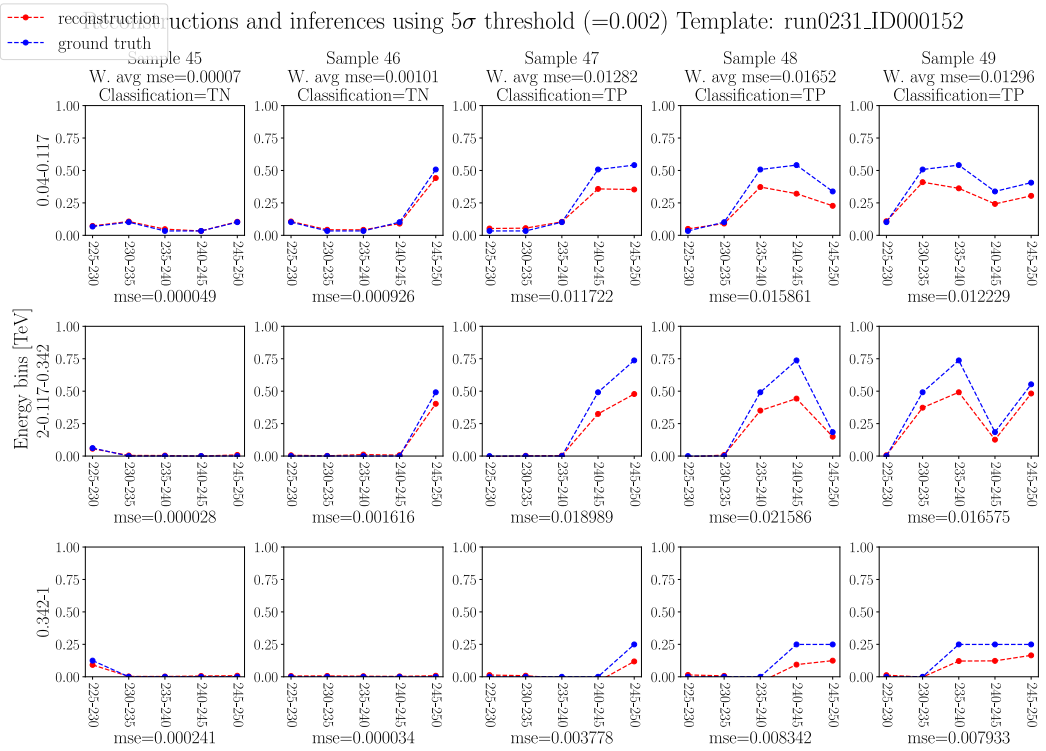


Figure 7.12: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

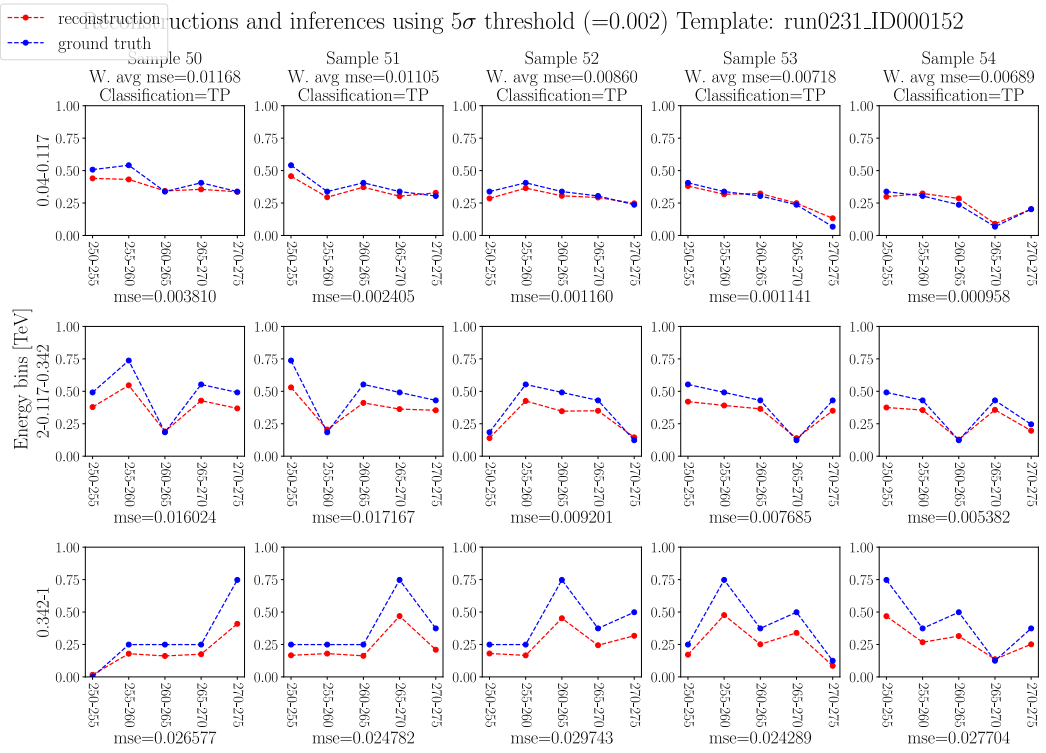


Figure 7.13: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

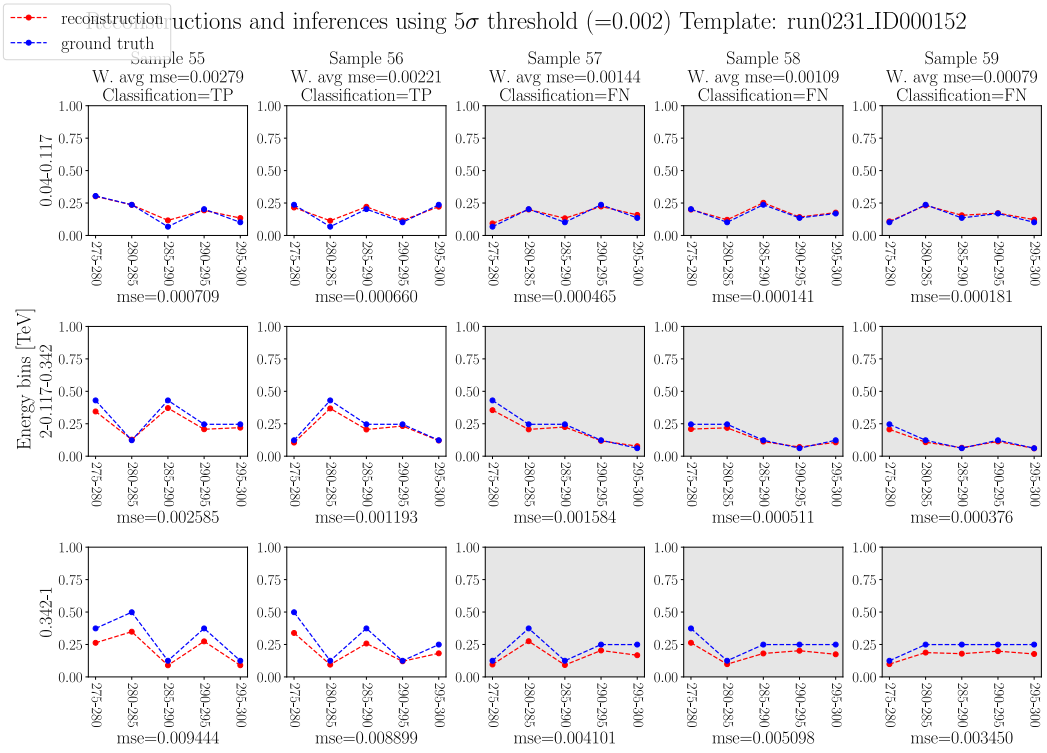


Figure 7.14: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

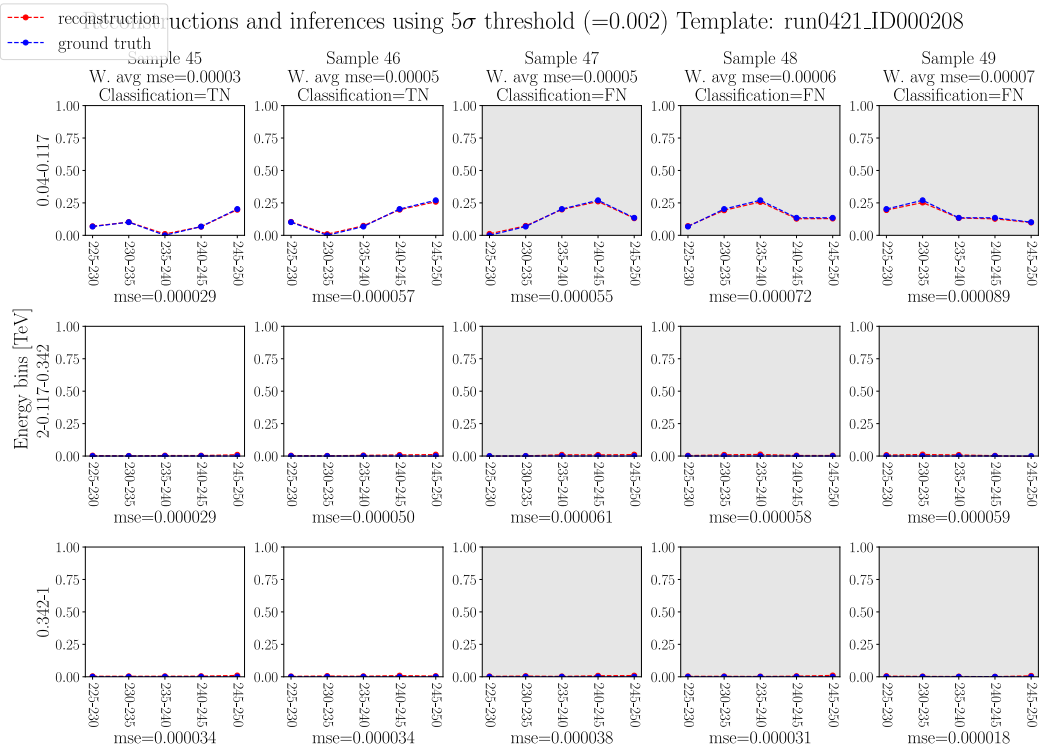


Figure 7.15: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

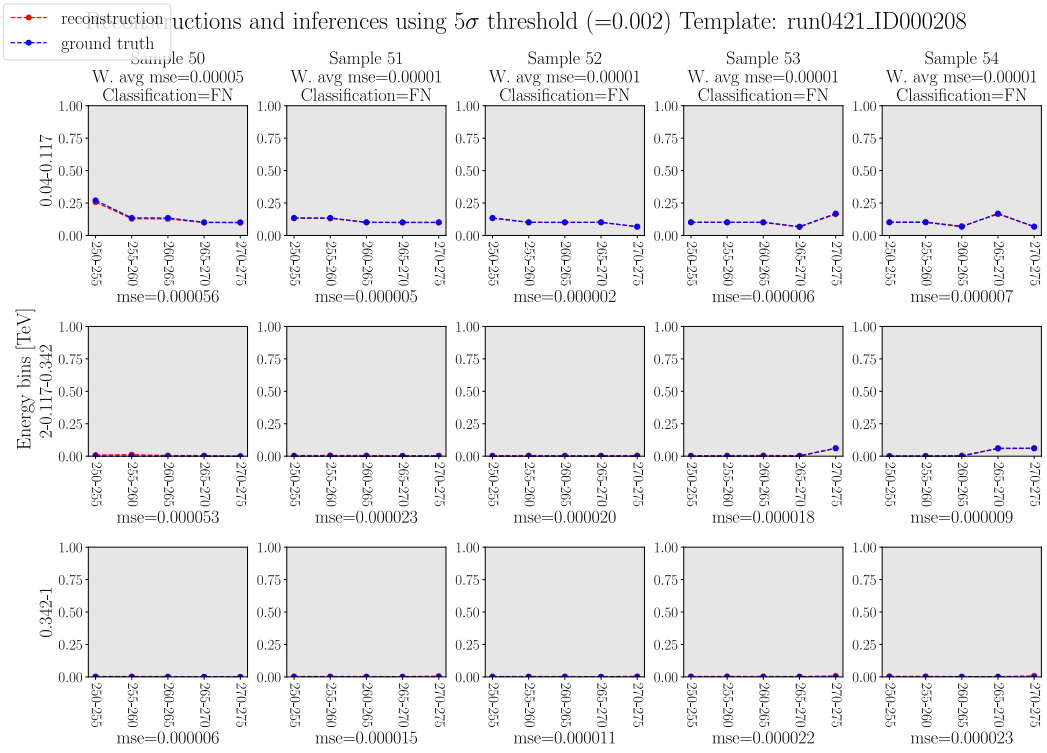


Figure 7.16: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

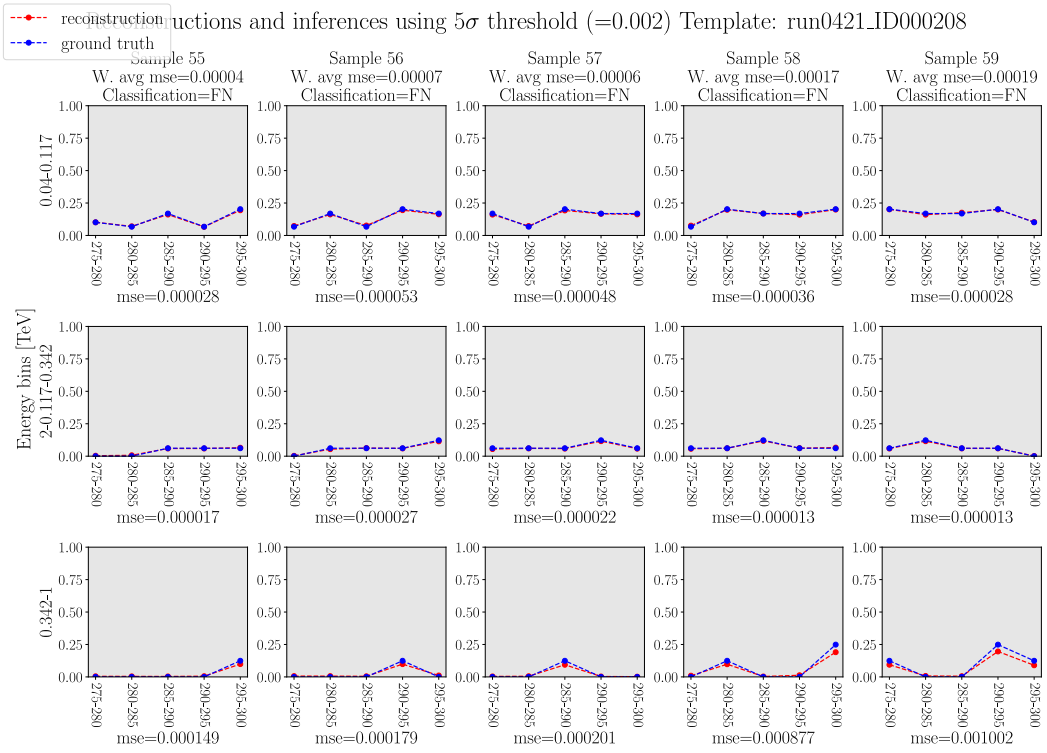


Figure 7.17: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

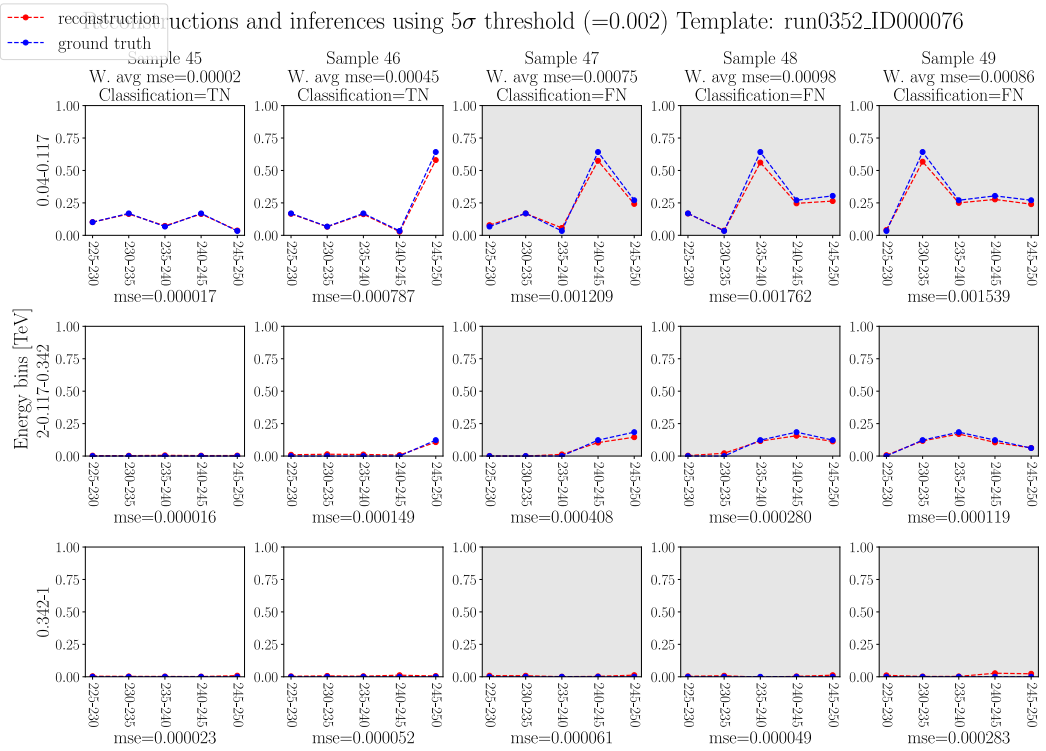


Figure 7.18: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

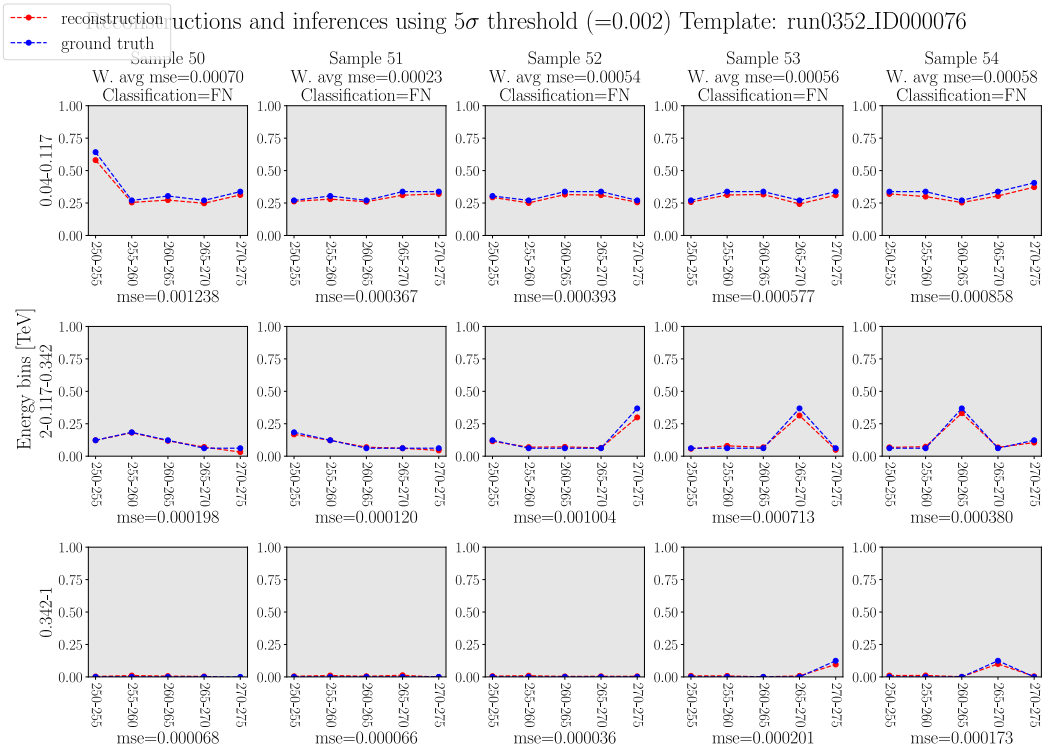


Figure 7.19: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

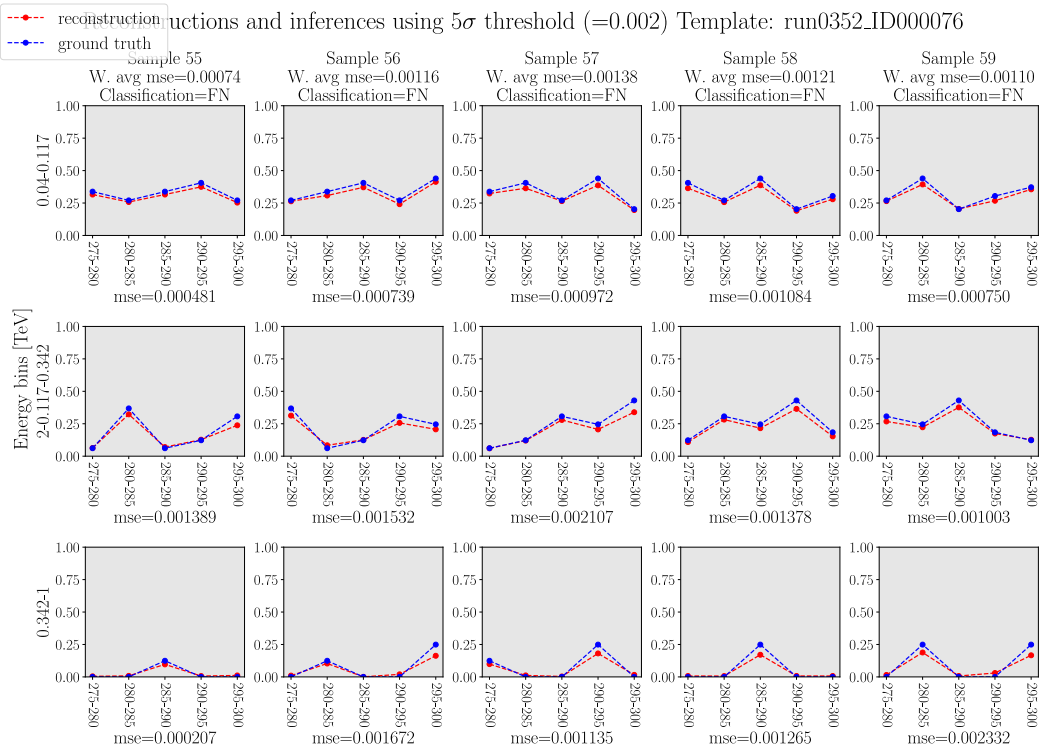


Figure 7.20: The reconstructions performed by the autoencoder with recurrent layers for five sub-sequences extracted from the time series using a sliding window with stride=1, in the short-exposure scenario (integration time = 5 seconds).

# 8

# APPENDIX C

## 8.1 p-value analysis

Figures 8.1, 8.2, 8.3, and 8.4, show the results of the p-value analyses, one for each autoencoder model (convolutional and recurrent) in the short-term and very short-term settings.

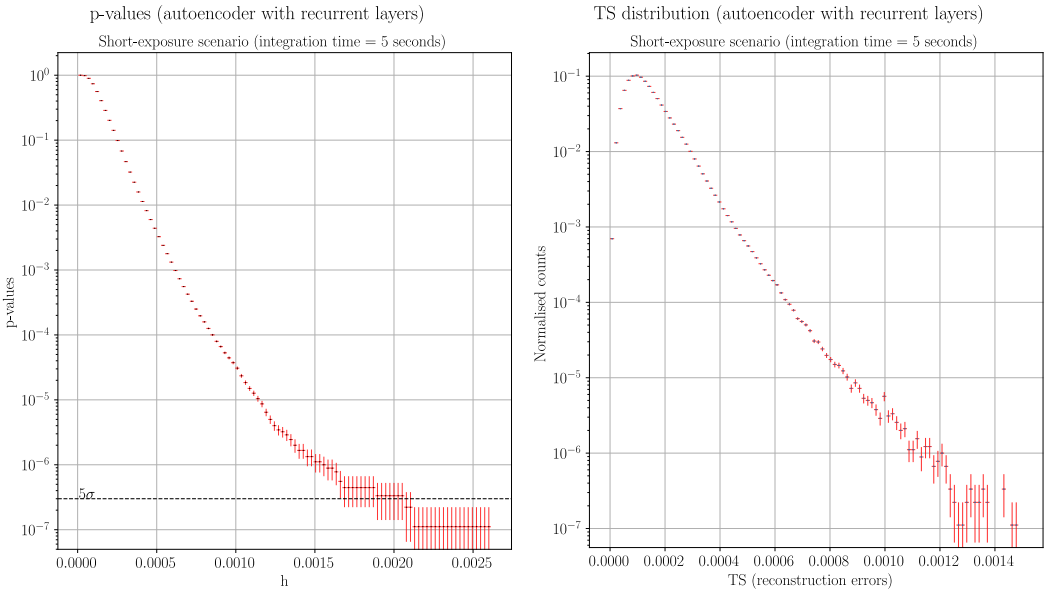


Figure 8.1: TS distribution and p-values for the autoencoder model with recurrent layers in the short-term scenario (integration time = 5 seconds).

| p-values | | | |
|---|---|---|---|
| Threshold | p-value | ± error | Sigma |
| 11 0.000227 | 1.47e-01 | 1.25e-04 | 1.073 |
| 0.000279 | 6.82e-02 | 8.69e-05 | 1.491 |
| 0.000357 | 2.25e-02 | 4.99e-05 | 2.005 |
| 0.000462 | 5.98e-03 | 2.58e-05 | 2.514 |
| 0.000592 | 1.32e-03 | 1.21e-05 | 3.006 |
| 0.000749 | 2.50e-04 | 5.27e-06 | 3.480 |
| 0.001009 | 3.08e-05 | 1.85e-06 | 4.007 |
| 0.001296 | 3.22e-06 | 5.98e-07 | 4.511 |
| 0.002078 | 2.22e-07 | 1.57e-07 | 5.049 |

Table 8.1: An example of p-value analysis for the autoencoder model with recurrent layers in the short-term scenario (integration time = 5 seconds). The table shows a subset of all the rows. Only the threshold values corresponding to predefined sigma levels are shown.

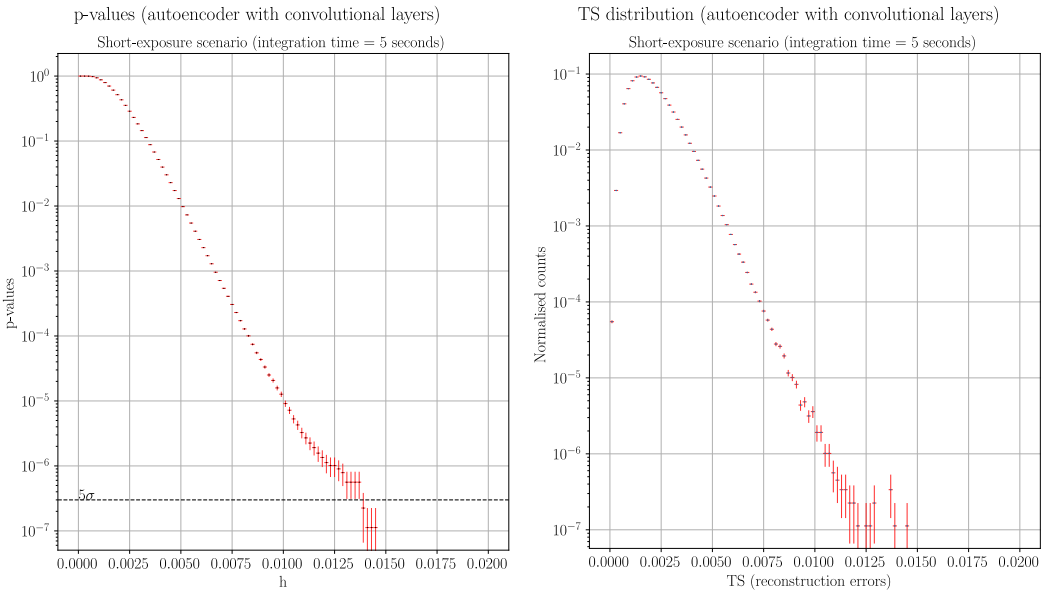


Figure 8.2: TS distribution and p-values for the autoencoder model with convolutional layers in the short-term scenario (integration time = 5 seconds).

| p-values | | | |
|---|---|---|---|
| Threshold | p-value | ± error | Sigma |
| 0.003171 | 1.28e-01 | 1.2e-04 | 1.134 |
| 0.003751 | 6.11e-02 | 8.29e-05 | 1.545 |
| 0.004476 | 2.28e-02 | 5.06e-05 | 2.000 |
| 0.005492 | 5.32e-03 | 2.45e-05 | 2.554 |
| 0.006507 | 1.23e-03 | 1.17e-05 | 3.029 |
| 0.007667 | 2.31e-04 | 5.10e-06 | 3.502 |
| 0.009118 | 3.15e-05 | 1.88e-06 | 4.001 |
| 0.010859 | 3.26e-06 | 6.06e-07 | 4.509 |
| 0.013905 | 2.25e-07 | 1.59e-07 | 5.047 |

Table 8.2: An example of p-value analysis for the autoencoder model with convolutional layers in the short-term scenario (integration time = 5 seconds). The table shows a subset of all the rows. Only the threshold values corresponding to predefined sigma levels are shown.

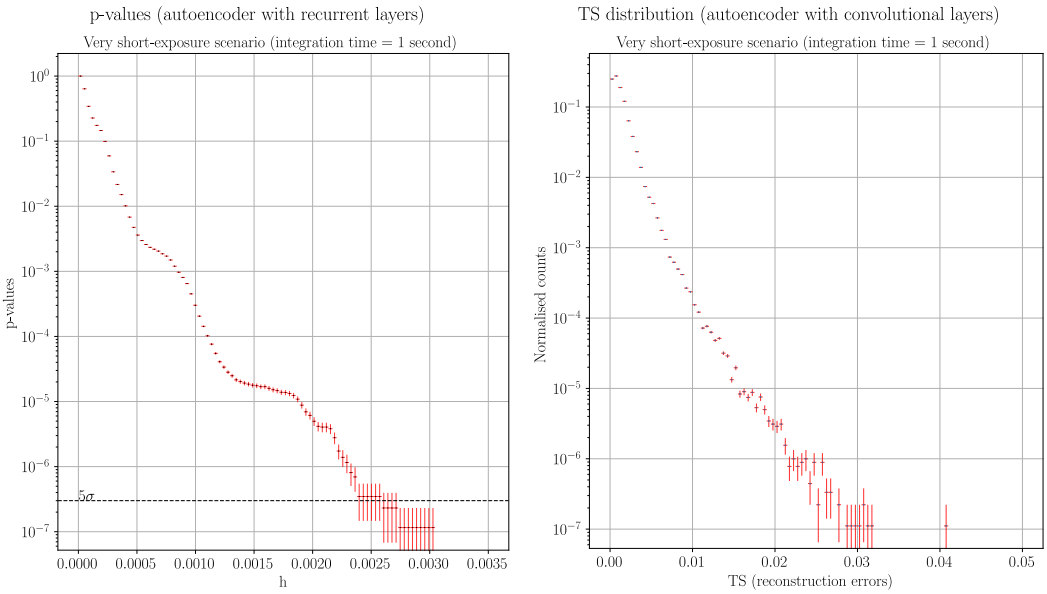


Figure 8.3: TS distribution and p-values for the autoencoder model with recurrent layers in the very short-term scenario (integration time = 1 second).

| p-values | | | |
|---|---|---|---|
| Threshold | p-value | ± error | Sigma |
| 0.000201 | 1.34e-01 | 1.24e-04 | 1.108 |
| 0.000261 | 5.84e-02 | 8.22e-05 | 1.568 |
| 0.000322 | 2.37e-02 | 5.24e-05 | 1.983 |
| 0.000443 | 6.27e-03 | 2.69e-05 | 2.497 |
| 0.000806 | 1.32e-03 | 1.24e-05 | 3.006 |
| 0.001018 | 2.31e-04 | 5.17e-06 | 3.502 |
| 0.001260 | 3.02e-05 | 1.87e-06 | 4.011 |
| 0.002168 | 3.37e-06 | 6.23e-07 | 4.503 |
| 0.002623 | 2.31e-07 | 1.64e-07 | 5.041 |

Table 8.3: An example of p-value analysis for the autoencoder model with recurrent layers in the very short-term scenario (integration time = 1 second). The table shows a subset of all the rows. Only the threshold values corresponding to predefined sigma levels are shown.

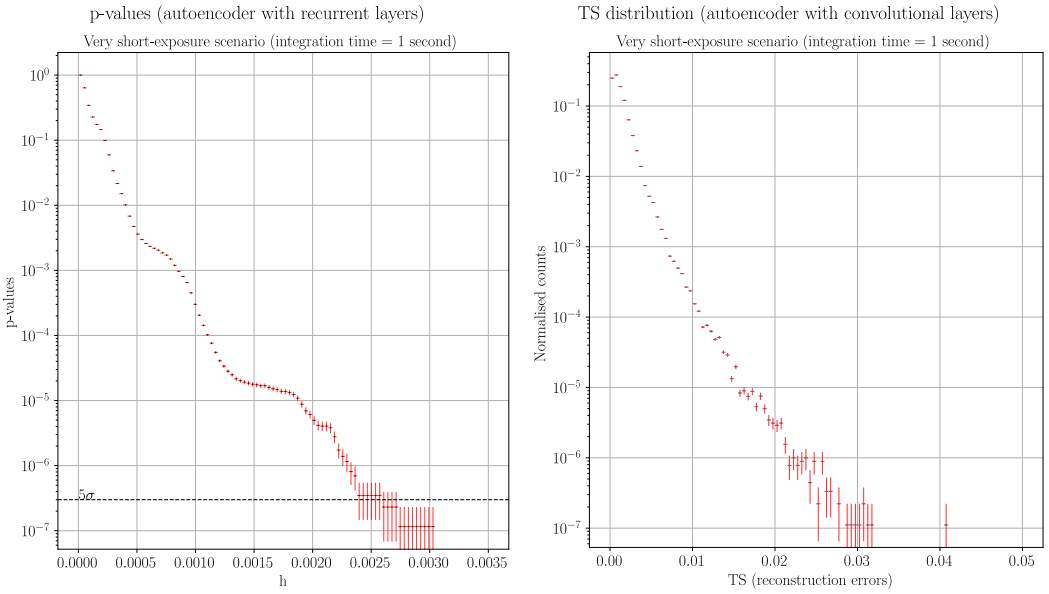


Figure 8.4: TS distribution and p-values for the autoencoder model with convolutional layers in the very short-term scenario (integration time = 1 second).

| p-values | | | |
|---|---|---|---|
| Threshold | p-value | ± error | Sigma |
| 0.002693 | 1.02e-01 | 1.07e-04 | 1.268 |
| 0.003101 | 6.95e-02 | 8.79e-05 | 1.479 |
| 0.004732 | 1.85e-02 | 4.53e-05 | 2.087 |
| 0.006363 | 5.94e-03 | 2.57e-05 | 2.516 |
| 0.009218 | 1.25e-03 | 1.18e-05 | 3.023 |
| 0.013295 | 2.11e-04 | 4.89e-06 | 3.525 |
| 0.018597 | 2.96e-05 | 1.81e-06 | 4.015 |
| 0.025529 | 2.78e-06 | 5.55e-07 | 4.543 |
| 0.031646 | 2.22e-07 | 1.57e-07 | 5.049 |

Table 8.4: An example of p-value analysis for the autoencoder model with convolutional layers in the very short-term scenario (integration time = 1 second). The table shows a subset of all the rows. Only the threshold values corresponding to predefined sigma levels are shown.

# BIBLIOGRAPHY


[Abb+16a] B. P. Abbott, R. Abbott, T. D. Abbott, and et al. "Observation of Gravitational Waves from a Binary Black Hole Merger." In: *Physical Review Letters* 116.6 (2016), p. 061102.

[Abb+16b] B. P. Abbott et al. "Observation of Gravitational Waves from a Binary Black Hole Merger." In: *Phys. Rev. Lett.* 116 (6 2016), p. 061102. DOI: 10.1103/PhysRevLett.116.061102. URL: https://link.aps.org/doi/10.1103/PhysRevLett.116.061102.

[Abb+17] B. P. Abbott et al. "Gravitational Waves and Gamma-Rays from a Binary Neutron Star Merger: GW170817 and GRB 170817A." In: *The Astrophysical Journal Letters* 848.2 (2017), p. L13. DOI: 10.3847/2041-8213/aa920c. URL: https://dx.doi.org/10.3847/2041-8213/aa920c.

[Abd+10] A. A. Abdo, M. Ackermann, M. Ajello, and et al. "The First Fermi Large Area Telescope Catalog of Gamma-Ray Pulsars." In: *The Astrophysical Journal Supplement Series* 187.1 (2010), pp. 460–494.

[Ach+19] A. Acharyya et al. "Monte Carlo studies for the optimisation of the Cherenkov Telescope Array layout." In: *Astroparticle Physics* 111 (2019), pp. 35–53. ISSN: 0927-6505. DOI: https://doi.org/10.1016/j.astropartphys.2019.04.001. URL: https://www.sciencedirect.com/science/article/pii/S0927650519300234.

[Ack+15] M. Ackermann, M. Ajello, A. Allafort, and et al. "The Third Catalog of Active Galactic Nuclei Detected by the Fermi Large Area Telescope." In: *The Astrophysical Journal* 810.2 (2015), p. 14.

[Agg16] C. Aggarwal. *Outlier Analysis*. 2nd. Springer, 2016.

[AMAZ17] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. "Understanding of a convolutional neural network." In: *2017 International Conference on Engineering and Technology (ICET)*. 2017, pp. 1–6. DOI: 10.1109/ICEngTechnol.2017.8308186.

[Aa08] J. Albert and et al. "Implementation of the Random Forest method for the Imaging Atmospheric Cherenkov Telescope MAGIC." In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 588.3 (2008), pp. 424–432. ISSN: 0168-9002.

[Aub18] Pierre Aubert. "Calcul haute performance pour la détection de rayon Gamma." Theses. Université Paris-Saclay, Oct. 2018. URL: https://theses.hal.science/tel-02119197.

[BP17] Dalya Baron and Dovi Poznanski. "The weirdest SDSS galaxies: results from an outlier detection algorithm." In: 465.4 (Mar. 2017), pp. 4530–4555. DOI: 10.1093/mnras/stw302110.48550/arXiv.1611.07526. arXiv: 1611.07526 [astro-ph.GA].

[Bar+22] L. Baroncelli et al. "Rta-dq-lib: Software Library to Perform Online Data Quality Analysis of Scientific Data." In: *Astronomical Society of the Pacific Conference Series*. Ed. by Jose Enrique Ruiz, Francesco Pierfedereci, and Peter Teuben. Vol. 532. Astronomical Society of the Pacific Conference Series. July 2022, p. 365. DOI: 10.48550/arXiv.2105.08648. arXiv: 2105.08648 [astro-ph.IM].

[Bar20] J.A. Barrio. "Status of the large size telescopes and medium size telescopes for the Cherenkov Telescope Array observatory." In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 952 (2020). 10th International Workshop on Ring Imaging Cherenkov Detectors (RICH 2018), p. 161588. ISSN: 0168-9002. DOI: https://doi.org/10.1016/j.nima.2018.11.047. URL: https://www.sciencedirect.com/science/article/pii/S016890021831622X.

[BFH07] D. Berge, S. Funk, and J. Hinton. "Background modelling in very-high-energy gamma-ray astronomy." In: 466.3 (May 2007), pp. 1219–1229. DOI: 10.1051/0004-6361:20066674. arXiv: astro-ph/0610959 [astro-ph].

[Ber+19] Maria Grazia Bernardini et al. *POSyTIVE – a GRB population study for the Cherenkov Telescope Array (ICRC-2019)*. 2019. DOI: 10.48550/ARXIV.1908.01544. URL: https://arxiv.org/abs/1908.01544.

[BG11] P. N. Bhat and S. Guiriec. "An overview of the current understanding of Gamma-ray Bursts in the Fermi era." In: (2011). DOI: 10.48550/ARXIV.1111.4909. URL: https://arxiv.org/abs/1111.4909.

[BG+20] A. Blázquez-García et al. "A Review on Outlier/Anomaly Detection in Time Series Data." In: *IEEE Access* (2020).

[BNP+12] Omer Bromberg, Ehud Nakar, Tsvi Piran, et al. "An observational imprint of the collapsar model of long gamma-ray bursts." In: *The Astrophysical Journal* 749.2 (2012), p. 110.

[Ba14] A. Bulgarelli and et al. "A prototype for the real-time analysis of the Cherenkov Telescope Array." In: *Proc. SPIE 9145*. 2014, p. 91452X. DOI: 10.1117/12.2054744.

[Bul+15] A. Bulgarelli et al. "The On-Site Analysis of the Cherenkov Telescope Array." In: *34th International Cosmic Ray Conference (ICRC2015)* 34 (2015), p. 763.

[Bul+19] A. Bulgarelli et al. "Second AGILE catalogue of gamma-ray sources." In: *aap* 627, A13 (July 2019), A13. DOI: 10.1051/0004-6361/201834143. arXiv: 1903.06957 [astro-ph.HE].

[Bul+21] A. Bulgarelli et al. "The Science Alert Generation system of the Cherenkov Telescope Array Observatory." In: *Proceedings of 37th International Cosmic Ray Conference — PoS(ICRC2021)*. Sissa Medialab, 2021. DOI: 10.22323/1.395.0937. URL: https://doi.org/10.22323\%2F1.395.0937.

[Bul+22] A. Bulgarelli et al. *Detailed Design of the Science Alert Generation of the ACADA System*. 2022. URL: https://redmine.cta-observatory.org.

[CG93] Gail A. Carpenter and Stephen Grossberg. "Normal and amnesic learning, recognition and memory by a neural model of cortico-hippocampal interactions." In: *Trends in Neurosciences* 16.4 (1993), pp. 131–137. ISSN: 0166-2236. DOI: https://doi.org/10.1016/0166-2236(93)90118-6. URL: https://www.sciencedirect.com/science/article/pii/0166223693901186.

[Cas79] W Cash. "Parameter estimation in astronomy through application of the likelihood ratio." In: *The Astrophysical Journal* 228 (1979). "cit. on p. 24", pp. 939–947. DOI: 10.1086/156922.

[Cen21] The HEASARC (High Energy Astrophysics Science Archive Research Center). *Flexible Image Transport System*. 2021. URL: https://fits.gsfc.nasa.gov.

[CBK09] Varun Chandola, Arindam Banerjee, and Vipin Kumar. "Anomaly Detection: A Survey." In: *ACM Comput. Surv.* 41 (July 2009). DOI: 10.1145/1541880.1541882.

[CBK12] Varun Chandola, Arindam Banerjee, and Vipin Kumar. "Anomaly Detection for Discrete Sequences: A Survey." In: *IEEE Transactions on Knowledge and Data Engineering* 24.5 (2012), pp. 823–839. DOI: 10.1109/TKDE.2010.235.

[Cha+17] Shiyu Chang, Yang Zhang, Wei Han, Mo Yu, Xiaoxiao Guo, Wei Tan, Xiaodong Cui, Michael Witbrock, Mark Hasegawa-Johnson, and Thomas Huang. "Dilated Recurrent Neural Networks." In: (Oct. 2017).

[Chi+01] Gianluca Chiozzi et al. "Common Software for the ALMA project." In: (Dec. 2001). DOI: 10.1117/12.461036.

[Cho+21] K. Choi et al. "Deep Learning for Anomaly Detection in Time-Series Data: Review, Analysis, and Guidelines." In: *IEEE Access* (2021).

[Chu+14] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling." In: *arXiv e-prints*, arXiv:1412.3555 (Dec. 2014), arXiv:1412.3555. arXiv: 1412.3555 [cs.NE].

[CP95] Ehud Cohen and Tsvi Piran. "The Distribution Of Cosmological gamma-ray Bursts." In: *arXiv preprint astro-ph/9502001* (1995).

[Ca14] J. Colome and et al. "Artificial intelligence for the CTA Observatory scheduler." In: *Proc SPIE 9149*. 2014, 91490H. DOI: 10.1117/12.2057090.

[Con23a] The CTA Consortium. *Cherenkov Telescope Array*. 2023. URL: https://www.cta-observatory.org/.

[Con23b] The CTA Consortium. *Cherenkov Telescope Array*. 2023. URL: https://www.cta-observatory.org/project/technology.

[Coo16] Asantha Cooray. "Extragalactic background light measurements and applications." In: *Royal Society Open Science* 3.3 (Mar. 2016), p. 150555. DOI: 10.1098/rsos.150555. arXiv: 1602.03512 [astro-ph.CO].

[Dev+18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." In: *arXiv e-prints*, arXiv:1810.04805 (Oct. 2018), arXiv:1810.04805. arXiv: 1810.04805 [cs.CL].

[DP22] A. Di Piano. *CTA Science Alert Generation*. https://github.com/ambra-dipiano/cta-sag-sci. 2022.

[DP+20] A. Di Piano et al. "Detection of short Gamma-Ray Bursts with CTA through real-time analysis." In: *Astroparticle Physics* (2020).

[DP+21] A. Di Piano et al. "Detection methods for the Cherenkov Telescope Array at very-short exposure times." In: *Astroparticle Physics* (2021).

[Doo+21] Lars Doorenbos, Stefano Cavuoti, Massimo Brescia, Antonio D'Isanto, and Giuseppe Longo. "Comparison of Outlier Detection Methods on Astronomical Image Data." In: *Intelligent Astrophysics*. Ed. by I. Zelinka, M. Brescia, and D. Baron. Vol. 39. 2021, pp. 197–223. DOI: 10.1007/978-3-030-65867-0_910.48550/arXiv.2006.08238.

[Fio+15] V. Fioretti et al. "Real-time analysis sensitivity evaluation of the cherenkov telescope array." In: *32th International Cosmic Ray Conference (ICRC2015)* 32 (2015).

[FM95] G. J. Fishman and C. A. Meegan. "Gamma-Ray Bursts." In: *Annual Review of Astronomy and Astrophysics* 33 (1995), pp. 415–458.

[Gar+21] Astha Garg, Wenyu Zhang, Jules Samaran, Savitha Ramasamy, and Chuan-Sheng Foo. "An Evaluation of Anomaly Detection and Diagnosis in Multivariate Time Series." In: *arXiv e-prints*, arXiv:2109.11428 (Sept. 2021), arXiv:2109.11428. DOI: 10.48550/arXiv.2109.11428. arXiv: 2109.11428 [cs.LG].

[Geh+04] N. Gehrels et al. "The Swift Gamma-Ray Burst Mission." In: 611.2 (Aug. 2004), pp. 1005–1020. DOI: 10.1086/422091. arXiv: astro-ph/0405233 [astro-ph].

[Gem+20] Rosana Noronha Gemaque, Albert França Josuá Costa, Rafael Giusti, and Eulanda Miranda Dos Santos. "An overview of unsupervised drift detection methods." In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 10.6 (2020), e1381.

[GW19] Daniel Giles and Lucianne Walkowicz. "Systematic serendipity: a test of unsupervised machine learning as a method for anomaly detection." In: 484.1 (Mar. 2019), pp. 834–849. DOI: 10.1093/mnras/sty346110.48550/arXiv.1812.07156. arXiv: 1812.07156 [astro-ph.IM].

[Gil+12] Rudy C. Gilmore, Rachel S. Somerville, Joel R. Primack, and Alberto Domínguez. "Semi-analytic modelling of the extragalactic background light and consequences for extragalactic gamma-ray spectra." In: *Monthly Notices of the Royal Astronomical Society* 422.4 (May 2012), pp. 3189–3207. ISSN: 0035-8711. DOI: 10.1111/j.1365-2966.2012.20841.x. eprint: https://academic.oup.com/mnras/article-pdf/422/4/3189/18601416/mnras0422-3189.pdf. URL: https://doi.org/10.1111/j.1365-2966.2012.20841.x.

[Gom12] Andreja Gomboc. "Unveiling the Secrets of Gamma Ray Bursts." In: *Contemporary Physics - CONTEMP PHYS* 53 (June 2012). DOI: 10.1080/00107514.2012.701453.

[Haw80] D. M. Hawkins. "Identification of Outliers." In: *Journal of the Royal Statistical Society. Series C (Applied Statistics)* (1980).

[Haw+16] J. Hawkins, S. Ahmad, S. Purdy, and A. Lavin. "Biological and Machine Intelligence (BAMI)." Initial online release 0.4. 2016. URL: https://numenta.com/resources/biological-and-machine-intelligence/.

[Hil85] A Michael Hillas. "Cherenkov light images of EAS produced by primary gamma." In: OG-9.5-3 (1985).

[Hjo+03] J Hjorth et al. "A very energetic supernova associated with the -ray burst of 29 March 2003." In: *Nature* 423 (2003). Crossref, PubMed, ISI, Google Scholar, 847–850. DOI: 10.1038/nature01750.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-term Memory." In: *Neural computation* 9 (Dec. 1997), pp. 1735–80. DOI: 10.1162/neco.1997.9.8.1735.

[IY19] Y Ichinohe and S Yamada. "Neural network-based anomaly detection for high-resolution X-ray spectroscopy." In: *Monthly Notices of the Royal Astronomical Society* 487.2 (June 2019), pp. 2874–2880. ISSN: 0035-8711. DOI: 10.1093/mnras/stz1528. eprint: https://academic.oup.com/mnras/article-pdf/487/2/2874/28806952/stz1528.pdf. URL: https://doi.org/10.1093/mnras/stz1528.

[KB14] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization." In: *arXiv e-prints*, arXiv:1412.6980 (Dec. 2014), arXiv:1412.6980. DOI: 10.48550/arXiv.1412.6980. arXiv: 1412.6980 [cs.LG].

[Kno+16] J. Knodlseder et al. "GammaLib and ctools: A software framework for the analysis of astronomical gamma-ray data." In: *Astronomy & Astrophysics* 593 (June 2016). DOI: 10.1051/0004-6361/201628822.

[Kno+19] Knosdlseder, J. et al. "Analysis of the H.E.S.S. public data release with ctools." In: *A&A* 632 (2019), A102. DOI: 10.1051/0004-6361/201936010. URL: https://doi.org/10.1051/0004-6361/201936010.

[KZ15] Pawan Kumar and Bing Zhang. "The physics of gamma-ray bursts & relativistic jets." In: 561 (Feb. 2015), pp. 1–109. DOI: 10.1016/j.physrep.2014.09.008. arXiv: 1410.0679 [astro-ph.HE].

[L.23] Baroncelli L. *Real-Time Anomaly Detection of Gamma-Ray Bursts for the Cherenkov Telescope Array using Deep Learning*. 2023. URL: https://github.com/LeoBaro/Real-Time-Anomaly-Detection-of-Gamma-Ray-Bursts-for-the-Cherenkov-Telescope-Array.

[Lea+16] Colin Lea, René Vidal, Austin Reiter, and Gregory D. Hager. "Temporal Convolutional Networks: A Unified Approach to Action Segmentation." In: *Computer Vision – ECCV 2016 Workshops*. Ed. by Gang Hua and Hervé Jégou. Cham: Springer International Publishing, 2016, pp. 47–54. ISBN: 978-3-319-49409-8.

[LM83] T-P Li and Y-Q Ma. "Analysis methods for results in gamma-ray astronomy." In: *The Astrophysical Journal* 272 (1983), pp. 317–324.

[LTZ08] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation Forest." In: *2008 Eighth IEEE International Conference on Data Mining*. 2008, pp. 413–422. DOI: 10.1109/ICDM.2008.17.

[LB21] M. Lochner and B.A. Bassett. "Astronomaly: Personalised active anomaly detection in astronomical data." In: *Astronomy and Computing* 36 (2021), p. 100481. DOI: 10.1016/j.ascom.2021.100481. URL: https://doi.org/10.1016\%2Fj.ascom.2021.100481.

[MW99] AI MacFadyen and SE Woosley. "Collapsars: gamma-ray bursts and explosions in failed supernovae." In: *Astrophys. J.* 524 (1999), p. 262. DOI: 10.1086/307790.

[MP92] S Mao and B Paczynski. "On the cosmological origin of gamma-ray bursts." In: *Astrophys. J. Lett.* 388 (1992), pp. 45–L48. DOI: 10.1086/186326.

[Mar+20] Berta Margalef-Bentabol, Marc Huertas-Company, Tom Charnock, Carla Margalef-Bentabol, Mariangela Bernardi, Yohan Dubois, Kate Storey-Fisher, and Lorenzo Zanisi. "Detecting outliers in astronomical images with deep generative networks." In: 496.2 (Aug. 2020), pp. 2346–2361. DOI: 10.1093/mnras/staa164710.48550/arXiv.2003.08263. arXiv: 2003.08263 [astro-ph.GA].

[MPP21] Tom Marianer, Dovi Poznanski, and J. Xavier Prochaska. "A semisupervised machine learning search for never-seen gravitational-wave sources." In: 500.4 (Feb. 2021), pp. 5408–5419. DOI: 10.1093/mnras/staa355010.48550/arXiv.2010.11949. arXiv: 2010.11949 [astro-ph.IM].

[Mat+96] J. R. Mattox et al. "The Likelihood Analysis of EGRET Data." In: *The Astrophysical Journal* 461 (1996). "cit. on p. 23", p. 396. DOI: 10.1086/177068.

[Maz+13] Daniel Mazin, Martin Raue, Bagmeet Behera, Susumu Inoue, Yoshiyuki Inoue, Takeshi Nakamori, Tomonori Totani, and CTA Consortium. "Potential of EBL and cosmology studies with the Cherenkov Telescope Array." In: *Astroparticle Physics* 43 (Mar. 2013), pp. 241–251. DOI: 10.1016/j.astropartphys.2012.09.002. arXiv: 1303.7124 [astro-ph.CO].

[Mee+92] CA Meegan, GJ Fishman, RB Wilson, WS Paciesas, GN Pendleton, JM Horack, MN Brock, and C Kouveliotou. "Spatial distribution of gamma-ray bursts observed by BATSE." In: *Nature* 355 (1992), pp. 143–145. DOI: 10.1038/355143a0.

[Ma19] D. Melkumyan and et al. "Prototyping the Resource Manager and Central Control System for the Cherenkov Telescope Array." In: *ICALEPCS'19*. New York, NY, USA, 2019, MOPHA092.

[Mye23] J.D. Myers. *Fermi Gamma-ray Space Telescope*. 2023. URL: https://fermi.gsfc.nasa.gov/science/eteu/catalogs/.

[OC21] Cherenkov Telescope Array Observatory and Cherenkov Telescope Array Consortium. *CTAO Instrument Response Functions - prod5 version v0.1*. 2021. DOI: 10.5281/zenodo.5499840.

[Ong09] R. A. Ong. "Gamma-Ray Astronomy with the Cherenkov Technique." In: *Annual Review of Nuclear and Particle Science* 59 (2009), pp. 465–489.

[Oa17] I. Oya and et al. "Designing and prototyping the control system for the Cherenkov Telescope Array." In: *Proc. ACAT'17*. Seattle, USA, 2017, p. 032045. DOI: 10.1088/1742-6596/1085/3/032045.

[Pac98] B Paczynski. "Are gamma-ray bursts in star-forming regions?" In: *Astrophys. J. Lett.* 494 (1998), p. L45. DOI: 10.1086/311148.

[PY10] Sinno Jialin Pan and Qiang Yang. "A Survey on Transfer Learning." In: *IEEE Transactions on Knowledge and Data Engineering* 22.10 (2010), pp. 1345–1359. DOI: 10.1109/TKDE.2009.191.

[Par+19] German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. "Continual lifelong learning with neural networks: A review." In: *Neural Networks* 113 (2019), pp. 54–71. ISSN: 0893-6080. DOI: https://doi.org/10.1016/j.neunet.2019.01.012. URL: https://www.sciencedirect.com/science/article/pii/S0893608019300231.

[Par+22] N. Parmiggiani, A. Bulgarelli, D. Beneventano, V. Fioretti, A. Di Piano, L. Baroncelli, A. Addis, M. Tavani, C. Pittori, and I. Oya. "The RTApipe framework for the gamma-ray real-time analysis software development." In: *Astronomy and Computing* 39 (2022), p. 100570. ISSN: 2213-1337. DOI: https://doi.org/10.1016/j.ascom.2022.100570. URL: https://www.sciencedirect.com/science/article/pii/S2213133722000178.

[Par+21] N. Parmiggiani, A. Bulgarelli, V. Fioretti, A. Di Piano, A. Giuliani, F. Longo, F. Verrecchia, M. Tavani, D. Beneventano, and A. Macaluso. "A Deep Learning Method for AGILE-GRID Gamma-Ray Burst Detection." In: *The Astrophysical Journal* 914.1 (2021), p. 67. DOI: 10.3847/1538-4357/abfa15. URL: https://dx.doi.org/10.3847/1538-4357/abfa15.

[Par+23] N. Parmiggiani et al. "A Deep-learning Anomaly-detection Method to Identify Gamma-Ray Bursts in the Ratemeters of the AGILE Anticoincidence System." In: 945.2, 106 (Mar. 2023), p. 106. DOI: 10.3847/1538-4357/acba0a.

[PLG16] Rosalba Perna, Davide Lazzati, and Bruno Giacomazzo. "SHORT GAMMA-RAY BURSTS FROM THE MERGER OF TWO BLACK HOLES." In: *The Astrophysical Journal Letters* 821.1 (2016), p. L18. DOI: 10.3847/2041-8205/821/1/L18. URL: https://dx.doi.org/10.3847/2041-8205/821/1/L18.

[PM99] D. Petry and MAGIC Telescope Collaboration. "The MAGIC Telescope - prospects for GRB research." In: 138 (Sept. 1999), pp. 601–602. DOI: 10.1051/aas:1999369. arXiv: astro-ph/9904178 [astro-ph].

[Pru+19] M V Pruzhinskaya, K L Malanchev, M V Kornilov, E E O Ishida, F Mondon, A A Volnova, and V S Korolev. "Anomaly detection in the Open Supernova Catalog." In: *Monthly Notices of the Royal Astronomical Society* 489.3 (Aug. 2019), pp. 3591–3608. ISSN: 0035-8711. DOI: 10.1093/mnras/stz2362. eprint: https://academic.oup.com/mnras/article-pdf/489/3/3591/30029513/stz2362.pdf. URL: https://doi.org/10.1093/mnras/stz2362.

[Sad+17] I. Sadeh, I. Oya, D. Dezman, E. Pietriga, and J. Schwarz. "The Graphical User Interface of the Operator of the Cherenkov Telescope Array." In: *Proc. ICALEPCS'17*. Barcelona, Spain, 2017, pp. 186–191. DOI: 10.18429/JACoWICALEPCS2017-TUBPL06.

[Sad20] Iftach Sadeh. "Data-driven Detection of Multimessenger Transients." In: *The Astrophysical Journal* 894.2 (2020), p. L25. DOI: 10.3847/2041-8213/ab8b5f. URL: https://doi.org/10.3847\%2F2041-8213\%2Fab8b5f.

[Sci] *Science with the Cherenkov Telescope Array*. WORLD SCIENTIFIC, 2018. DOI: 10.1142/10986. URL: https://doi.org/10.1142\%2F10986.

[SA+19] M. Seglar-Arroyo et al. "The gravitational wave follow-up program of the Cherenkov Telescope Array." In: *36th International Cosmic Ray Conference (ICRC2019)* 36 (2019).

[Shi+15] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun Woo. "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting." In: *arXiv e-prints*, arXiv:1506.04214 (June 2015), arXiv:1506.04214. arXiv: 1506.04214 [cs.CV].

[Spu18] Maurizio Spurio. *Probes of Multimessenger Astrophysics. Charged Cosmic Rays, Neutrinos, gamma-Rays and Gravitational Waves*. Springer, 2018.

[Ste00] R. Steenkamp. "H.E.S.S. - An Array of Gamma Ray Telescopes in Namibia." In: *African Skies* (Jan. 2000).

[Tag+22] G. Tagliaferri et al. "The small-sized telescope of CTAO." In: *Ground-based and Airborne Telescopes IX*. Ed. by Heather K. Marshall, Jason Spyromilio, and Tomonori Usuda. Vol. 12182. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series. Aug. 2022, 121820K, 121820K. DOI: 10.1117/12.2627956.

[Tam20] S. Tampieri. "Real time aperture photometry with Cherenkov Telescope Array." In: (2020).

[Tan+18] Zhiyi Tang, Zhicheng Chen, Yuequan Bao, and Hui Li. "Convolutional neural network-based data anomaly detection method using multiple information for structural health monitoring." In: *Structural Control and Health Monitoring* 26 (Nov. 2018). DOI: 10.1002/stc.2296.

[Tav+09] M. Tavani et al. "The AGILE Mission." In: 502.3 (Aug. 2009), pp. 995–1013. DOI: 10.1051/0004-6361/200810527. arXiv: 0807.4254 [astro-ph].

[The00] The HDF Group. *Hierarchical data format version 5*. 2000-2010. URL: http://www.hdfgroup.org/HDF5.

[TC02] Vyacheslav P. Tuzlukov and Cheng. *Signal Processing Noise*. USA: CRC Press, Inc., 2002. ISBN: 0849310253.

[VBK19] Alireza Vafaei Sadr, Bruce A. Bassett, and Martin Kunz. "A Flexible Framework for Anomaly Detection via Dimensionality Reduction." In: *arXiv e-prints*, arXiv:1909.04060 (Sept. 2019), arXiv:1909.04060. DOI: 10.48550/arXiv.1909.04060. arXiv: 1909.04060 [cs.LG].

[Vas+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. "Attention is All you Need." In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[Ver16] Stefano Vercellone. "The ASTRI mini-array within the future Cherenkov Telescope Array." In: *European Physical Journal Web of Conferences*. Vol. 121. European Physical Journal Web of Conferences. July 2016, 04006, p. 04006. DOI: 10.1051/epjconf/201612104006. arXiv: 1508.00799 [astro-ph.IM].

[Web+20] Sara Webb, Michelle Lochner, Daniel Muthukrishna, Jeff Cooke, Chris Flynn, Ashish Mahabal, Simon Goode, Igor Andreoni, Tyler Pritchard, and Timothy M. C. Abbott. "Unsupervised machine learning for transient discovery in deeper, wider, faster light curves." In: 498.3 (Nov. 2020), pp. 3077–3094. DOI: 10.1093/mnras/staa239510.48550/arXiv.2008.04666. arXiv: 2008.04666 [astro-ph.IM].

[Wee+02] T. C. Weekes, H. Badran, S. Biller, and et al. "The Very Energetic Radiation Imaging Telescope Array System (VERITAS): A Status Report." In: *Astroparticle Physics* 17.2 (2002), pp. 221–243.

[WK94] Gerhard Widmer and M. Kubat. "Learning in the Presence of Concept Drift and Hidden Contexts." In: *Machine Learning* 23 (Nov. 1994). DOI: 10.1007/BF00116900.

[Wil38] S. S. Wilks. "The Large-Sample Distribution of the Likelihood Ratio for Testing Composite Hypotheses." In: *Ann. Math. Statist.* 9.1 (1938). "cit. on p. 24", pp. 60–62. DOI: 10.1214/aoms/1177732360.

[ZZ18] Runtian Zhang and Qian Zou. "Time Series Prediction and Anomaly Detection of Light Curve Using LSTM Neural Network." In: *Journal of Physics: Conference Series* 1061.1 (2018), p. 012012. DOI: 10.1088/1742-6596/1061/1/012012. URL: https://dx.doi.org/10.1088/1742-6596/1061/1/012012.