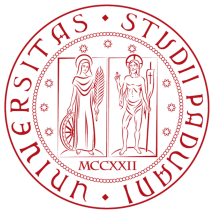

UNIVERSITÀ DEGLI STUDI DI PADOVA

Dipartimento di Fisica e Astronomia

Corso di Laurea Triennale in Fisica

Tesi di Laurea

Inverse Beta Decay events selection in JUNO using Machine Learning algorithms

Relatore

prof. Alberto Garfagnini

Correlatori

dott. XXXXXX

dott. XXXXXX

Laureando

Candidate

Badge Number

Anno Accademico YYYY/YYYY

Discussion date

Abstract

The Jiangmen Underground Neutrino Observatory (JUNO) will be the largest liquid scintillator-based neutrino detectors in the World, for the next decade. Thanks to its large active mass (20 kt) and state of the art performances (3% effective energy resolution at 1 MeV), it will be able to perform important measurements in neutrino physics. The proposed thesis will study the application of different Machine Learning inspired algorithms for the discrimination of signal events (interactions of anti-neutrinos coming from the nearby nuclear power plants) from background events.

Contents

Chapter 1

Introduction

The Jiangmen Underground Neutrino Observatory (JUNO), currently under construction in southern China, is a large liquid scintillator neutrino detector. It is designed to detect electron antineutrino interactions produced by nearby Nuclear Power Plants (NPP) through the inverse beta decay reaction. The primary objective of this experiment is to determine the neutrino mass hierarchy, thereby addressing the Neutrino Mass Ordering (NMO) problem.

The field of neutrino physics has entered a new era of precision following the measurement of the third lepton mixing angle, the so-called reactor angle θ_{13}. This has had a significant impact on models of neutrino mass and mixing. The JUNO experiment, with its excellent energy resolution and large fiducial volume, is expected to make significant contributions to this field.

This leads us to the theory of neutrino oscillation, a quantum mechanical phenomenon whereby a neutrino created with a specific lepton flavor can later be measured to have a different flavor. The oscillation is quantified in terms of parameters that the JUNO experiment aims to measure with high precision.

1.1 Neutrinos Oscillation

The Standard Model of elementary particle interactions provides an accurate description of strong, weak, and electromagnetic interactions, but it treats these interactions as distinct and unrelated. Within this framework, neutrinos are assumed to be massless, but this assumption has been called into question by physicists. Neutrino oscillations, which occur when neutrinos change from one flavor to another, are a potential indication of neutrino mass.
The term "neutrino oscillations" refers to this phenomena and it involve the conversion of a neutrino of a particular flavor to another as it propagates through space.

Each known flavor eigenstate, $(\nu_e, \nu_\mu, \nu_\tau)$, linked to three respective charged leptons (e, μ, τ) via the charged current interactions can be considered a complex combination of neutrino mass eigenstates as follow:

$$\begin{pmatrix} v_e \\ v_\mu \\ v_\tau \end{pmatrix} = U_{\text{PMNS}} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix}$$

in wich ν_i are the three mass eigensates, that have 3 masses m_i $(i = 1, 2, 3)$, which are non-degenerate, with $m_i \neq m_j$ for $i \neq j$.

The matrix U_{PMNS}, called Pontecorvo-Maki-Nakagawa-Sakata (PMNS) matrix, is composed of three rotation matrices, R_{23}, R_{13}, and R_{12}, each corresponding to a different mixing angle, θ_{23}, θ_{13}, and θ_{12}, respectively and a parameter δ_{CP} called the Dirac CP-violating phase. For this case, the Majorana CP phases are $\eta_i (i = 1, 2)$, which are only physically possible if neutrinos are Majorana-type particles and do not participate in neutrino oscillations. Therefore, U can be expressed as:

$$
\begin{aligned}
U_{\text{PMNS}} &= \\
&= \begin{pmatrix} 1 & 0 & 0 \\ 0 & c_{23} & s_{23} \\ 0 & -s_{23} & c_{23} \end{pmatrix} \begin{pmatrix} c_{13} & 0 & s_{13}\mathrm{e}^{-\mathrm{i}\delta_{CP}} \\ 0 & 1 & 0 \\ -s_{13}\mathrm{e}^{\mathrm{i}\delta_{CP}} & 0 & c_{13} \end{pmatrix} \begin{pmatrix} c_{12} & s_{12} & 0 \\ -s_{12} & c_{12} & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathrm{e}^{\mathrm{i}\eta_1} & 0 & 0 \\ 0 & \mathrm{e}^{\mathrm{i}\eta_2} & 0 \\ 0 & 0 & 1 \end{pmatrix}
\end{aligned}
$$

where $s_{ij} \equiv \sin\theta_{ij}, c_{ij} \equiv \cos\theta_{ij}$.

The theoretical framework for neutrino oscillations involves the calculation of the oscillation probability as a function of the distance traveled by the neutrino, the neutrino mixing matrix, and the difference in squared masses between the three neutrino mass states, $\Delta m^2_{ij} = m^2_i - m^2_j$ for $i, j = 1, 2, 3, i > j$. Specifically, two nuclear power reactors 53 km away from the detector, which mostly produce anti-electron neutrinos $\bar{\nu}_e$ with energy below 10 MeV, are the principal sources of neutrinos for the JUNO experiment. So, it is necessary for the JUNO experiment to calculate the survival probability $P(\bar{\nu}_e \to \bar{\nu}_e)$ of electron antineutrinos.

$$
P(\bar{\nu}_e \to \bar{\nu}_e) = 1 - \sin^2 2\theta_{12} c^4_{13} \sin^2\left(\frac{\Delta m^2_{21} L}{4\mathcal{E}}\right) - \sin^2 2\theta_{13}\left[c^2_{12}\sin^2\left(\frac{\Delta m^2_{31} L}{4\mathcal{E}}\right) + s^2_{12}\sin^2\left(\frac{\Delta m^2_{32} L}{4\mathcal{E}}\right)\right]
$$

where $\mathcal{E}$ is the neutrino energy, L the travelled distance and $\Delta m^2_{ij} \equiv m^2_i - m^2_j$.
Past experiments have already given estimates for $\Delta m^2_{21}, |\Delta m^2_{31}|$ and the 3 mixing angles.

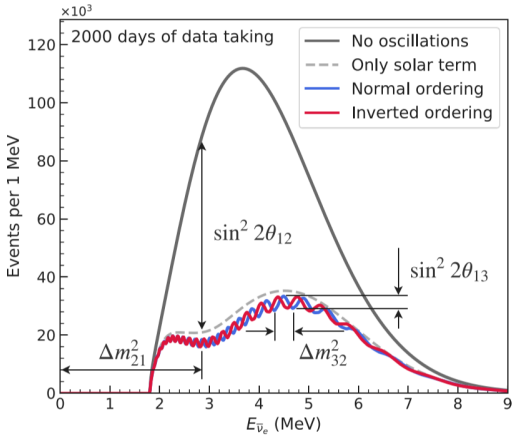

Figure 1.1: JUNO's reactor antineutrino energy spectrum is shown with and without the effect of neutrino oscillation. The gray dashed curve includes only the term in the disappearance probability modulated by $sin^2(2\theta_{12})$, while the blue and red curves use the full oscillation probability for normal and inverted mass orderings. Spectral features driven by oscillation parameters are illustrated, highlighting the rich information available in JUNO's high-resolution measurement of the oscillated spectrum.

JUNO's primary objective is to refine these results, particularly to ascertain the sign of Δm^2_{31}, which will distinguish between two potential scenarios: Normal Ordering (NO), where $|\Delta m^2_{31}| = |\Delta m^2_{32}| + |\Delta m^2_{21}|$ and the mass hierarchy is $m_1 < m_2 < m_3$, and Inverted Ordering (IO), where $|\Delta m^2_{31}| = |\Delta m^2_{32}| - |\Delta m^2_{21}|$ and the mass hierarchy is $m_3 < m_1 < m_2$. The sign of Δm^2_{31} subtly alters the plot of 1.1. However, it remains uncertain whether the ν_3 neutrino mass eigenstate is heavier or lighter than the ν_1 and ν_2 mass eigenstates.

1.2 The JUNO detector

Nestled beneath the Dashi hill in Jinji town, Southern China, the Jiangmen Underground Neutrino Observatory (JUNO) is an ongoing experiment. Its placement 43 km southwest of Kaiping city was strategically chosen to significantly reduce background noise from cosmic rays due to its underground location. JUNO is anticipated to detect a plethora of antineutrinos, predominantly originating from the Taishan and Yangjiang nuclear power plants (NPPs). These power plants are approximately 52.5 km away from the JUNO detector and together, they have a combined nominal thermal power of 26.6 GW_{th}. The detector's design has been meticulously optimized for the highest sensitivity to the ordering of neutrino masses.

A schematic illustration of JUNO is presented in Fig.1.2.

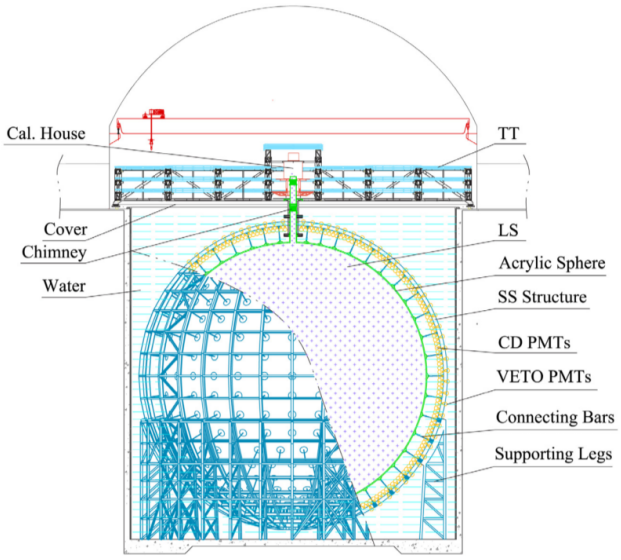

Figure 1.2: Schematic view of the JUNO experiment

Furthermore, the JUNO experiment deploys a specialized compact detector named TAO. Situated approximately 30 meters from one of the Taishan reactors, TAO serves to measure the unoscillated reactor antineutrino spectrum shape precisely. The data collected by TAO is intended to provide a crucial data-driven input to refine the spectra from the other reactor cores. The core of the JUNO detector, the **Central Detector (CD)**, is complemented by a water **Cherenkov detector** and a **Top Tracker (TT)**. Notably, the CD, designed as a compact, non-segmented detector, boasts an effective energy resolution of $\sigma_E/E = 3\%/\sqrt{E(MeV)}$, a testament to the advantage of opting for a compact design over a segmented one.

The CD contains a 20 kton liquid scintillator (LS), safely housed within a spherical acrylic vessel and submerged in a water pool. The pool, with a diameter of 43.5 m and a height of 44 m, provides an adequate buffer to shield the LS from the radioactive influence of the surrounding rock.

The vessel is supported by a stainless steel (SS) structure through connecting bars. Additional CD PMTs are mounted on the inner surface of this structure, which also hosts compensation coils designed to mitigate the Earth's magnetic field and thereby minimize its impact on the photoelectron collection efficiency of the PMTs.

Above the water pool resides the Top Tracker, an assembly of a plastic scintillator array, meticulously arranged to measure muon tracks accurately. The CD is connected to the external environment through a chimney, which facilitates calibration operations. Located above this chimney is the Calibration House, equipped with special radioactivity shielding and a muon detector, playing a crucial role in the overall experimental setup.

1.3 JUNO signals and backgrounds

1.3.1 Signal

The JUNO experiment primarily draws its sources from the Taishan and Yangjiang Nuclear Power Plants (NPPs), which house two and six cores respectively. In addition to these, the Daya Bay reactor complex contributes to the antineutrino flux. The reactor power, baselines, and anticipated Inverse Beta Decay (IBD) rates from the Taishan, Yangjiang, and Daya Bay reactor cores are detailed in Table 1.1.

Reactor	Power [GW_{th}]	Baseline [Km]	IBD Rate [day^{-1}]
Taishan	9.2	52.71	15.1
Yangjiang	17.4	52.46	29.0
Daya Bay	17.4	215	3

Table 1.1: Information on nuclear reactors

JUNO employs a Liquid Scintillator (LS) primarily composed of Linear Alkyl-Benzene (LAB), known for its transparency, high flash points, robust light yield, and low chemical reactivity. The LS, with a density of 0.859 g/mL, is further enhanced with 3 g/L of 2,5-diphenyloxazole (PPO) as the fluor, and 15 mg/L of p-bis-(o-methylstyryl)-benzene (bis-MSB) as the wavelength shifter. The scintillator is doped with a small amount of gadolinium, increasing its sensitivity to antineutrinos via the inverse beta decay (IBD) process.

This process is initiated when an antineutrino interacts with a proton in the liquid scintillator, producing a positron and a neutron. It can be described by the following reaction:

$$\overline{\nu}_e + p \rightarrow e^+ + n \tag{1.1}$$

The positron, carrying the majority of the antineutrino's initial energy, deposits this energy in the scintillator through ionization. This energy deposition, coupled with the positron's subsequent annihilation typically into two 0.511 MeV photons, forms the prompt signal, characterized as follow: $e^+ + e^- \rightarrow 2\gamma$. The energy deposited by the positron directly correlates with

the antineutrino energy, providing a precise measure critical for neutrino oscillation studies.

Following the prompt signal, the neutron is captured primarily on hydrogen (approximately 99% of the time) after an average delay of about 220 µs. This capture event releases a single 2.2 MeV photon, creating the delayed signal. Occasionally, the neutron is captured on carbon (around 1% of the time), resulting in a gamma-ray signal with a total energy of 4.9 MeV. Despite carrying only a small fraction of the initial antineutrino energy, typically from zero to a few tens of keV, neutron recoils are considered in the calculations due to JUNO's exceptional energy resolution.

The light output from these events is detected by the photomultiplier tubes (PMTs). PMTs are sensitive detectors that convert light into an electrical signal. They operate based on the photoelectric effect and subsequent electron multiplication. When a photon hits the photocathode (the light-sensitive surface inside the PMT), it can eject an electron through the photoelectric effect. This electron is then accelerated by an electric field towards a series of electrodes called dynodes. Each time an electron hits a dynode, more electrons are released. This process is repeated multiple times, resulting in a cascade of electrons and a significant amplification of the original signal. The final electrical signal, which can be easily measured, is proportional to the number of photons that hit the photocathode.

The PMTs detect the light and convert it into an electrical signal. The signals from all the PMTs are then combined to reconstruct the position and energy of the original neutrino interaction. This technique allows JUNO to measure the energy of the incoming neutrino to high precision, which is crucial for studying neutrino oscillation.

1.3.2 Backgrounds

The design and composition of the scintillator in the JUNO experiment are meticulously optimized to minimize background noise from various radiation sources, such as cosmic rays and natural radioactivity. Despite these efforts, several types of background signals are inevitably produced in the detector. For the purpose of analysis, we focus primarily on the three most significant contributors:

Radiogenic Backgrounds

Radiogenic backgrounds in the JUNO experiment primarily originate from the radioactive decay of isotopes such as ^{238}U, ^{232}Th, and ^{40}K. These isotopes are naturally present in the materials comprising the JUNO detector, including acrylic used for the detector walls, the metal structure supporting the detector, PMT glass, the gas during early filling phases, and the surrounding water. They are also found in the surrounding rock. These isotopes undergo radioactive decay, emitting various forms of radiation. The decay of ^{238}U and ^{232}Th occurs through decay chains, where each isotope successively decays into different isotopes, releasing radiation in the process. The emitted radiation includes alpha particles, beta particles, and gamma rays. As for ^{40}K, it undergoes beta decay to ^{40}Ca or electron capture to ^{40}Ar, with a small fraction (0.001%) resulting in the emission of a gamma ray. These radiogenic backgrounds need to be carefully accounted for and minimized to accurately detect reactor antineutrinos in the JUNO experiment.

These radiogenic backgrounds can potentially mimic the signal from inverse beta decay (IBD) in several ways:

1. **Beta decays and electron captures**: These processes result in the emission of electrons or positrons, which can produce a scintillation signal similar to the prompt signal from IBD.

2. **Gamma rays**: High-energy gamma rays can Compton scatter in the detector, producing electrons with enough energy to mimic the prompt signal from IBD. In addition, gamma rays can produce electron-positron pairs in the detector, which can mimic both the prompt and delayed signals from IBD.

3. **Neutrons**: Some decays in the ^{238}U and ^{232}Th chains emit neutrons, which can be captured on protons in the detector, mimicking the delayed signal from IBD.

Cosmogenic Backgrounds

Cosmogenic backgrounds in JUNO primarily result from the interaction of cosmic rays, particularly high-energy muons (~ GeV), with the detector materials. These interactions lead to the production of fast neutrons and unstable isotopes through the process of spallation in which a high-energy particle strikes a target atom, causing it to emit smaller particles such as neutrons and unstable isotopes. Specifically, these muons interact with the detector materials, resulting in the production of isotopes like ^{9}Li, ^{8}He and ^{11}C, which are unstable and subsequently decay, contributing to additional background events.

These fast neutrons and unstable isotopes, produced from the interactions of muons with the detector materials, can generate signals that mimic an inverse beta decay (IBD) event. Specifically, there are two distinct signals to consider.

The first, known as the prompt signal, is generated by an electron. The energy and momentum of this electron can make it appear like a positron, the particle that would be expected in an IBD event. The second, known as the delayed signal, is generated by a neutron. This neutron can be captured by a proton in the detector, producing a signal identical to what would be expected from the neutron in an IBD event.

Therefore, even though they are not IBD events, these signals mimic an IBD event as they consist of an electron being mistaken for a positron in the prompt signal and a true neutron in the delayed signal.

Other source of $\overline{\nu}_e$

Other sources of antineutrinos also contribute to the background. Those are geoneutrinos, atmospheric neutrinos, and reactor antineutrinos:

Geoneutrinos are antineutrinos produced by natural radioactivity within the Earth, primarily below 2.5 MeV in antineutrino energy. Natural radioactivity exists in materials present in the Earth's crust and mantle, such as U, Th,and ^{40}K. These materials undergo radioactive decays, generating antineutrinos as decay products, that produce IBD signals.

Atmospheric neutrinos are generated by interactions of cosmic rays with the Earth's atmosphere. When high-energy cosmic rays collide with the atmosphere, they produce a cascade of particles, including muons and neutrinos. The muons generated in these interactions can

decay, producing antineutrinos.

Reactor antineutrinos are an artificial source of antineutrinos. Besides the reactors that are used to generate the signal to be analyzed, there are various other reactors that contribute to the total event count. Given the vast number of nuclear reactors worldwide, the collective signal from these reactors becomes significant.

It's beneficial to categorize the aforementioned background sources into two distinct groups:

- *Accidental Background*: This category includes background events that result from the coincidence of two independent events, typically of radiogenic origin. These events primarily influence the low-energy region of the spectrum. A portion of the cosmogenic background also falls into this category. The goal of this thesis work is to significantly reduce these accidental background events, a topic that will be discussed in detail in the following sections
- *Correlated Backgrounds*: These backgrounds originate from a single physics process and produce both a prompt and a delayed signal. Significant correlated backgrounds include cosmogenic Li/He and fast neutrons. Among all the radiogenic processes, only one correlated background requires consideration: the C(α,n)^{16}O, decay that produces an alpha particle (prompt signal) and a neutron that is captured as delayed, exactly like an IBD, occurring within the liquid scintillator.

Here a viasualization sumary of all the bacgrounds contributions:

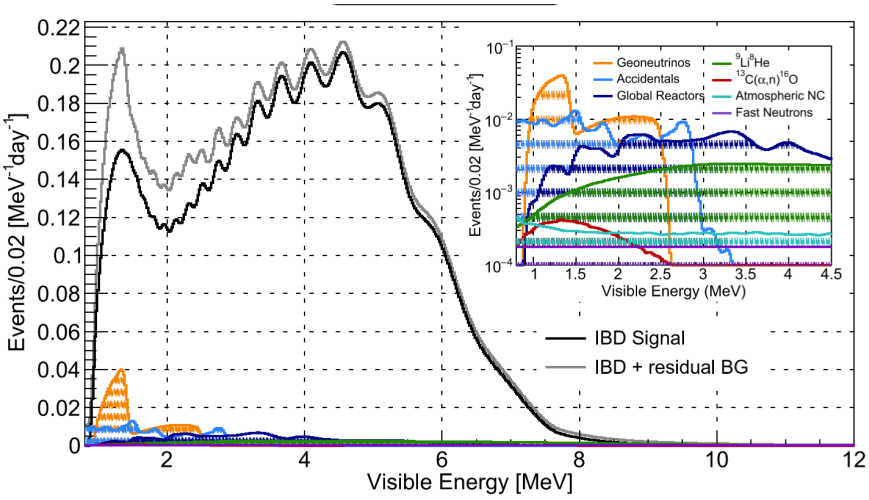

Figure 1.3: Visible energy spectrum as measured by the LPMT system with (grey) and without (black) backgrounds is that which is anticipated for JUNO. The predicted backgrounds, which make up around 7% of the whole sample of IBD candidates and are primarily confined below, are shown in the inset as spectra. $\approx$ 3 MeV

Following the comprehensive discussion of all background events in the JUNO experiment, it becomes clear that due to the significant presence of various types of background events, efforts are being made to reduce their contribution in every possible way. Several strategies have been employed to mitigate these background signals. Methods include the use of shielding materials to block external radiation, careful selection and treatment of detector materials to

minimize internal radioactivity, and sophisticated data analysis techniques to identify and reject background events.

However, it's important to note that a large portion of the accidental background events are the only ones where significant reduction can be achieved. These are the events that occur randomly and independently, and their reduction requires a different approach compared to correlated backgrounds. The focus of this thesis is precisely on these accidental background events, exploring strategies and techniques to further minimize their impact on the experiment. This is a crucial aspect of the experiment's success, as reducing these events can significantly improve the sensitivity and accuracy of the neutrino measurements.

Chapter 2

Frameworks

2.1 Introduction to Machine Learning

Machine learning is a powerful tool that can be used to identify patterns in complex datasets. In the context of particle physics, machine learning algorithms can be used to detect signals from background noise in large datasets generated by detectors. In particular, for the detection of IBD signals from background, machine learning algorithms can be used to identify patterns in the data that are indicative of an IBD event, and to distinguish these signals from the background noise explained above. Moreover, one advantage of machine learning for particle physics is that it can handle large amounts of data and identify subtle patterns that may be difficult for humans to detect.

2.1.1 Supervised Learning

Supervised learning is a machine learning technique in which the algorithm is trained on a labelled dataset, where the input data is accompanied by the correct output. The goal of the algorithm is to learn a function that can map input data to output data. Some examples of supervised learning algorithms include linear regression, logistic regression, decision trees, and support vector machines. Despite the complexity and diversity of these methods, it's more advantageous to illustrate the profound concepts of machine learning through a simple machine learning algorithm, such as linear regression.

2.1.2 Linear Regression

Linear regression is a type of supervised learning algorithm used in machine learning for predictive analysis. It is used to model the relationship between a dependent variable, called the target, and one or more independent variables, called the features.
The basic idea behind linear regression is to find the best-fitting hyper-plane that describes the relationship between the independent and dependent variables. The equation for the hyper-plane can be written as:

$$y = w_0 + w_1 x_1 + w_2 x_2 + ... + w_n x_n \tag{2.1}$$

where

- y is the dependent variable
- $x_1, x_2, ..., x_p$ are the independent variables
- $w_0, w_1, w_2, ..., w_n$ are the coefficients or parameters of the model

In order to determine the values of the coefficients $w_0, w_1, w_2, ..., w_n$, a common approach is to minimize a loss function, which measures the difference between the predicted values of the dependent variable and the actual values. The most commonly used loss function in linear regression is the mean squared error (MSE) function, which is defined as:

$$L(w_0, w_1, w_2, ..., w_n) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2.2}$$

where n is the number of observations, y_i is the actual value of the dependent variable for the i-th observation, and $\hat{y}_i$ is the predicted value of the dependent variable for the i-th observation.

The objective of linear regression is to determine the optimal values of coefficients $w_0, w_1, w_2, ..., w_n$ that minimize a predefined loss function $L(w_0, w_1, w_2, ..., w_n)$. One commonly employed method to accomplish this is the gradient descent algorithm.

Gradient descent is an iterative optimization technique for finding the local minimum of a function. To apply gradient descent in the context of a linear regression problem, we initialize the coefficients with random values and then iteratively update these values in the direction that decreases the loss function the most.

Mathematically, the update rule for each coefficient is:

$$w_j^{(new)} = w_j^{(old)} - \alpha\frac{\partial L}{\partial w_j} \tag{2.3}$$

where $w_j^{(new)}$ and $w_j^{(old)}$ are the new and old values of the j-th coefficient, α is the learning rate, and $\frac{\partial L}{\partial w_j}$ is the partial derivative of the loss function with respect to the j-th coefficient. The learning rate α determines the size of the steps we take towards the minimum.

Once we reach a point where the loss function no longer decreases (or decreases very slowly), we stop the iteration and accept the current values of coefficients as the solution.

However, it is important to note that linear regression, and also other machine learning alghoritms can suffer from overfitting or underfitting. Overfitting occurs when the model is too complex and captures noise in the data, while underfitting occurs when the model is too simple and fails to capture the underlying patterns in the data. To prevent overfitting or underfitting, regularization techniques can be used.

2.2 Binary Classification

The binary classification problem is a fundamental task in supervised learning, a branch of machine learning. It involves classifying elements of a dataset into one of two possible groups based on a set of features. For instance, a binary classification problem could involve determining whether an email is spam or not spam, whether a transaction is fraudulent or legitimate, or whether a tumor is malignant or benign.

To tackle the binary classification problem, various machine learning algorithms can be employed. Two common approaches are *Decision Trees* and *Neural Networks*.

2.3 Decision Tree

A Decision Tree algorithm, used in supervised machine learning for classification and regression tasks, models the predictive outcome of a target variable based on decision rules inferred from input features. The process involves dissecting the overall dataset into distinct regions, where each region contains data points that are as similar as possible to each other in terms of their target class. Formally, each internal node of the tree represents a decision rule based on an input feature, which bifurcates the data into two child nodes. The decision for splitting the data at each node is determined using a metric known as Information Gain, which in turn is based on the concept of Entropy.

In the context of a binary classification, entropy (H) is a measure of the impurity or disorder within a set (S) of instances. It quantifies the uncertainty involved in predicting the class of a random instance from the set (S). It is mathematically formulated as:

$$H(S) = -p_+ \log_2(p_+) - p_- \log_2(p_-) \tag{2.4}$$

Here, p_+ and p_- denote the proportions of positive and negative instances in the set S, respectively. Entropy attains a maximum value when the set S contains an equal number of positive and negative instances, reflecting the highest uncertainty.

Information Gain (IG) measures the reduction in entropy achieved by partitioning the instances based on a feature (A). It is the difference between the entropy of the set before the split (H(S)) and the weighted sum of the entropies of each subset resulting from the split. It can be formulated as:

$$IG(S, A) = H(S) - \sum_{v \in V(A)} \left(\frac{|S_v|}{|S|} \right) H(S_v) \tag{2.5}$$

where $V(A)$ indicates the set of all possible values of feature A.
In this equation, S_v denotes the subset of instances in S for which the feature A takes on the value v. $|S_v|$ and $|S|$ are the cardinalities of the sets S_v and S, respectively.

The algorithm constructs the tree by recursively applying these splits, each time selecting the feature that results in the maximum information gain. This process continues until a stopping criterion is met, such as reaching a pre-specified maximum depth of the tree or a minimum number of samples per leaf.

While Decision Trees are straightforward and practical models, their ability to decipher complex patterns in data can be limited. This limitation paves the way for a more advanced

technique known as Gradient Boosting Decision Trees. This innovative method combines several simple trees into a robust model that can effectively interpret intricate patterns in complex datasets. The following sections will delve into the mechanics of Gradient Boosting Decision Trees and their enhanced ability to handle complex data structures.

2.3.1 Gradient Boosting Decision Trees

Gradient Boosting is a machine learning algorithm that stems from the concept of boosting, with the application of gradient descent methodology. Its goal is to produce a robust predictive model through the combination of multiple weak learners, typically decision trees.

The primary innovation in Gradient Boosting over classical boosting techniques is its approach to error correction. Instead of modifying the weights of misclassified instances, Gradient Boosting fits each new tree to the residuals (or the negative gradient) of the loss function with respect to the prediction of the existing ensemble of trees. This means each new tree is trained to predict the error of the existing model, thereby iteratively reducing the overall error.

Let's formalize this process:

1. **Initialization**: We begin by initializing our model with a constant value. This is denoted as $F_0(x) = \arg\min_\gamma \sum_{i=1}^{N} L(y_i, \gamma)$, where $L(y, F(x))$ represents the loss function, y represents the true target value, and $F(x)$ is the model's prediction for the input features x. This constant prediction, γ, is chosen to minimize the total loss over all N instances. Thus, our initial model starts with a prediction that globally minimizes the loss.

2. **Computation of Residuals**: Next, we iteratively construct an ensemble of M trees. For each iteration $m = 1$ to M, we calculate the residuals as

$$r_{im} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)} \tag{2.6}$$

   for each instance $i = 1, 2, ..., N$. These residuals are essentially the negative gradients (or first derivatives) of the loss function with respect to the model's predictions. They provide a measure of the direction that would decrease the loss function fastest.

3. **Fitting a Decision Tree**: After computing the residuals, we fit a new decision tree, $h_m(x)$, to these residuals. This tree is thus trained to predict the negative gradient of the loss function, using train it using the training set $(x_i, r_{im})_{i=1}^{n}$. By doing so, it attempts to correct the errors made by the current ensemble model.

4. **Model Update**: The model is then updated by applying the rule

$$F_m(x) = F_{m-1}(x) + \nu \cdot h_m(x) \tag{2.7}$$

   Here, ν represents the learning rate, a parameter typically less than 1, which controls the contribution of each tree to the final prediction. This essentially adjusts the previous model's prediction in the direction that most decreases the loss.

5. **Final Model**: The final model's prediction is given by $F_M(x) = F_0(x) + \sum_{m=1}^{M} \nu \cdot h_m(x)$. In the final ensemble model, each decision tree provides a small correction to the predictions of the previous trees, collaboratively reducing the loss function's value and improving the overall model's performance.

In XGBoost, an advanced implementation of Gradient Boosting, additional improvements such as the introduction of regularization terms to prevent overfitting, computation of the second-order gradient for faster convergence, and mechanisms to handle missing values and enable parallel processing, are present.

An advanced and highly efficient implementation of this method is XGBoost, which introduces several improvements such as regularization terms in the objective function to prevent overfitting, the computation of the second-order gradient for faster convergence, and built-in mechanisms to handle missing values and enable parallel processing.

Here a grafic representation of the process discussed above:

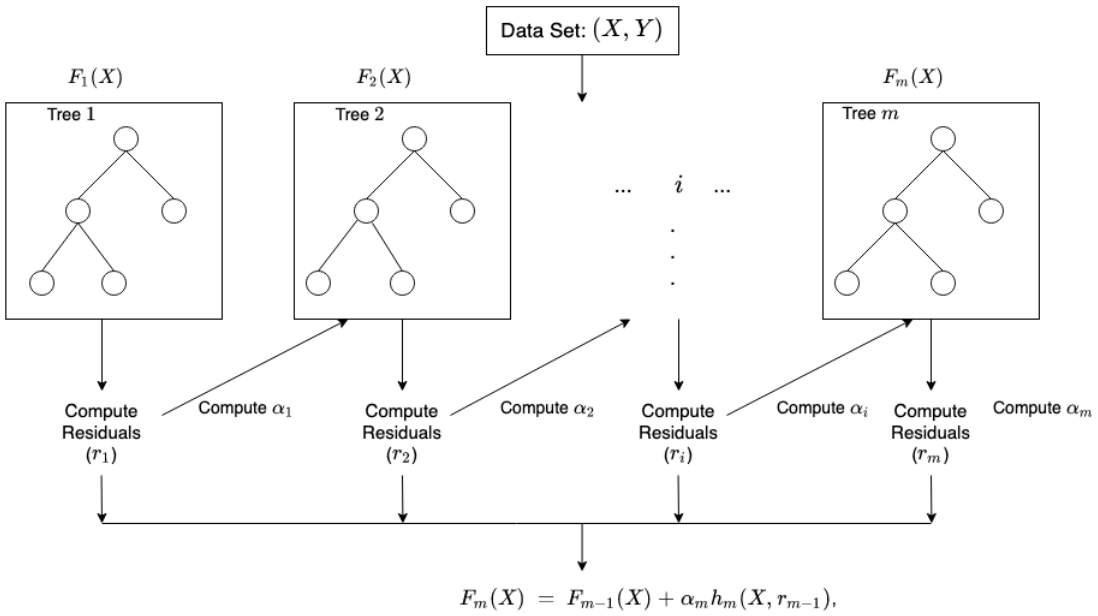

Figure 2.1: A concise illustration of how Gradient Boosted Trees work.

2.4 Neural Networks

Artificial Neural Networks (ANNs) are computational models that draw inspiration from the interconnected structure of the human brain. Each individual computational unit, often referred to as an "artificial neuron" or simply "neuron", is designed to mimic the fundamental working mechanism of a biological neuron.

Let's denote the inputs to an artificial neuron as $x = [x_1, x_2, ..., x_n]$, a representation analogous to dendrites in a biological neuron. These inputs are linearly transformed by a set of weights, $w = [w_1, w_2, ..., w_n]$, summed together, and a bias term, b, is added to the result. This operation can be expressed mathematically as:

$$z = \sum_{i=1}^{n} w_i x_i + b$$

The calculated value, z, is then passed through an activation function, f, to generate the neuron's output, a. This process can be viewed as the equivalent of the activation potential and

the subsequent firing (or not firing) in a biological neuron. The mathematical representation is as follows:

$$a = f(z)$$

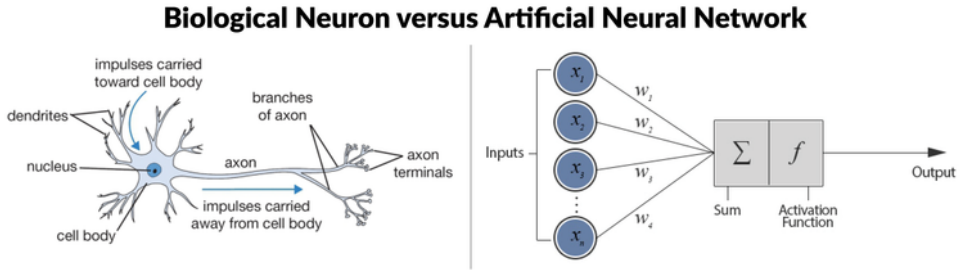

Figure 2.2

The activation function introduces non-linearity into the model, which is crucial for the network's ability to learn complex patterns. Common choices for f include the sigmoid, hyperbolic tangent, and ReLU (Rectified Linear Unit) functions.

An Artificial Neural Network builds upon the concept of the artificial neuron to form an interconnected assembly of these neurons, structured in layers. An ANN typically comprises an input layer, one or more hidden layers, and an output layer. Each layer may contain one or more neurons, and the layers are fully connected, meaning every neuron in one layer connects with all neurons in the following layer.

Follows a grafic rapresentation of a ANN:

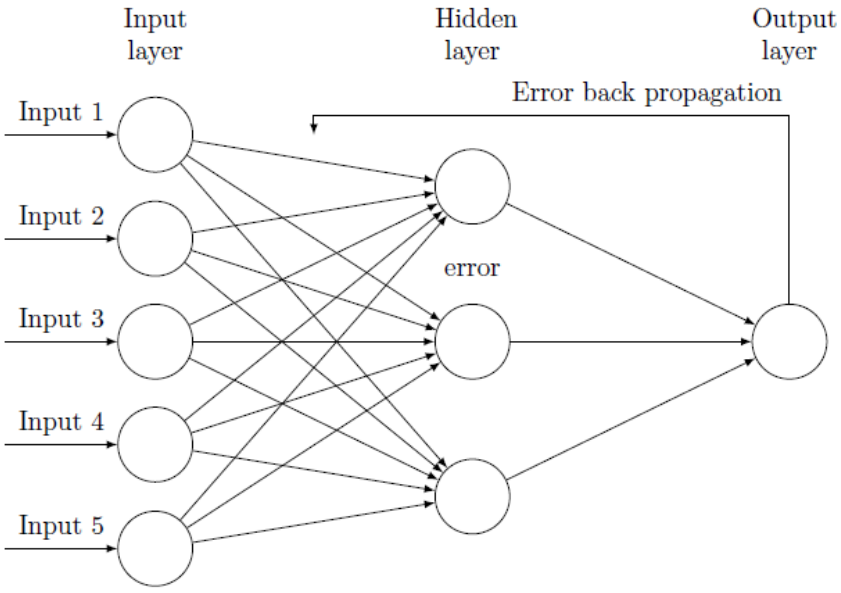

Figure 2.3

For classification problems, the output layer typically uses a softmax function for multi-class problems to output a probability distribution over the classes, or a sigmoid function for binary classification problems to provide the probability of the positive class.

Training a neural network involves a two-step process: forward propagation and backpropagation. In forward propagation, the input is passed through the network to generate an output. This output is then compared with the actual target to compute the loss function L.

Backpropagation uses the chain rule of calculus to compute the gradient of L with respect to the network's parameters, which are then used to update the weights and biases:

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial a}\frac{\partial a}{\partial z}\frac{\partial z}{\partial w}$$

Here, $\frac{\partial L}{\partial a}$ is the derivative of the loss function with respect to the activation output, $\frac{\partial a}{\partial z}$ is the derivative of the activation function, and $\frac{\partial z}{\partial w}$ is the derivative of the weighted sum with respect to the weights.

Once these gradients are calculated, they are used to update the weights and biases via gradient descent, a process that iteratively adjusts the parameters to minimize the loss function:

$$w_{\text{new}} = w_{\text{old}} - \alpha\frac{\partial L}{\partial w}$$

$$b_{\text{new}} = b_{\text{old}} - \alpha\frac{\partial L}{\partial b}$$

In these equations, α is the learning rate, a hyperparameter that determines the size of the steps the algorithm takes down the gradient towards the minimum.

The interconnected structure of ANNs, combined with the ability of backpropagation and gradient descent to effectively adjust the model parameters, allows these networks to learn and represent complex, non-linear relationships in the data.

Chapter 3

Analysis

3.1 Datasets

In the context of this research, I have been granted access to two distinct generated datasets, produced utilizing SNIPER, a leading-edge simulation tool deployed within the framework of the JUNO experiment.

The first of the datasets provided is specifically tailored for the study of Inverse Beta Decay (IBD) events. Each event within this dataset, simulated and injected into the system, is tagged with a unique Simulation Identifier, or SimID. Furthermore, events which trigger a sufficient number of PMTs to be captured by the electronic system are assigned an EventID. This intricate labeling system allows for a clear differentiation between correlated IBD events, which represent actual IBD occurrences, and uncorrelated IBD events.

The second dataset focuses primarily on radioactivity events. Similar to the IBD dataset, it encompasses a large number of simulated events, each reflecting the complex reality of real-world physics phenomena. Additionally, the inherent electronic noise prevalent in actual physical environments is accurately accounted for, ensuring a realistic representation within the simulated context.

In this research undertaking, my central task will focus on a detailed examination and evaluation of the provided datasets. My work will primarily involve not just interpreting the inherent characteristics and peculiarities of the recorded events, but also harnessing these insights to construct comprehensive feature tables. These feature tables, generated from the datasets, will serve as the basis for my subsequent analysis and interpretation, a process which will be elaborated upon in the following sections of this study. The aim is to provide a meaningful understanding of the correlations and implications of these events within the broader context of the JUNO experiment. Per affrontare il problema, si parallelizza la simulazione su una infrastruttura chiamata DCI (Distributed Computing Infrastructure). In questo modo si può ad esempio dividere i 1500000 eventi in 1500 jobs (simulati quindi da 1500 CPU diverse) da 1000 event ciascuno, completando la produzione in poche ore invece che in mesi. Questo approccio ha però il drawback che ogni simulazione parallela sarà indipendente dalle altre e quindi per ciascuna di queste il tempo, i SimID e tutte le altre quantità partirano da 0""

Ciao Fabio, la radioattività con cui hai lavorato rappresenta tutto il contributo da decadimenti radioattivi (alfa, beta, gamma) interno ed esterno a detector, ma che hanno depositato energia all'interno del detector. Ti lascio la tabella degli isotopi e delle rate di decadimento generate qui di seguito:

| **Dataset Name** | **Number of Events** | **Rates (used in elecsim)** | |:————————-: |:————————————: |:————————————: | | U238@LS | 1,000,000 events | 3.234 Hz | | Th232@LS | 1,000,000 events | 0.733 Hz | | K40@LS | 1,000,000 events | 0.53 Hz | | Pb210@LS | 1,000,000 events | 17.04 Hz | | C14@LS | 1,000,000,000 events | 3.3e4 Hz | | Kr85@LS | 1,000,000 events | 1.163 Hz | | U238@Acrylic | 10,000,000 events | 98.41 Hz | | Th232@Acrylic | 10,000,000 events | 22.29 Hz | | K40@Acrylic | 10,000,000 events | 161.25 Hz | | U238@node/bar | 100,000,000 events | 2102.36 Hz | | Th232@node/bar | 100,000,000 events | 1428.57 Hz | | K40@node/bar | 100,000,000 events | 344.5 Hz | | Co60@node/bar | 100,000,000 events | 97.5 Hz | | U238@PMTGlass | 1,000,000,000 events | 4.90e6 Hz | | Th232@PMTGlass | 1,000,000,000 events | 8.64e5 Hz | | K40@PMTGlass | 1,000,000,000 events | 4.44e5 Hz | | Tl208@PMTGlass | 1,000,000,000 events | 1.39e5 Hz | | Co60@Truss | 0 | ? Hz | | Tl208@Truss | 0 | ? Hz | | Rn222@WaterRadon | 100,000,000 events | 90 Hz |

3.2 Feature creation

The development of machine learning models for the detection of Inverse Beta Decay (IBD) events necessitates a systematic and efficient approach to feature engineering. This process begins with the loading of two separate datasets, one for IBDs and one for radioactivity background, each containing a multitude of potential IBD events. The primary objective is to construct a feature table that encapsulates the unique characteristics of these events, providing a robust foundation for subsequent model training.

3.2.1 IBD dataset

As we mentioned earlier, an IBD event is characterized by two distinct signals with different energies, positions, and times. The first, known as the prompt signal, is caused by the annihilation of a positron with an electron in the scintillator liquid. This interaction yields a signal with a characteristic energy. The second, the delayed signal, results from the capture of a neutron by the scintillator liquid. This signal occurs with a significant delay, at a different position, and with a different energy compared to the prompt signal.

To create the feature table, all possible pairs of events within the dataset were considered, without repetition. Each possible combination was ordered temporally, meaning the second event followed the first. This temporal ordering is crucial in feature determination. Given a pair $i - j$, and considering that neutron capture occurs temporally subsequent to electron-positron annihilation, the following features were constructed:

- R_{prompt}: This feature represents the distance of the prompt signal, calculated as the distance from the origin to the point (x_i, y_i, z_i) in the detector space where the prompt signal occurred.
- $R_{delayed}$: Similar to R_{prompt}, this feature represents the distance of the delayed signal, calculated as the distance from the origin to the point (x_j, y_j, z_j) in the detector space where the delayed signal occurred.
- E_{prompt}: This feature represents the energy of the prompt signal. It captures the characteristic energy released during the annihilation of a positron with an electron in the scintillator liquid.

- $E_{delayed}$: This feature represents the energy of the delayed signal. It captures the energy released when a neutron is captured by the scintillator liquid. This capture can occur by hydrogen, resulting in a gamma ray with an energy of approximately 2.2 MeV, or by carbon, resulting in gamma rays with combined energies of about 4.95 MeV to 5.12 MeV.

- Δ_{Time}: This feature represents the time difference between the two events. It captures the temporal delay between the occurrence of the prompt and delayed signals.

- Δ_{Radius}: This feature represents the spatial distance between the two events. It captures the spatial separation between the points in the detector space where the prompt and delayed signals occurred.

These features encapsulate the temporal and spatial differences between the prompt and delayed signals, as well as their respective energies, providing a comprehensive representation of the unique characteristics of IBD events.

Event Labeling

In the context of supervised learning, the process of labeling is crucial as it provides the ground truth against which the performance of the machine learning model is evaluated. In this scenario, each pair of events in the dataset is assigned a label that indicates whether it represents a true Inverse Beta Decay (IBD) event or a background signal (BKG).

The label is a binary value: a label of 1 signifies a true IBD event, while a label of 0 signifies a BKG event. The assignment of these labels is not arbitrary but is guided by a specific criterion based on the simulation identifier (SimID) associated with each event pair.

The SimID is a unique identifier assigned to each simulated event pair during the generation of the dataset. If a pair of events share the same SimID, it means they were generated as part of the same simulation and thus are considered to represent a true IBD event. In this case, they are assigned a label of 1.

Conversely, if a pair of events do not share the same SimID, it means they were generated as part of different simulations. These events are not correlated and thus are considered to represent BKG events. In this case, they are assigned a label of 0.

This labeling strategy based on the SimID ensures a systematic and consistent methodology for event classification. It provides a clear and objective criterion to distinguish between true IBD events and BKG events, which is essential for the training and evaluation of the machine learning model.

Efficient Feature Calculation

Given the large size of the dataset and the computational complexity of feature calculation, a parallel computing approach was adopted to enhance efficiency. The feature calculation task was divided into multiple sub-tasks that could be executed simultaneously by different cores of a CPU. This parallelization significantly reduced the total computation time, particularly beneficial when working with large volumes of data.

To further optimize the computation, a method was implemented to only consider event pairs where the delayed event occurs within a time window of $5 * \tau$ from the prompt event. This approach is based on the fact that the time delay between the prompt and delayed events in Inverse Beta Decay (IBD) typically follows an exponential distribution, a characteristic of

radioactive decay processes. While this method significantly reduces the number of potential event pairs, it might exclude about 0.7% of IBD events that occur outside this time window.

While this percentage is relatively small, it's important to consider the potential impact on the analysis results.

3.2.2 Radioactivity dataset

For the radioactivity dataset, the feature calculation was performed in a manner analogous to the IBD dataset. The key difference is that event pairs from the radioactivity dataset are labeled as BKG events, hence assigned a label of 0.

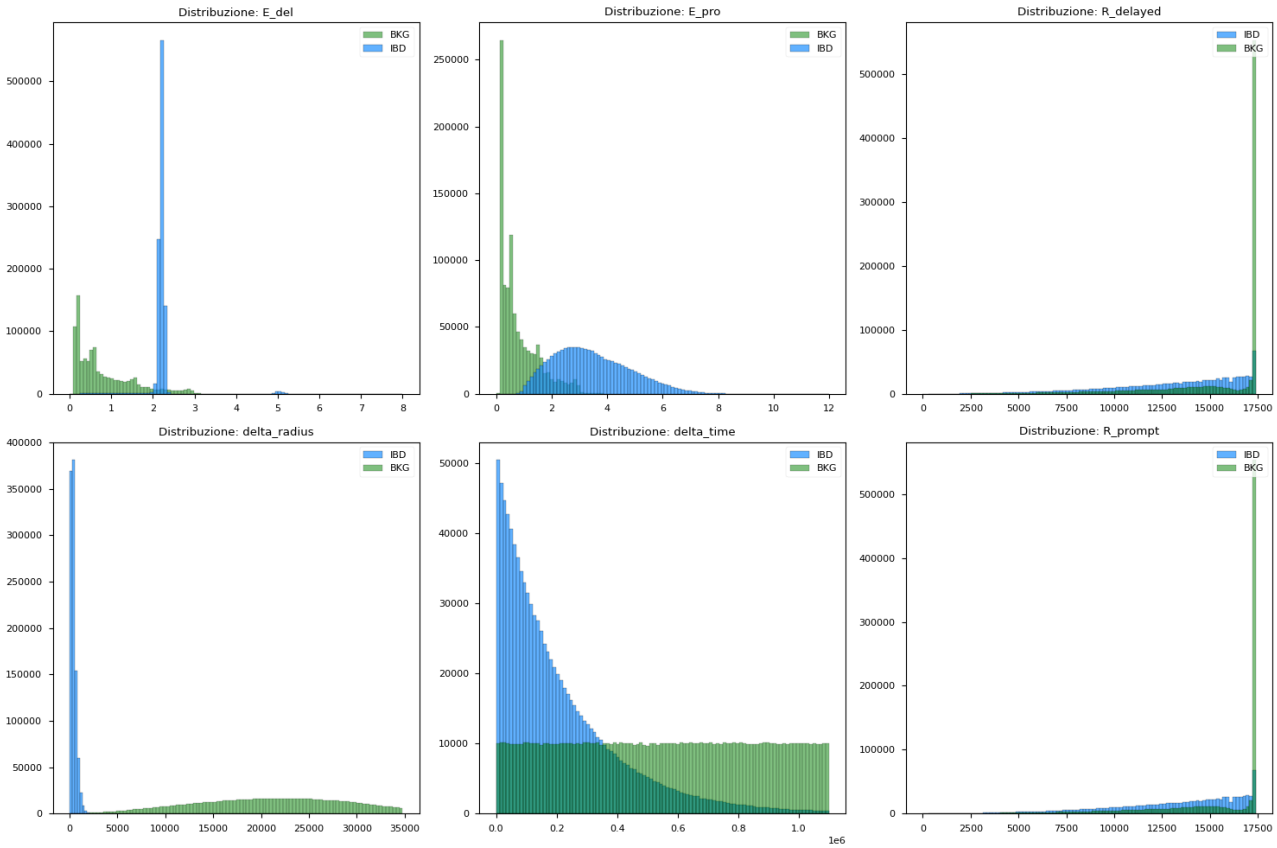

Figure 3.1: Features histograms

In summary, the feature engineering process for IBD event detection involves careful consideration of the unique characteristics of these events, systematic feature construction, and efficient computation strategies. This process provides a robust foundation for the development and training of machine learning models for IBD event detection.

3.3 Models

3.3.1 Manual Cut

3.3.2 XGBoost

3.3.3 PyThorch

3.4 Conclusion

	Manual Cut Algorithm	**BDT Algorithm**
Radioactivity	Efficiency: 99.9973% Purity: 100%	Efficiency: 99.997684% Purity: 100%
True IBDs	Efficiency: 97.734% Purity:100%	Efficiency: 99.997616% Purity: 100%

References

[Kaj16] Takaaki Kajita. “Nobel Lecture: Discovery of atmospheric neutrino oscillations”. In: *Reviews of Modern Physics* 88.3 (July 2016). DOI: 10.1103/revmodphys.88.030501.