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

Contents **ii**

1 Introduction **1**

1.1 Neutrinos Oscillation 1

1.2 The JUNO detector 3

1.3 JUNO signals and backgrounds 3

2 Frameworks **7**

2.1 Introduction to Machine Learning 7

2.2 Binary Classification 9

2.3 Decision Tree 9

2.4 Neural Networks 9

3 Analysis **11**

3.1 Datasets 11

3.2 Feature creation 12

3.3 Models 14

3.4 Results 14

References **15**

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 $P\left(\bar{\nu}_e \to \bar{\nu}_e\right)$ of electron antineutrinos.

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

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

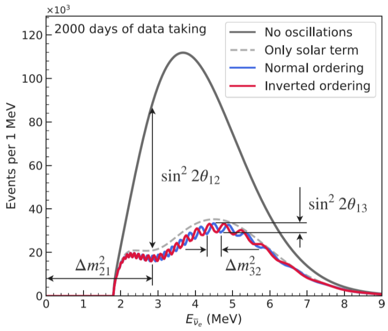


Figure 1.1: JUNO's reactor antineutrino energy spectrum is shown with and without the effect of neutrino oscillation. The gray dashed curve includes only the term in the disappearance probability modulated by $sin^2(2\theta_{12})$, while the blue and red curves use the full oscillation probability for 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.

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.

A schematic illustration of both JUNO and TAO's location is presented in Fig. 1.2.

1.3 JUNO signals and backgrounds

1.3.1 Signal

The primary ingredient of the LS is Linear Alkyl-Benzene (LAB) with a density of 0.859 g/mL. LAB is characterized by a straight alkyl chain of 10-13 carbons attached to a benzene ring,

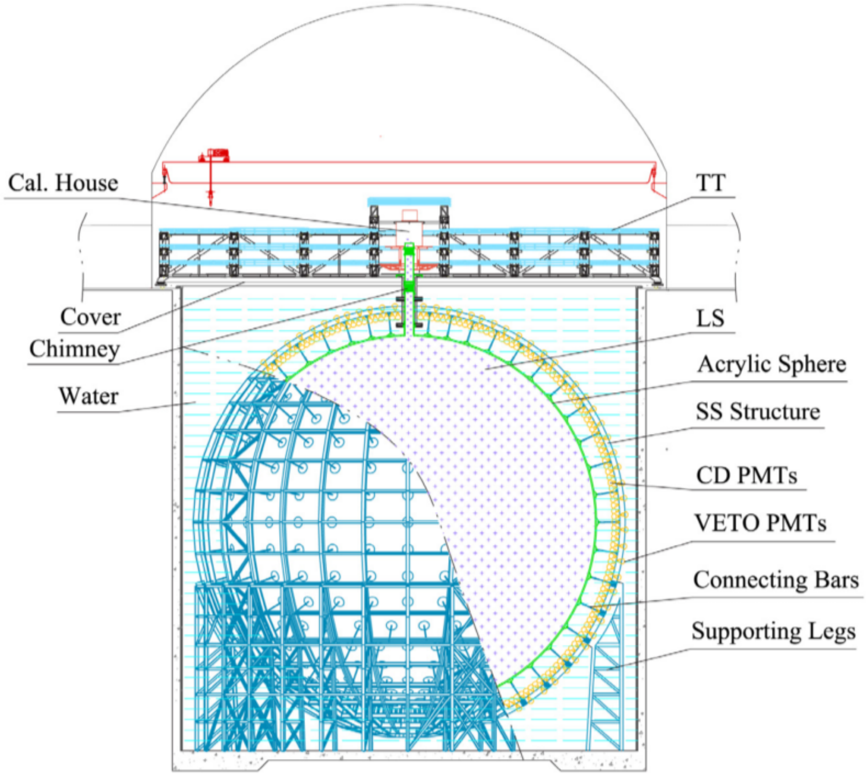


Figure 1.2: Schematic view of the JUNO experiment

known for its remarkable transparency, high flash points, robust light yield, and low chemical reactivity, all critical factors in enhancing the detector's performance. The LS also contains 3 g/L of 2,5-diphenyloxazole, serving as the fluor, and 15 mg/L of p-bis-(o-methylstyryl)-benzene, which acts as the wavelength shifter. These ingredients, in collaboration with 17612 large 20-inch photomultiplier tubes (PMTs) and 25600 smaller 3-inch PMTs installed on a spherical structure with a 19.5 m radius, amplify the scintillation light signals, significantly contributing to the detection of neutrino events.

The scintillator is doped with a small amount of gadolinium to enhance its sensitivity to antineutrinos via the inverse beta decay (IBD) process. The liquid scintillator used in JUNO is a combination of LAB (linear alkyl benzene) and PPO (2,5-diphenyloxazole) doped with a small amount of bis-MSB (1,4-bis(2-methylstyryl) benzene). When a neutrino interacts with the scintillator, it can produce charged particles such as electrons, protons, and alpha particles that travel through the scintillator and excite the scintillation molecules. This excitation results in the emission of photons with a wavelength of around 430 nm. These photons are detected by 20,000 20-inch photomultiplier tubes (PMTs) distributed in a 3-dimensional arrangement inside the detector.

$$\begin{aligned}&\overline{\nu}e + p \rightarrow e^+ + n\\ &n +_Z^A X \rightarrow_{Z-1}^A X^* + \gamma\\ &e^+ + e^- \rightarrow 2\gamma\end{aligned} \tag{1.1}$$

=

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.

Moreover, the scintillator's composition and the detector's design are optimized to reduce background noise from other sources of radiation, such as cosmic rays and natural radioactivity. By carefully controlling these backgrounds, JUNO aims to achieve a signal-to-background ratio of better than 1:10,000, which is essential for observing the subtle effects of neutrino oscillation.

In JUNO's location, the energy spectrum will be distorted by two types of oscillations. The first is a slow (low frequency) oscillation driven by Δm^2_{21} and modulated by $\sin^2\theta_{12}$, while the second is a fast (high frequency) oscillation driven by Δm^2_{31} and modulated by $\sin^2\theta_{13}$. Fitting the data spectrum against the predicted spectrum distorted by standard neutrino oscillations enables measuring the oscillation parameters.

1.3.2 Background

Several different types of backgrounds signal are produced in the detector. For analysis we deeply analysed only the three most important contributes:

Radiogenic Backgrounds Radiogenic backgrounds arise from decays of radioactive isotopes in detector materials and surrounding rock. These decays can produce various forms of radiation, such as gamma rays and neutrons, which can interact with the detector and produce background events. Examples of radiogenic isotopes include ^{238}U, ^{232}Th, ^{40}K, and their daughter products. The main contributions to the radiogenic backgrounds come from the ^{238}U and ^{232}Th decay chains, with a smaller contribution from ^{40}K.

Cosmogenic Backgrounds Cosmogenic backgrounds arise from interactions of cosmic rays with materials surrounding the detector, such as the atmosphere and the Earth's crust. Muons produced in these interactions can penetrate the detector and produce background events. Specifically they interact with detector materials, producing isotopes such as ^{11}C, ^{9}Li, and ^{8}He, instable atoms which decay and produce additional background events.

Atmospheric Neutrino Backgrounds Atmospheric neutrino backgrounds arise from interactions of cosmic ray protons and nuclei with the Earth's atmosphere, which produce a flux of neutrinos that can interact with the detector. These interactions can produce both charged and neutral current events, which can mimic the signal from reactor neutrinos.

Reactor Antineutrino Backgrounds Reactor antineutrino backgrounds arise from the neutrinos produced in the nuclear reactors that power the JUNO experiment. These antineutrinos are the main signal for JUNO, but a small fraction of them can interact with the detector in ways that mimic background events. These interactions can produce both charged and neutral current events, which can be difficult to distinguish from the signal.

Here a viasualization sumary of all the bacgrounds contributions:

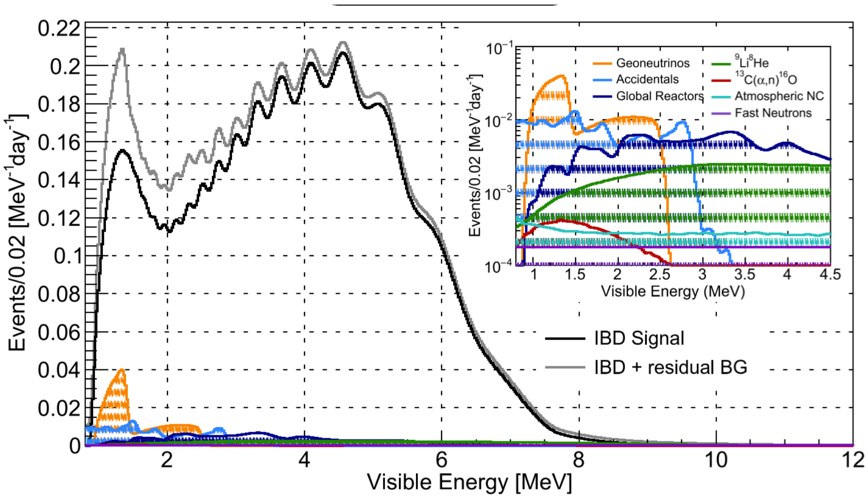


Figure 1.3: Visible energy spectrum as measured by the LPMT system with (grey) and without (black) backgrounds is that which is anticipated for JUNO. The energy resolution is one of the assumptions listed in the text. The predicted backgrounds, which make up around 7% of the whole sample of IBD candidates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

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

2.3 Decision Tree

Decision trees are a cornerstone of machine learning algorithms, providing a robust model that segments the feature space into various non-overlapping regions. The model is capable of conducting both regression and classification tasks, creating rules from the available features to predict the value of a target variable.

Mathematically, we can represent the decision tree model as follows. Given a dataset D containing n instances, where each instance i is an input-output pair (x_i, y_i) with x_i belonging to the input space X and y_i to the output space Y. The decision tree maps an instance x_i to an output y_i through a series of binary tests:

$$y_i = f(x_i) = \sum_{j=1}^{J} c_j I(x_i \in R_j) \tag{2.4}$$

where $f(x_i)$ is the decision tree, R_j are the disjoint regions of the feature space, $I()$ is the indicator function, and c_j is the predicted value in region j.

To grow a decision tree, we start at the root and recursively split the data based on the feature that maximizes the reduction of a chosen impurity measure. Common measures include entropy and the Gini index, calculated as follows:

Entropy: $Entropy(S) = -\sum_{i=1}^{c} p_i \log_2(p_i)$

Gini Impurity: $Gini(S) = 1 - \sum_{i=1}^{c} (p_i)^2$

2.3.1 Boosted Decision Trees

Boosting is a meta-algorithm in machine learning, developed to convert weak learners into a strong predictive model. Boosted decision trees are an implementation of this concept, often leading to significantly improved model performance.

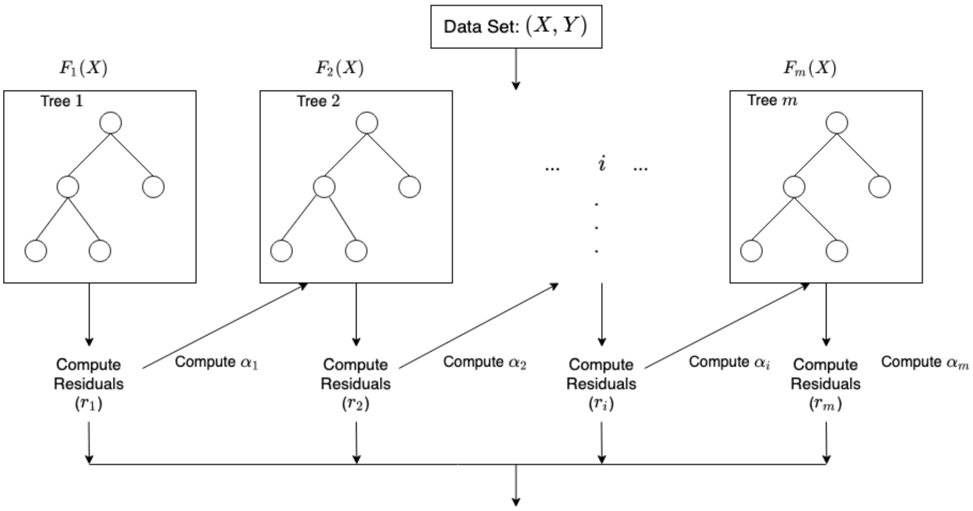
Boosted decision trees work on the principle of fitting the boosting model $F(x)$ by minimizing the loss function $L(y, F(x))$ over the training data. This is typically done in a stage-wise fashion. Given the current model $F_m(x)$, we fit a weak learner (a small decision tree, $h(x)$) to the negative gradient of the loss function, evaluated with the current model and updated as:

$F_{m+1}(x) = F_m(x) + \alpha h(x)$

where α is a constant, often set via line search to minimize the loss function.

Gradient Boosting and AdaBoost are two common methods. XGBoost, or eXtreme Gradient Boosting, stands as a notable Gradient Boosting variant. It introduces regularization parameters to prevent overfitting, handles missing values, and utilizes both parallelized and distributed computing, making it suitable for large-scale problems.

2.4 Neural Networks



$$F_m(X) \;=\; F_{m-1}(X) + \alpha_m h_m(X, r_{m-1}),$$

where α_i, and r_i are the regularization parameters and residuals computed with the i^{th} tree respectfully, and h_i is a function that is trained to predict residuals, r_i using X for the i^{th} tree. To compute α_i we use the residuals computed, r_i and compute the following: $arg \min\limits_{\alpha} \;=\; \sum_{i=1}^{m} L(Y_i, F_{i-1}(X_i) + \alpha h_i(X_i, r_{i-1}))$ where $L(Y, F(X))$ is a differentiable loss function.

Figure 2.1

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""

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.

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.

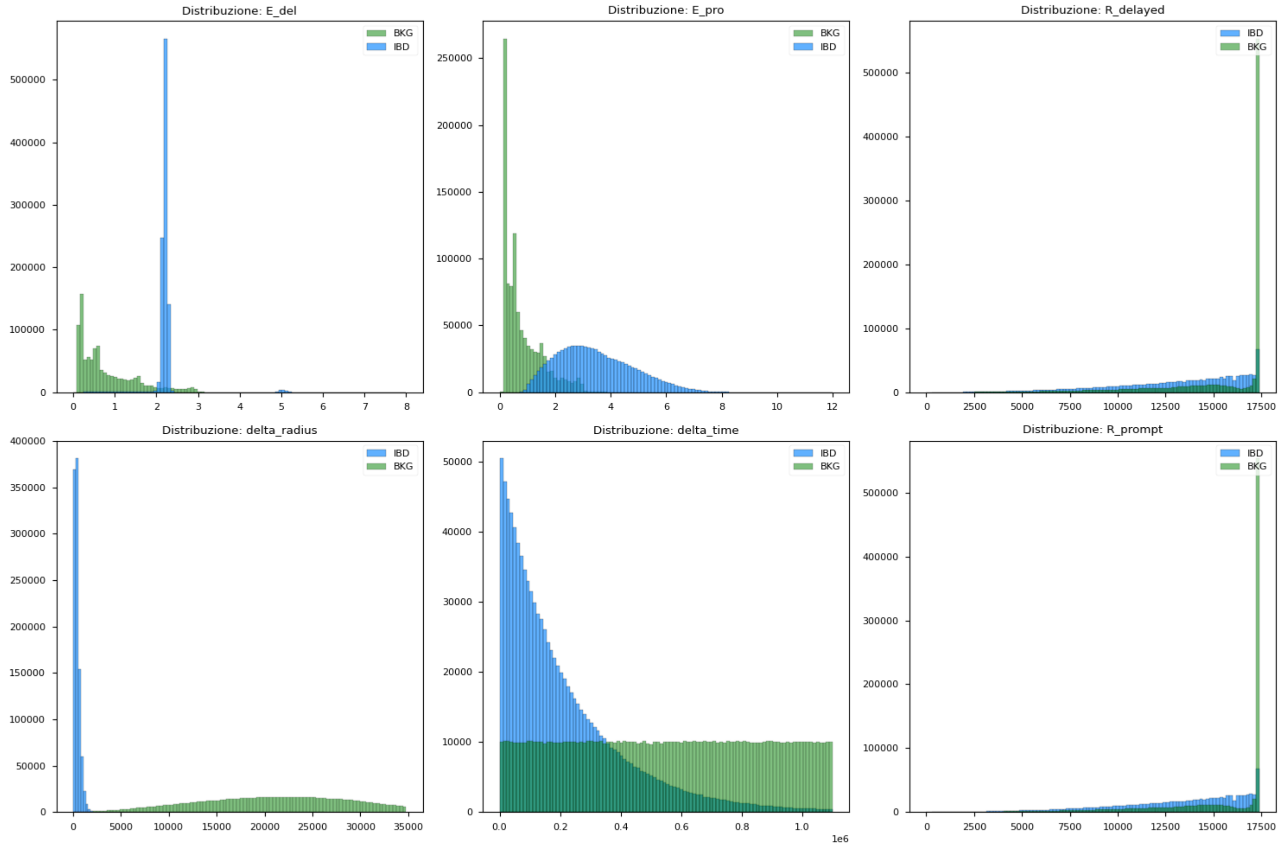


Figure 3.1: Features histograms

3.3 Models

3.3.1 Manual Cut

3.3.2 XGBoost

3.3.3 PyThorch

3.4 Results

	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`.