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

Correlatore

dott. Andrea Serafini

Laureando

Fabio Cufino

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, 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:

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

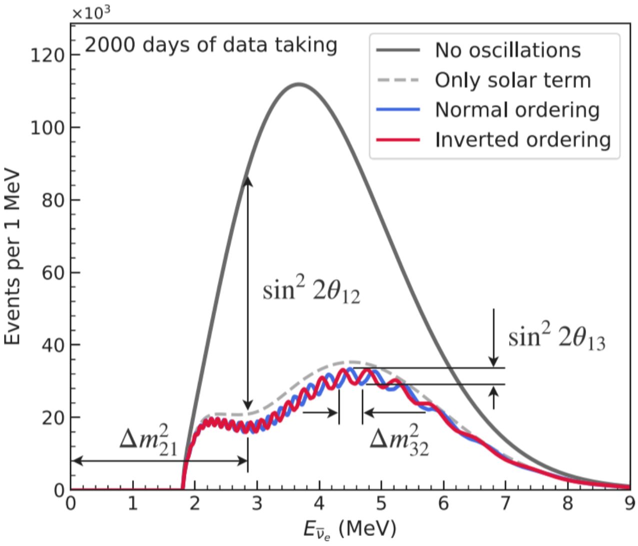
where $s_{ij} \equiv \sin\theta_{ij}, c_{ij} \equiv \cos\theta_{ij}$.

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 $m_1 < m_2 < m_3$,
- *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.

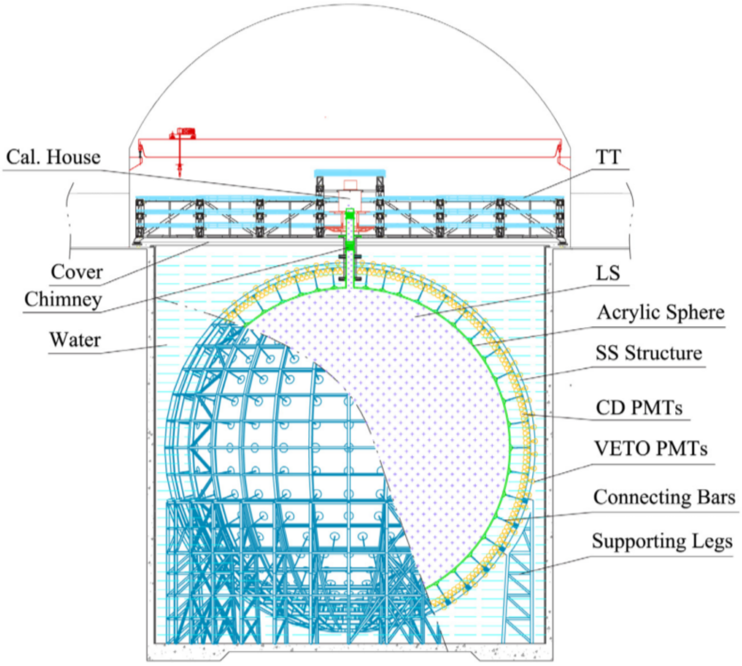


Figure 1.2: Schematic view of the JUNO experiment

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 signal and background

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 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:

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

IBD is characterized by a comparatively low threshold of 1.8 MeV, a substantial cross section, and it can be readily differentiated from the background due to its delayed γ signature.

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: $e^+ + e^- \to 2\gamma$. 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. The process is described as follows:

$$n +^1 \mathrm{H} \to^2 \mathrm{H}^* \to^2 \mathrm{H} + \gamma \tag{1.2}$$

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), sensitive detectors that convert light into an electrical signal. They operate based on the photoelectric effect and subsequent electron multiplication. 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 Background

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 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 ($\mathcal{O}$(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 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.

Additional Sources 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:

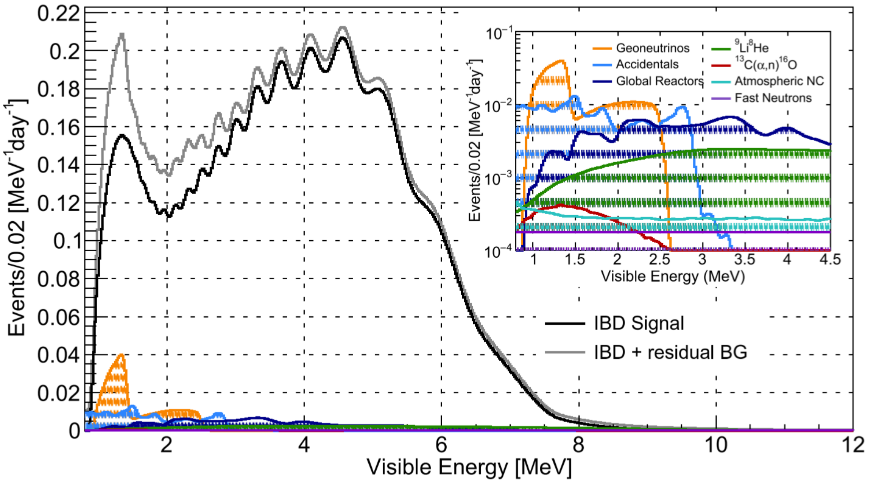


Figure 1.3: Visible energy spectrum with (grey) and without (black) backgrounds is that which is anticipated for JUNO. The predicted backgrounds, which make up around 7% of the whole sample of IBD candidates and are primarily confined below, are shown in the inset as spectra. $\approx$ 3 MeV

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.

2.1.1 Supervised Learning

Supervised learning, a cornerstone of machine learning, operates on the premise of training an algorithm using a labelled dataset. This training ensures that each input aligns with a correct output, with the overarching aim to develop a function that can link input data to output data accurately. A primary task stemming from supervised learning is binary classification, which is responsible for classifying elements of a dataset into one of two possible categories based on inherent features. In the context of this thesis, the emphasis is on applying the binary classification task to the JUNO experiment. The task is to distinguish whether a specific event is an inverse beta decay event or a background event.

Two distinct machine learning algorithms, **Gradient Boosting Decision Trees** and **Neural Networks**, are deployed to achieve this.

2.2 Decision Tree

A Decision Tree is a supervised machine learning algorithm used for classification and regression. It segments a dataset into subsets by applying decision rules inferred from the data's features. Each internal node represents a rule, dividing the data into two groups. The choice of rule is based on Information Gain, which relies on Entropy.

Entropy H quantifies the impurity within a set S and is defined as:

$$H(S) = -p_+ \log_2(p_+) - p_- \log_2(p_-) \tag{2.1}$$

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:

$$IG(S,A) = H(S) - \sum_{v \in V(A)} \left(\frac{|S_v|}{|S|} \right) H(S_v) \tag{2.2}$$

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.

2.2.1 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

$$r_{im} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)} \tag{2.3}$$

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

$$F_m(x) = F_{m-1}(x) + \nu \cdot h_m(x) \tag{2.4}$$

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.

An advanced and highly efficient implementation of this method is XGBoost, which introduces several improvements such as parallel processing.

2.3 Neural Networks

Neural Networks (NNs) 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:

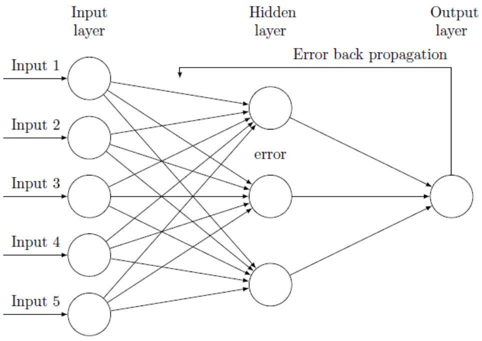
$$z = \sum_{i=1}^{n} w_i x_i + b$$

The calculated value, z, is then passed through an *activation function*, f, to generate the neuron's output, $a = f(z)$.

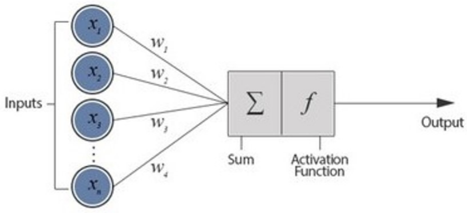
The activation function introduces non-linearity into the model, which is crucial for the network's ability to learn complex patterns. Common choices for f include the sigmoid, hyperbolic tangent, and ReLU (Rectified Linear Unit) functions.

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.

The following is a graphical representation of an ANN and a single neuron:



(a) Graphic representation of ANN



(b) Single Neuron representation

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

This study employs two distinct datasets to construct a feature table for use in classification algorithms. Both datasets are the result of Monte Carlo simulations conducted via the SNiPER software.

IBD Dataset

The primary dataset, known as the IBD dataset, is designed specifically for analyzing potential Inverse Beta Decay events. Notably, it excludes any contributions from background events.

The IBD dataset comprises the following key features:

- `SimID`: This serves as a unique identifier for each true IBD pair. It should be noted that a prompt-delayed pair originating from an IBD event will share the same SimID.
- `(x, y, z)`: These represent the reconstructed coordinates of the point within the detector where the IBD event occurred.
- `E`: This represents the energy of the individual event as recorded by the detector.
- `t`: This represents the timestamp of when the event occurred.

For the purposes of this analysis, we employed a dataset that simulates data collection over a period of 99999999999 days.

In this dataset, a total of 2,977,856 events have been recorded.

BKG Dataset

The second dataset primarily focuses on radioactivity events. Contrary to the IBD dataset, the BKG dataset does not necessitate the use of SimID. However, it retains similar characteristics to the IBD dataset, with each event defined by its energy `E`, the timestamp of occurrence `t`, and its Cartesian coordinate position `(x, y, z)`.

It includes events generated from various isotopes and decay rates, as shown in the provided Table 3.1:

The table provides a comprehensive view of the events generated from various isotopes and their decay rates used in the BKG dataset. This radioactivity represents all the contributions

Dataset Name	Number of Events	Rates
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

Table 3.1: Table of isotopes and decay rates used in the BKG dataset

from radioactive decays (α, β, γ) both internal and external to the detector, but that have deposited energy within the detector.

It is important to underline the differences in the location of the isotopes in the detector:

- **Liquid Scintillator (LS):** The central part of the detector, where isotopes U238, Th232, K40, Pb210, C14, and Kr85 are found. It's worth noting that C14 contributes a significant number of events, i.e., 1,000,000,000 events with a high decay rate of 3.3e4 Hz.

- **Acrylic:** This constitutes the detector walls, hosting isotopes U238, Th232, and K40, each contributing to 10,000,000 events.

- **Node/bar:** This is the metallic structure supporting the detector. Here, isotopes U238, Th232, K40, and Co60 are located, contributing 100,000,000 events each.

- **PMT Glass:** The glass of the photomultipliers containing isotopes U238, Th232, K40, and Tl208. The isotope U238 stands out with 1,000,000,000 events and an extremely high decay rate of 4.90e6 Hz.

- **WaterRadon:** This represents the gas present in the water surrounding the detector and inside the detector during the initial filling stages. The isotope Rn222 contributes 100,000,000 events.

However, not all generated - and listed in the table - events interact with the detector. Some events are so low in energy that they aren't sufficient to illuminate the photomultiplier tubes (PMTs) and, therefore, to trigger the acquisition of the event. Despite these caveats, the table offers a valuable understanding of the decay contributions within the detector's different components.

It is important to note that U238@PMTGlass has one of the highest decay rates, at approximately 4.90 million Hz. This high rate indicates that Uranium-238 within the PMT Glass is highly active and undergoes decay at a very rapid pace. Such high activity could be significant in the context of the study, as it will contribute substantially to the background noise or signals within the detector.

Similarly, Th232@PMTGlass also exhibits a high decay rate, around 864,000 Hz. Like Uranium-238, Thorium-232 in the PMT Glass is highly active.

In the dataset provided, a total 8,841,188 events has been recorded.

3.2 Feature creation

The development of models for the detection of IBD events necessitates a systematic and efficient approach to feature engineering. This process begins with the loading of the two separate datasets discussed above, one for IBDs and one for radioactivity background.

3.2.1 IBD Features Table

As we mentioned earlier, an IBD event is characterized by two distinct signals with different energies, positions, and times.

To create the feature table, all possible pairs of events within the dataset were considered, without repetition. The ascending temporal order in which the features are created is crucial in feature determination, considering that neutron capture occurs temporally subsequent to electron-positron annihilation. Given a pair $i - j$ in the IBD dataset, the following features were constructed:

- $\mathbf{R_{prompt}}$: 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.
- $\mathbf{R_{delayed}}$: 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.
- $\mathbf{E_{prompt}}$: 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.
- $\mathbf{E_{delayed}}$: 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.
- $\mathbf{\Delta t}$: This feature represents the time difference between the two events. It captures the temporal delay between the occurrence of the prompt and delayed signals.
- $\mathbf{\Delta R}$: 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

The labeling system is based on the SimID. A label of 1 signifies a true IBD event generated as part of the same simulation, while a label of 0 signifies uncorrelated BKG events that emerged from different simulations.

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 calculation is done on a virtual machine hosted on Cloud Veneto, with 14 cores of CPU. The feature calculation task was divided into multiple sub-tasks that could be executed simultaneously by the different cores of the CPU. This parallelization significantly reduced the total computation time.

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 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 Features Table

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.

The distribution of the features are presented in the Graph 3.1.

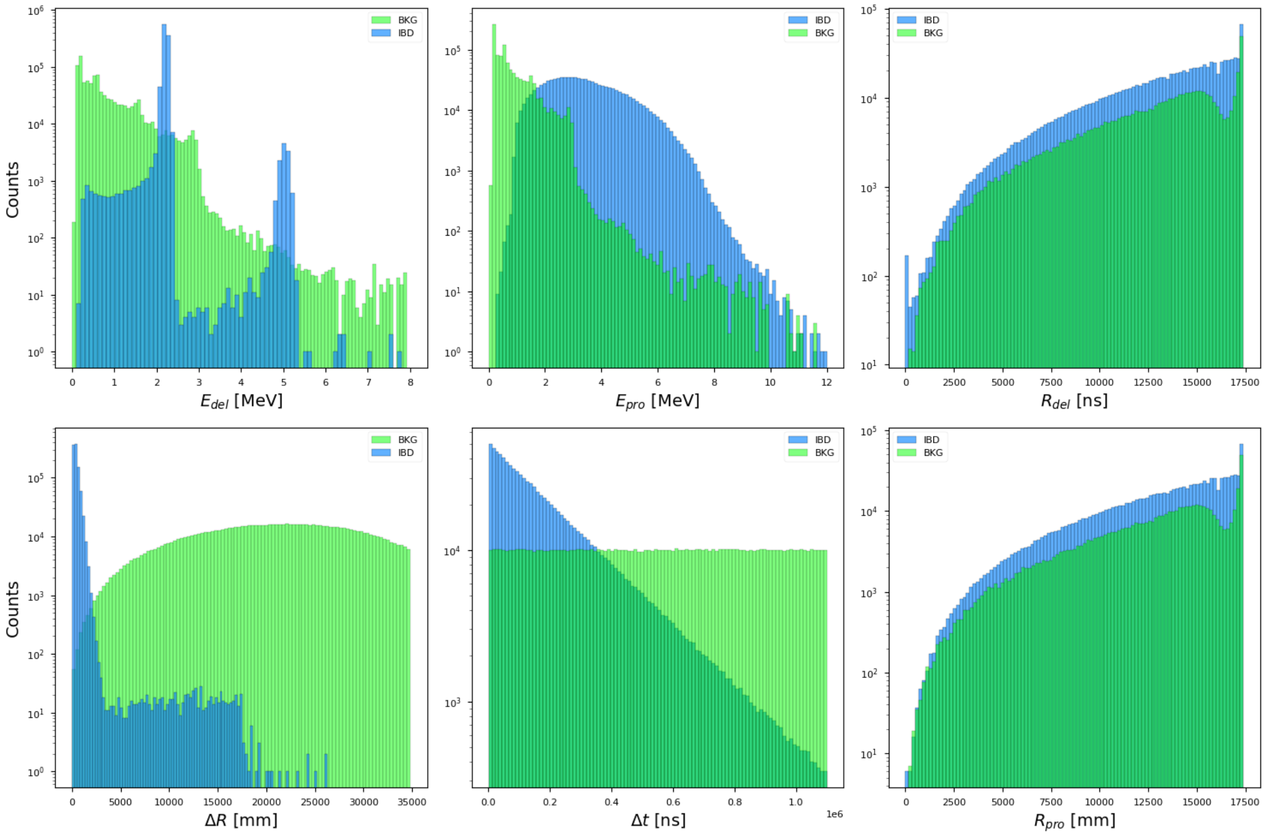


Figure 3.1: Features histograms

Feature distribution

In the presented graph, focusing only on IBD events, it is clearly observable for E_{del}, the distinct characteristics of the IBD events,suchthe peaks at 2.2 MeV and 5.9 MeV, and the clearly visible positron spectrum in the E_{pro}. The distribution of R_{del} and R_{prompt} is reasonable, considering the detector's spherical shape, making it more likely for an IBD event to occur in the outer part of the detector due to the greater volume of liquid scintillator. Additionally, the plot for ΔR clearly shows that IBD events are generally very close to each other, with a peak observed for $\Delta R < 3000$ mm. The Δt distribution for IBD events follows a decreasing exponential distribution, which appears as a straight line since the plot is logarithmic on the y-axis.

For the BKG events, on the other hand, we can see that they have a fundamentally different distribution for ΔR compared to IBD events. There is a higher occurrence of events with $\Delta R > 3000$ mm, and even at $\Delta R \approx 35000$ mm, which corresponds to events occurring in opposite parts of the detector, are significantly probable. This starkly distinguishes IBD events from BKG events. For the Δt distribution of BKG events, it is observed to be nearly constant across all possible Δt values. The number of E_{del} events decreases as E_{del} increases, and no well-defined peaks are observed as in the case of IBD events. R_{del} and R_{prompt} for BKG events show a distribution similar to IBD events, with the distinction that there are more counts in the final part of the detector, where there is a higher presence of BKG events. This is due to, as evident from the Table 3.1, the acrylic, steel bars, PMT glass, and the presence of radon in the water.

Upon inspection, a noticeable decline around the 16500mm range is evident in the distribution of both R_{del} and R_{pro}. This can be largely attributed to C14, a low-energy decay isotope. Its presence leads to more detections at the detector's center than the edge due to its relatively low energy release, which barely meets the threshold for activating the necessary PMTs.

As C14 events are closer to the detector's edge, fewer PMTs are triggered due to the increased distance the light has to travel. This results in fewer detected events near the boundary.

3.3 Models

In the JUNO experiment, a crucial aspect is developing and optimizing an event selection algorithm. This chapter introduces several algorithms, starting with a manual cut-based approach, **Manual Cut**, which sets criteria based on event physics and background noise. Additionally, machine learning-based algorithms, specifically **Boosted Decision Trees** and **Neural Networks**, are discussed. We aim to offer an in-depth comparison of these methods, emphasizing their advantages and limitations within the JUNO experiment.

3.3.1 Manual Cut

The algorithm is designed to suppress various types of background noise while maintaining high efficiency for true IBD events. The selection criteria, or "**cuts**" are implemented using Python, and are applied to the Features Tables discussed above. Each cut within the algorithm serves a distinct purpose in the overall event selection process. It is essential to emphasize that the selection criteria for the cuts have been diligently employed based on thorough research as outlined in the referenced paper

The key components of the event selection algorithm are as follows:

1. **Delta Time (Δt) and Delta Radius (ΔR) cuts**: The first cut is applied on the time delay and the radial distance between the prompt and delayed signals. The criteria are:

   - Time separation between the prompt and delayed signals should be less than 1.0 ms.
   - Spatial 3D separation should be less than 1.5 m.

   The cut values for Delta Time (Δt) and Delta Radius (ΔR) are empirically set based on Inverse Beta Decay events. The 1.0 ms time cut represents the usual neutron thermalization and capture timeframe, while the 1.5 m spatial cut considers the short distance neutrons typically travel before capture. These values aim to isolate genuine IBD events by optimizing the signal-to-noise ratio through simulations and detector response analysis.

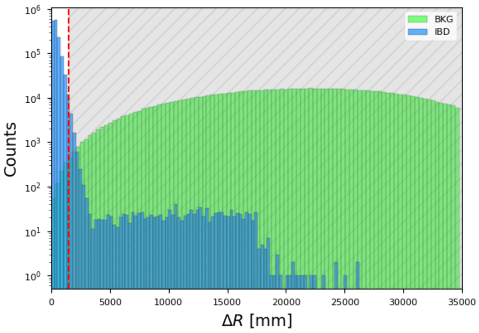


Figure 3.2: ΔR cut

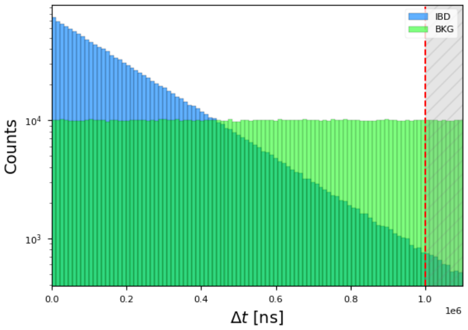


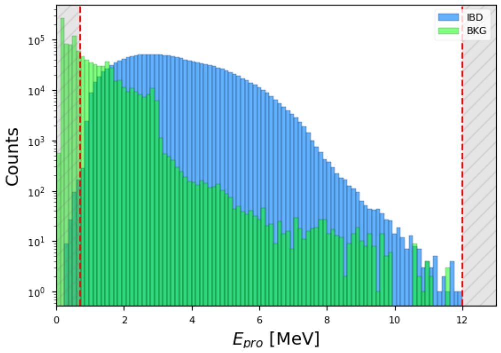
Figure 3.3: Δt cut

2. **Energy of the Prompt Signal (E_{pro}) Cut**: The next cut is applied on the energy of the prompt signal, which is the initial signal produced by the antineutrino interaction. The criteria are:

   - Energy of the prompt signal should be within the [0.7, 12.0] MeV range.

   This cut is grounded in the anticipation that the IBD events predominantly occupy this energy range. The energy of the prompt signal is associated with the energy of the positron emanating from the IBD reaction. The selection of this particular range is strategic, aiming to optimize the signal-to-background ratio by focusing on the energy window where IBD events are most likely to occur and where the detector has optimal sensitivity and resolution.

Figure 3.4: E_{pro} cut

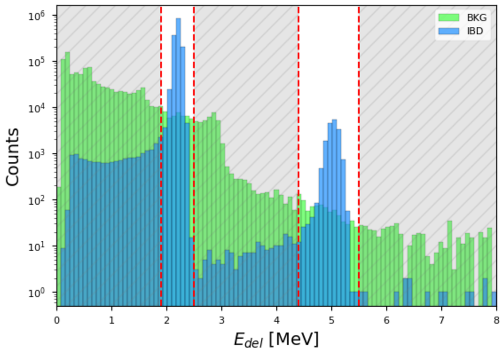


3. **Energy of the Delayed Signal (E_{del}) Cut**: The final cut is applied on the energy of the delayed signal, which is the signal produced by the neutron capture that follows the antineutrino interaction. The criteria are:

- Energy of the delayed signal should be within the [1.9, 2.5] MeV or [4.4, 5.5] MeV ranges.

These energy selection windows are aligned with the energies characteristic of neutron capture on hydrogen and carbon atoms. The energy of the delayed signal is a hallmark of the neutron capture process and varies based on the capturing element. The chosen ranges are deliberately selected to coincide with the expected energy signatures for neutron capture on hydrogen and carbon within the detector, which is essential for accurately isolating and analyzing the events of interest.

Figure 3.5: E_{del} cut



Results

The evaluation showcased the algorithm's adeptness in pinpointing true IBD events and distinguishing them from background noise. The findings, encompassing the accuracy for true IBD events and efficiency for background events, are organized in two tables. The confusion matrix (Table 3.3) offers a comprehensive classification breakdown, while the summary table (Table 3.2) succinctly highlights accuracy rates. The notable efficiency and scarce misclassification of background events underscore the algorithm's prowess in curbing false positives. It is important to highlight that the evaluation was conducted on a test dataset, which is 20% of the original size of the feature table.

	Manual Cut
IBD Efficiency	97.702%
BKG Efficiency	99.997%

Table 3.2: Performance

	Predicted BKG	**Predicted IBD**
Actual BKG	200640	7
Actual IBD	4542	194844

Table 3.3: Confusion Matrix

3.3.2 XGBoost

XGBoost is an optimized gradient-boosting decision tree algorithm, known for its speed and performance, achieved through parallel processing. It's well-suited for complex patterns, making it ideal for the JUNO experiment's event selection. The XGBoost model was fine-tuned with specific hyperparameters:

- **Number of parallel threads (`nthread`)** : Set to -1, utilizing the maximum available threads for faster training.

- **Random seed (`seed`)** : Set to 1 for reproducibility, ensuring consistent random number generation.

- **Number of estimators (`n_estimators`)** : Configured with 10,000 decision trees, controlling model complexity. More trees can improve training performance but may lead to overfitting.

- **Learning rate (`learning_rate`)** : Set at 0.05, dictating each tree's contribution to the final prediction. A smaller rate makes the model more robust to overfitting.
- **Maximum tree depth (`max_depth`)** : Limited to 3, controlling the complexity of each tree. A larger depth can lead to overfitting.

The chosen hyperparameters strike a balance between computational efficiency and model performance, allowing control over the learning process and model complexity. To optimize the XGBoost model, a Grid Search technique was used. This method systematically evaluated various hyperparameter combinations to identify the optimal configuration that maximizes model accuracy.

Results

The algorithm exhibited remarkable efficiency in identifying true IBD events and distinguishing them from background events. A confusion matrix was constructed to provide a comprehensive understanding of the model's precision and effectiveness. The analysis was performed on the total number of IBD and BKG, separately.

The confusion matrix, presented in Table 3.5, reveals the number of true positives, false positives, true negatives, and false negatives. The exceptionally low number of false positives and false negatives underscores the algorithm's effectiveness in minimizing misclassifications.

Additionally, the efficiency rates for IBD and background classifications are summarized in a separate table. The high efficiency rates further emphasize the algorithm's proficiency in both identifying true IBD events and rejecting background events.

To avoid bias from the dataset used for the model evaluation, it's important to note that the evaluation was carried out on a test dataset, which constitutes 20% of the original feature table's size.

	XGBoost
IBD Efficiency	99.9985%
BKG Efficiency	99.9979%

Table 3.4: Performance

	Predicted IBD	**Predicted BKG**
Actual IBD	199811	4
Actual BKG	3	200215

Table 3.5: Confusion Matrix

Interpretation of the model

In our study, we used **SHAP** (SHapley Additive exPlanations) to interpret the predictions of a trained XGBoost model. SHAP utilizes concepts from game theory, treating predictions as a "game" where features are the "players". The SHAP value for a feature is its average contribution to every possible combination of features.

The SHAP value, ϕ_i, for feature i is calculated using:

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} [f(S \cup \{i\}) - f(S)] \tag{3.1}$$

Here, N is the set of all features, S is a subset of N excluding feature i, and $f(S)$ is the model's prediction with feature set S. The term $\frac{|S|!(|N|-|S|-1)!}{|N|!}$ assigns a weight to each subset based on the number of times it appears in all permutations of the features.

Based on the calculation of SHAP values, we can construct visualizations that aid in analyzing and understanding how the model has learned to differentiate between Inverse Beta Decay events and background events, contributing to model interpretability.

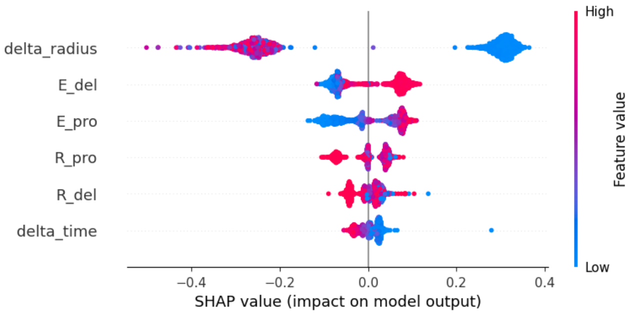


Figure 3.6: XGBoost Summary Plot

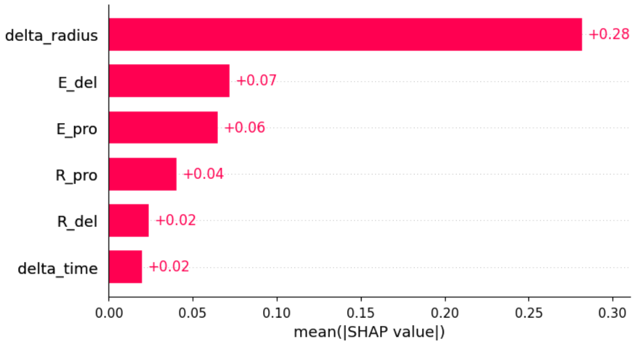


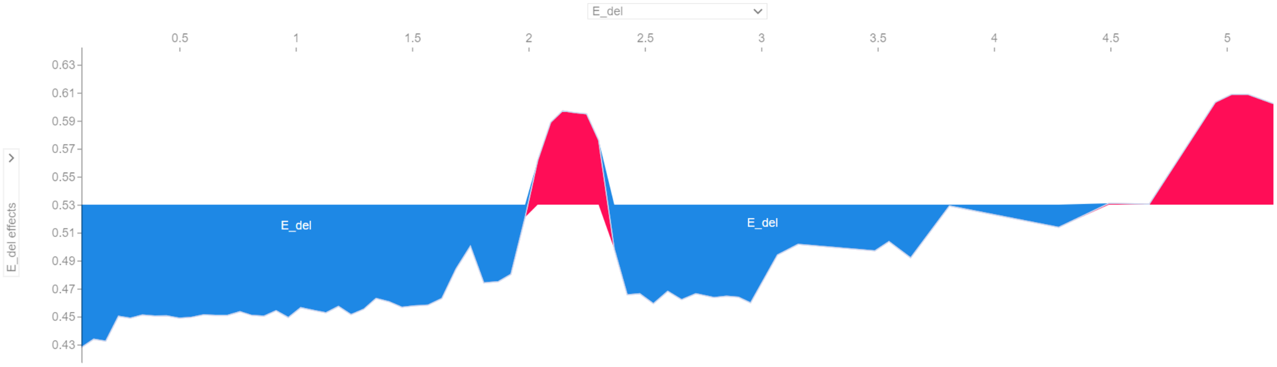
Figure 3.7: XGBoost feature importance

The presented graphs depict the importance of each feature used by the algorithm for learning, measured by calculating the mean of the SHAP values. On the right, we see a histogram where the x-axis represents the mean absolute SHAP value for each feature. The first key characteristic of the model is evident here: the feature with the most importance in classification is ΔR. Here, the prominence of the feature ΔR is underscored by the fact that it possesses the highest mean absolute SHAP value, which is 0.28. Moreover, referring back to the Graphs 3.1, it was already observable that ΔR is the feature that separates the IBD class most distinctly from the BKG class. A clear separation is evident from $2500mm$ onwards, where BKG events prevail, while IBD events are more prevalent below $2500mm$.

In the left summary plot, where the x-axis represents the SHAP value and the y-axis represents various features, two distinct data clusters for the ΔR feature are strikingly evident. For high values of ΔR, the algorithm yields a negative SHAP value, which corresponds to the expected classification as background (BKG) events. Conversely, for lower values of ΔR, the algorithm returns positive SHAP values, signifying events accurately identified as inverse beta decay. Notably, there is a clear separation between these clusters, demonstrating the algorithm's high confidence in categorizing events based on this feature.

Second in order of importance, with a SHAP value approximately four times smaller than that of ΔR, is E_{del}, the energy of the delayed event. Comparing with the feature histogram, Figure 3.1, it's clear that, focusing on the E_{del} distribution, most BKG data occupy the initial part of the histogram, thus at lower energies, and the algorithm has learned to determine that for lower delayed signal energies, the event is classified as a BKG event, based on the summary plot and the value of the SHAP value. For slightly higher energies, given the presence of characteristic peaks that significantly increase the counts of IBD events, the algorithm learns to correctly determine an IBD event.

Delving deeper into the analysis of this feature, a plot was created where the x-axis represents individual events, and the y-axis represents the effect that each event had on the E_{del} feature. It is observed that for events in the range $\approx [1.8, 2.3]MeV$ and $\approx [4.5, 5.2]MeV$, the algorithm has learned to perfectly distinguish the characteristic peaks of neutron capture compared to all background events.



Regarding the E_{pro} feature, for values in the range [0, 1] MeV, the histogram is predominantly occupied by BKG events, and as seen from the summary plot, these are correctly identified by the algorithm. However, for prompt signals with energies within the positron-like spectrum, the algorithm identifies these events as IBD. The features R_{pro}, R_{del}, and Δt do not contribute significantly to the algorithm's ability to discern between the two classes from their distribution, as there are no clear differences between the feature histograms for IBD and BKG.

Based on the aforementioned information, two distinct graphs, known as waterfall plots, are presented, Figure 3.9 and Figure 3.8. These plots visually display the individual contributions of each feature to the model's final prediction, which is 1 if it's an Inverse Beta Decay event and 0 if it's a background event. The starting point of these plots is the '*base value*', which is calculated as the average of the model's predictions across the entire training dataset. The $f(x)$ shown in the graph represents the predicted value, and is mathematically expressed as:

$$\text{Final Output} = \text{Base Value} + \sum_{i=1}^{n} \text{SHAP Val}_i \tag{3.2}$$

This equation demonstrates how the model arrives at its final prediction by combining the base value with the contributions of each feature through their SHAP values.

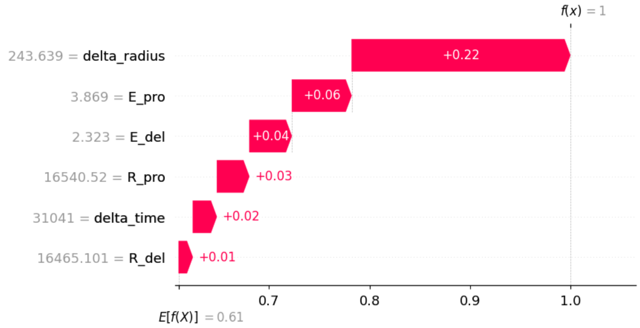


Figure 3.8: Waterfall IBD

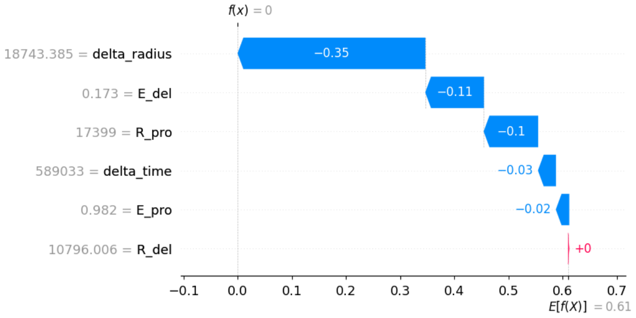


Figure 3.9: Waterfall BKG

In the SHAP waterfall plots, each feature is represented by a bar, with the length proportional to its SHAP value, indicating its contribution to the prediction. Notably, '`delta_radius`' has the longest bar, reflecting its SHAP value, for both the predictions, indicating that it is the most influential feature in this instance, as we expected. Conversely, '`delta_time`','`E_del`','`E_pro`' have the shortest bar due to the SHAP value, signifying a lesser contribution of the prediction.

This graphical representation provides an intuitive understanding of how each feature is influencing the model's prediction for this particular event, crucial for the interpretability of this models.

It is noteworthy to observe how the SHAP values exert influence on the predictions of two mislabeled events, which are, respectively, *False Positive* and *False Negative* predictions:

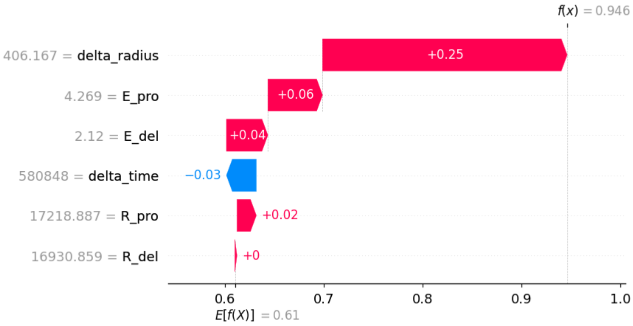


Figure 3.10: Waterfall FP

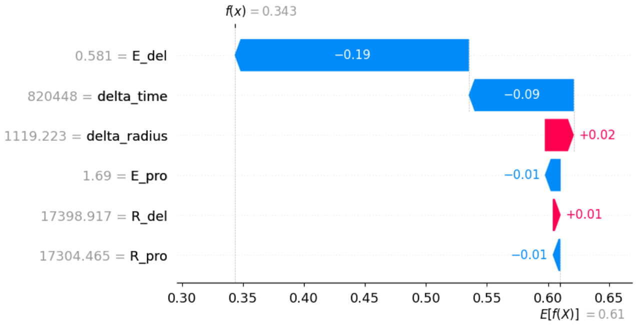


Figure 3.11: Waterfall FN

For the *False Negative* case, the most significant feature contributing to the misclassification of the event is E_{del}, which is approximately 0.6 MeV. This value does not fall within the characteristic peaks of neutron capture but instead lies in a region where background events are much more probable, as can be compared with Figure 3.1. It's important to note that for this particular event, E_{del} is the most influential feature, whereas for most other events, ΔR (`delta_radius`) tends to have a greater contribution. The SHAP value for the ΔR feature is positive, indicating that this feature suggests the event to be a true IBD, but it is not sufficient for correct labeling. Additionally, the (`delta_time`) feature contributes to the misclassification because, with a value of approximately 0.8e6 ns, the event falls into a region where background events are more prevalent. The other features are not decisive in the misclassification as they have very small positive and negative SHAP values and do not significantly contribute to correctly identifying the event.

For the *False Positive* case, the feature that contributes the most to the misclassification is ΔR (`delta_radius`). In this case, a coincidence of background events that are very close to each other spatially is misclassified as IBD primarily for this reason, as seen in Figure 3.10. Additionally, contributing to the incorrect classification of this event are E_{del} and E_{pro}, which have values in regions that are perfectly compatible with the positron spectrum for E_{pro} and compatible with neutron capture for E_{del}. Specifically, E_{del} has a value of approximately 2.2 MeV, which is within the expected range for neutron capture.

In conclusion, the SHAP values and the corresponding plots provide a valuable tool for understanding the decision-making process of the model, highlighting the importance of different features in the classification task.

3.3.3 PyThorch

The ANN, implemented using the PyTorch library, is comprised of one input layer, four hidden layers, and one output layer. The number of neurons in the input layer is determined based on the number of features used in the training dataset, so the input layer has 6 input neurons.

Each hidden layer contains 64 neurons and utilizes the *Rectified Linear Unit* (ReLU), defined as $ReLU(x) = \max(0, x)$, as the activation function. The network eschews an explicit activation function in the output layer and instead produces a direct linear output.

For training the network, it is first instantiated and the computation has been transferred to a CUDA-enabled Graphics Processing Unit (GPU) to leverage hardware acceleration, using a Virtual Machine on CLoudVeneto, thereby enhancing computational efficiency. The *Cross-Entropy* Loss is chosen as the *loss* function due to its efficacy in classification problems. The network's weights are iteratively adjusted through the use of the Adam optimization algorithm.

The training process consists of up to *2000 epochs*; however, an early stopping mechanism is integrated to prevent overfitting and to reduce computational overhead. Early stopping functions as an intelligent termination criterion for the training process of a machine learning model. When the model is being trained on a dataset and ceases to exhibit improvement in its performance on an independent validation set, early stopping intervenes to halt the training. This ensures that the model maintains a robust ability to generalize to unseen data and does not overfit by excessively adapting to the idiosyncrasies of the training dataset. Specifically, the training is terminated if the validation loss does not exhibit improvement for a span of 10 consecutive epochs.

Results

The model showcased remarkable adeptness in accurately classifying true IBD events and segregating them from the background. To obtain a more detailed representation of the model's accuracy and reliability, a confusion matrix was formed, it is presented in Table 3.7. This analysis was executed independently for the IBD and BKG sets. The confusion matrix, when constructed, will disclose the count of true positives, false positives, true negatives, and false negatives. The notably small quantity of incorrectly classified events in both the IBD and BKG datasets reflects the model's success in reducing misclassifications. In addition, a distinct table summarizes the efficiency rates for the classifications of IBD and background events. The exemplary efficiency rates further underscore the model's capability in correctly identifying true IBD incidents and effectively filtering out background noise.

	Neural Network
IBD Efficiency	99.9849%
BKG Efficiency	99.9770%

Table 3.6: Performance Evaluation of the Neural Network

	Predicted IBD	**Predicted BKG**
Actual IBD	200231	46
Actual BKG	30	199726

Table 3.7: Confusion Matrix of the Neural Network

3.4 Model Comparison

It ispossible to conduct a comparison among the models based on the results presented in Table 3.3, Table 3.5, Table 3.7.

The *XGBoost* model demonstrated the fewest misclassifications for both IBD and BKG events. It erroneously predicted only 4 IBD events as BKG and 3 BKG events as IBD. This indicates a high degree of model precision, making it particularly reliable for the identification of both IBD and BKG events.

On the other hand, the *Manual Cut* model showed a significantly higher number of IBD mispredictions, misclassifying 4542 IBD events as background. This could suggest that, although the Manual Cut model has high background efficiency, it may not be the most reliable for IBD event identification.

The *PyTorch Neural Network*, finally, demonstrated a slightly higher number of BKG mispredictions compared to XGBoost, misclassifying 30 BKG events as IBD. However, it had a very low number of IBD mispredictions, misclassifying only 46 IBD events as BKG.

In summary, although all three models showed high efficiency in detecting BKG events, the XGBoost model stands out as the most accurate and reliable model for the identification of both IBD and BKG events, according to the provided data. This makes it the model of choice based solely on these tables.

It is crucial to note that the tables provided offer compelling evidence of the advancement in predictive modeling brought about by machine learning algorithms, notably surpassing traditional Manual Cut algorithm. Comparing the efficiencies of the models, we observe a significant difference between the Manual Cut model and the machine learning models, XGBoost and PyTorch Neural Network.

The Manual Cut model has an IBD efficiency of 97.702%, which, while respectable, is markedly lower than the efficiencies achieved by the machine learning models. The XGBoost model achieves an IBD efficiency of 99.9985%, and the PyTorch Neural Network reaches 99.9849% - both considerably higher than the Manual Cut model.

This difference in IBD efficiency underscores the transformative impact of machine learning in predictive modeling, showing its superior capacity to identify complex patterns within the data. The background efficiencies of all models are comparable, showing high performance across the board. However, the edge given by machine learning in IBD efficiency demonstrates the powerful advantage of these advanced algorithms.

Having clearly shown that the efficiency of machine learning models outperforms that of the Manual Cut model, it becomes crucial to delve further into an intricate comparison between two standout machine learning models: XGBoost and the Neural Network.

To deeply compare the performance of PyTorch and XGBoost, are presented graphs to illustrate how the two algorithms, evaluate the determination of IBD events in comparison to BKG, based on the values of the features ΔR and Δt and R_{pro}.

The graphs presented illustrate how the two algorithms, XGBoost and PyTorch, evaluate the determination of IBD events in comparison to BKG events, based on the values of the features ΔR and Δt. As expected, by observing the distribution of histograms for IBD events only, it is evident that both algorithms perform accurately for events with $\Delta R < 2200$mm, where the counts for BKG events begin to surpass the counts of IBD events. The most significant aspect that allows for a comparison between the models through this plot is the feature Δt. Specifically, for XGBoost, as shown in the graph, Δt does not seem to be very important for determining IBD or BKG events, which is consistent with the feature importance discussed earlier and presented in Figure 3.7. In contrast, for the PyTorch model, the feature Δt appears to have notable importance for the accurate identification of events. Observing the distribution of the Δt feature for IBD events reveals an exponential decay, indicating that many IBD events are expected for very low values of Δt, and gradually decreasing. Combining this trend with the decrease in Δt events, a graph similar to the one presented for PyTorch is expected. Therefore,

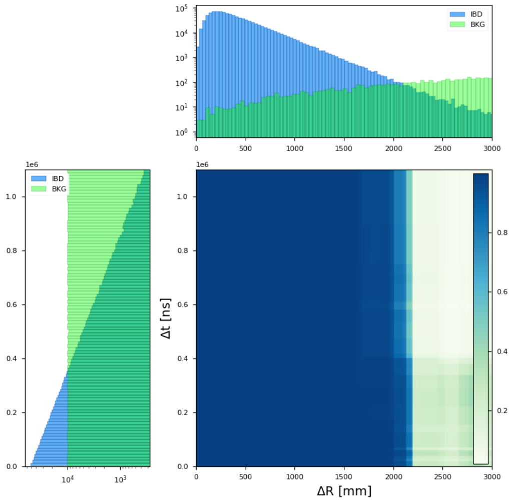


Figure 3.12: (Δt, ΔR) plot

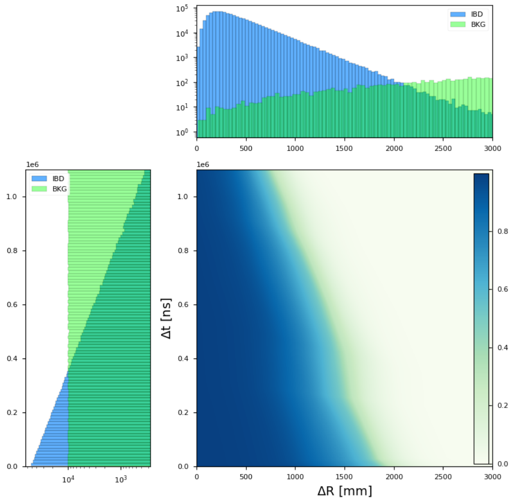


Figure 3.13: (Δt, ΔR) plot

even though the PyTorch model has lower efficiency, it seems to better evaluate the importance of the Δt feature, which is not achieved with the XGBoost algorithm.

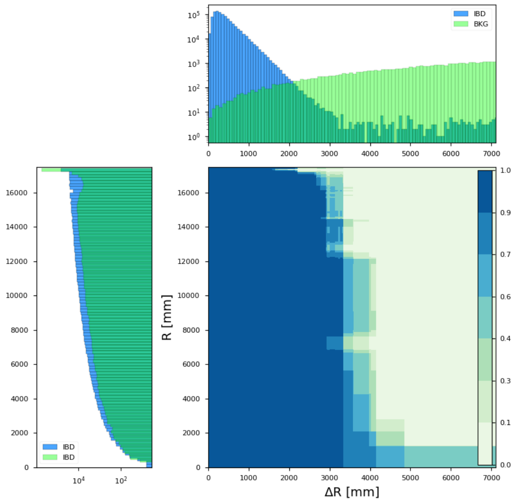


Figure 3.14: (R_{pro}, ΔR) plot

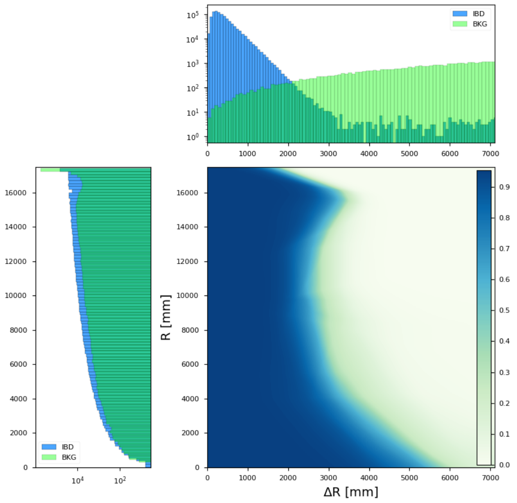


Figure 3.15: (R_{pro}, ΔR) plot

Regarding the plots involving the features (R_{pro}, ΔR), we can observe that both algorithms, XGBoost and PyTorch, have effectively learned to discern the substantial differences between IBD and BKG events based on these features. Specifically, when approaching the boundary, there is a significant presence of BKG events attributed to various materials such as acrylic, steel bars, the glass of the photomultiplier tubes (PMTs), and radon in the water.

More precisely, for $\Delta R > 4000mm$ in XGBoost and $\Delta R > 5500mm$ in PyTorch, there is a clear separation between IBD and BKG events.

However, it is important to consider the distribution of R_{pro}. As observed during the feature presentation, around $R_{pro} \approx 16000$ mm, there is a drop n the count of background (BKG) events, followed by a resurgence. Interestingly, the (R_{pro}, ΔR) PyTorch graph shows that the neural network has successfully captured and utilized this behavior to effectively distinguish

IBD events that fall within this range. This is evident as a peak in probability in the (R_{pro}, ΔR) plot along the R_{pro} axis.

From a physics perspective, this is significant as it demonstrates the algorithm's ability to understand the underlying physical phenomena and interactions that characterize IBD events for the features mentioned.

3.5 Conclusion

- Si secondo me puoi mettere una tabella con colonne i 3 modelli, righe gli eventi/giorno per ogni componente, e due righe finali aggiuntive con efficiency e purity

	Manual Cut	**XGBoost**	**PyTorch**
Reactor	51.4	52.6	52.6
Geo-U	1.033676967385257	1.0579641475791388	1.0578211091531668
Geo-Th	0.30876065259559626	0.3160152648613012	0.3159725390976992
Li9	0.7913933567572716	0.8099878632290803	0.8098783515889386
He8	0.08793259519525239	0.08999865146989781	0.08998648350988206
World__reactors	1.091849264251094	1.1175032554515578	1.1173521672440376
Atmospheric__nu	0.1789916826641138	0.1831972549920587	0.18317248643344886
Fast__neutron	0.10739500959846826	0.1099183529952352	0.10990349186006929
CO__alpha__n	0.05369750479923413	0.0549591764976176	0.05495174593003464
	55.02430995784693	56.31715614979673	56.30954198121709

Efficiency
0.6610106982742568
0.7054165087074837
0.1981388147732311

Purity
0.8949584133205688
0.9159862749602934
0.9158624321672442