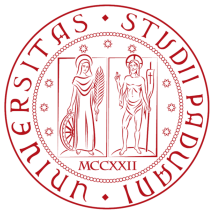


UNIVERSITÀ DEGLI STUDI DI PADOVA

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

Sviluppo di tecniche di Machine Learning per la selezione di interazioni di neutrini da reattori nell'esperimento JUNO

Relatore

Prof. Alberto Garfagnini

Correlatore

Dott. Andrea Serafini

Laureando

Fabio Cufino

Anno Accademico 2022/2023

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. This study focuses on improving the event selection performance in JUNO by applying a machine learning techniques, specifically Boosted Decision Trees (BDT) and Neural Networks (NN), to discriminate between signal events, that are interactions of anti-neutrinos coming from the nearby nuclear power plants from uncorrelated background events. The BDT model demonstrated exceptional performance in event classification, achieving high precision and accuracy in distinguishing between signal events and background events. It achieved a remarkably low number of misclassifications, by surpassing the number of misclassified events by the Manual Cut, the current state-of-the-art for classification. The Neural Network model also exhibited strong performance, with a slightly higher number of misclassifications compared to BDT. However, it showcased a notable capability to accurately identify signal events while maintaining a relatively low number of misclassifications for background events.

Abstract

Il Jiangmen Underground Neutrino Observatory (JUNO) sarà il più grande rilevatore a scintillazione liquido di neutrini al mondo, per il prossimo decennio. Grazie alla sua grande massa attiva (20 kt) e alle prestazioni all'avanguardia (3% di risoluzione energetica a 1 MeV), sarà in grado di effettuare importanti misurazioni nella fisica dei neutrini. Questo studio si concentra sul miglioramento delle prestazioni di selezione degli eventi in JUNO applicando tecniche di apprendimento automatico, nello specifico Boosted Decision Trees (BDT) e Neural Networks (NN), per discriminare tra eventi di segnale, che sono interazioni di anti-neutrini provenienti dalle vicine centrali nucleari, e eventi di fondo non correlati. Il modello BDT ha dimostrato prestazioni eccezionali nella classificazione degli eventi, ottenendo una alta precisione e accuratezza nel distinguere tra eventi di segnale e eventi di fondo. Ha raggiunto un numero notevolmente basso di errori di classificazione, superando il numero di eventi erroneamente classificati dal Manual Cut, l'attuale stato dell'arte per la classificazione. Il modello della NN ha mostrato prestazioni eccellenti, con un numero leggermente maggiore di errori di classificazione rispetto al BDT. Tuttavia, ha dimostrato una notevole capacità di identificare accuratamente gli eventi di segnale mantenendo un numero relativamente basso di errori di classificazione per gli eventi di fondo.

Contents

Chapter 1

Introduction

The Jiangmen Underground Neutrino Observatory (JUNO) [1], currently under construction in south 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, addressing the Neutrino Mass Ordering (NMO) problem.

The JUNO experiment, thanks to its excellent energy resolution and large fiducial volume, is expected to make significant contributions to this field.

1.1 Neutrino Oscillations

Neutrinos, originally assumed to be massless, have been experimentally proven to have non zero mass. This result is mainly due to the discovery of neutrino oscillations, a quantum mechanical phenomenon where a neutrino changes its flavor during propagation, providing strong evidence of their mass [2].

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 which ν_i are the three mass eigensates, with non-degenerate masses m_i $(i = 1, 2, 3)$.

The matrix U_{PMNS} [3], [4], [5], the so-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. In addition, the Majorana CP phases $\eta_i (i = 1, 2)$, are only physically possible if neutrinos are Majorana-type particles and do not participate in neutrino oscillations. Therefore, U_{PMNS} can be expressed as:

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

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 $i, j = 1, 2, 3, i > j$. In the case of JUNO, two nuclear power reactors 53 km away from the detector, produce electron anti-neutrinos $\bar{\nu}_e$ with energy below 10 MeV, as principal sources of neutrinos for the experiment. Therefore, the survival probability $P(\bar{\nu}_e \to \bar{\nu}_e)$ of electron antineutrinos reads:

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

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 three mixing angles.

Current values for the neutrino oscillation parameters are reported in Table 1.1 [6]. The last column denotes the relative uncertainties, expressed in percentage.

Parameter	**Value**	**Relative Uncertainty**
Δm^2_{32} (NO)	$(2.453 \pm 0.034) \times 10^{-3}\,\text{eV}^2$	1.4%
Δm^2_{32} (IO)	$-(2.546 \pm 0.037) \times 10^{-3}\,\text{eV}^2$	1.5%
Δm^2_{21}	$(7.53 \pm 0.18) \times 10^{-5}\,\text{eV}^2$	2.4%
$\sin^2\theta_{12}$	0.307 ± 0.013	4.2%
$\sin^2\theta_{13}$	0.0218 ± 0.0007	3.2%

Table 1.1: Neutrino oscillation parameters within the reach of JUNO and their 1σ uncertainties, as reported in PDG2020. [6]

JUNO's primary objective is to improve these measurements and to determine 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} alters the curves of Figure 1.1.

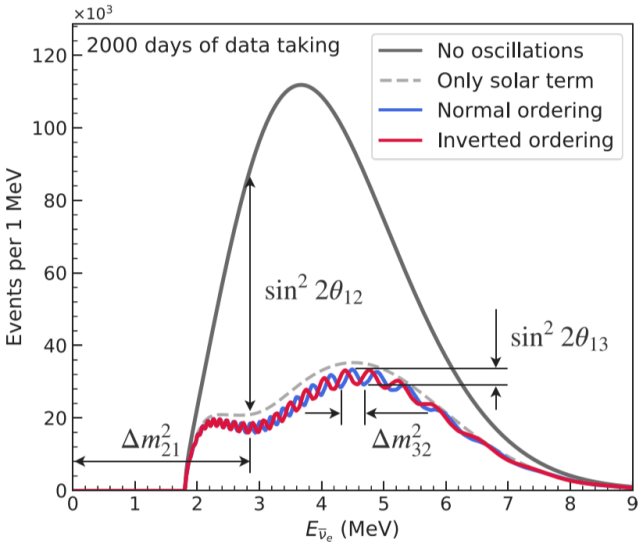


Figure 1.1: JUNO's reactor antineutrino energy spectrum is shown with and without the effect of neutrino oscillation. The gray dashed curve includes only the term in the disappearance probability modulated by $sin^2(2\theta_{12})$, while the blue and red curves use the full oscillation probability for 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. [7]

1.2 The JUNO detector

The Jiangmen Underground Neutrino Observatory is currently being constructed beneath Dashi Hill in the town of Jinji, Southern China. Its placement 43 km southwest of Kaiping city was strategically chosen to significantly reduce background from cosmic rays due to its underground location. JUNO is anticipated to detect antineutrinos, predominantly originating from the Taishan and Yangjiang nuclear power plants. NPPs are approximately 52.5 km away from the JUNO detector and will provide a combined nominal thermal power of 26.6 GW_{th}.

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

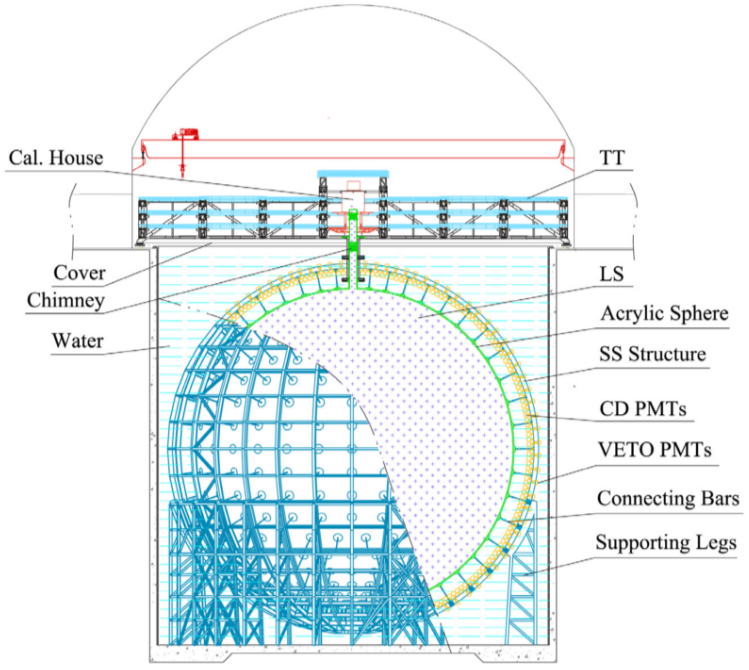


Figure 1.2: Schematic view of the JUNO experiment, from [1]

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 $\sigma_E/E = 3\%/\sqrt{E(MeV)}$ [1].

The CD houses 20 kton of liquid scintillator (LS) within a spherical acrylic vessel, submerged in a water pool. This pool, with a diameter of 43.5 m and a height of 44 m, not only serves as an effective buffer to shield the LS from the radioactive influence of the surrounding rock, but it acts as active scintillation medium. Equipped with PMTs for Cherenkov detection, the pool plays an instrumental role in the active shielding process.

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.

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.

1.3 JUNO signal and background

1.3.1 Signal

The JUNO experiment primarily draws its sources from the Taishan and Yangjiang NPPs, which are made of two and six cores, respectively. In addition to these, the Daya Bay reactor complex contributes to the antineutrino flux. The reactor power and baselines from the Taishan, Yangjiang, and Daya Bay reactor cores are detailed in Table 1.2.

Reactor	Power [GW_{th}]	Baseline [Km]
Taishan	9.2	52.71
Yangjiang	17.4	52.46
Daya Bay	17.4	215

Table 1.2: Information on nuclear reactors

JUNO employs a Liquid Scintillator 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.

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 Inverse Beta Decay (IBD) reaction:

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

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: $e^+ + e^- \rightarrow 2\gamma$. The energy deposited by the positron directly correlates with the antineutrino energy, providing a precise measure critical for neutrino oscillation studies.

Following the prompt signal, the neutron is captured primarily on hydrogen (approximately 99% of the time) after an average delay of about 220 µs. This capture event releases a single 2.2 MeV photon, creating the **delayed signal**. Occasionally, the neutron is captured on carbon (around 1% of the time), resulting in a gamma-ray signal with a total energy of 4.9 MeV [7]. The processes are described as follows:

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

$$n + {}^{12}\mathrm{C} \rightarrow {}^{13}\mathrm{C}^* \rightarrow {}^{13}\mathrm{C} + \gamma \tag{1.3}$$

The scintillation light output originating 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 with high precision, which is crucial for studying neutrino oscillation.

1.3.2 Background

The design and composition of the scintillator in the JUNO experiment have been optimized to minimize background coming from various radiation sources. Despite these efforts, several types of background signals are inevitably produced inside the detector. For the purpose of the 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 the JUNO detector is made of, including the acrylic used for the detector walls, the metal structure supporting the detector, the 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, 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.

The coincidence of two otherwise uncorrelated events, typically of radiogenic origin, forms the so-called *Accidental Background*. This background dominates the low energy part of the spectrum due to its nature.

Accidental background can potentially mimic the signal from IBD in several ways:

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

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

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

The goal of this thesis work is to study these accidental background events, a topic that will be discussed in detail in the following sections.

The term *Correlated Background* refers to cosmogenic backgrounds and other sources of $\bar{\nu}$, which will be discussed in detail below.

These background sources stem from a singular physics process and generate both a prompt and delayed signal, closely mimicking the IBD events originating from the reactor as listed in Table 1.2. These events share essential characteristics with IBD events, including a prompt signal, a delayed signal, and a similar temporal separation. As a result, these events are indistinguishable from IBD events, presenting substantial challenges in terms of background reduction through cuts or other more sophisticated techniques.

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. 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 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 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. Natural radioactivity exists in materials present in the Earth's crust and mantle, such as ^{238}U,^{232}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.

World reactors serve as a significant source of antineutrinos. These reactors, utilized for the production of electrical energy through the process of nuclear fission, also emit antineutrinos as a byproduct of this process. With a total of 832 nuclear reactors globally [8], the collective emission of antineutrinos becomes significant.

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:

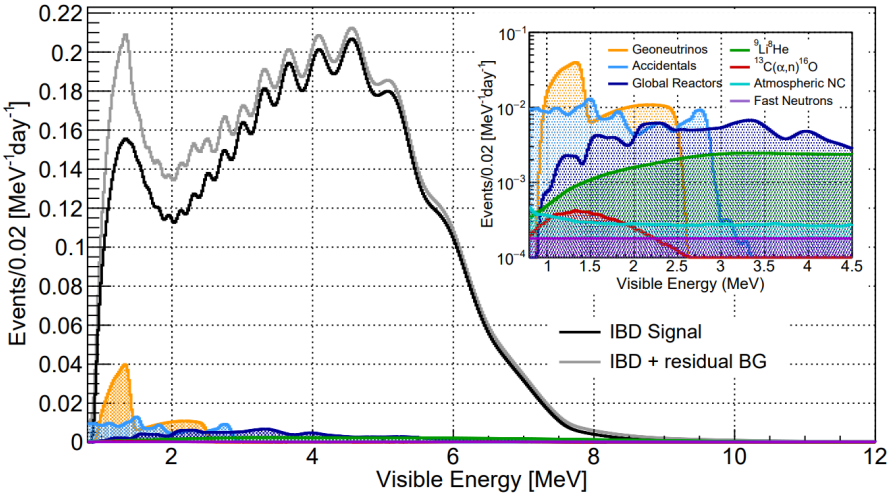


Figure 1.3: Visible energy spectrum with (grey) and without (black) backgrounds. The predicted backgrounds, which make up around 7% of the whole sample of IBD candidates and are primarily confined below $\approx$ 3 MeV, are shown in the inset as spectra, from [7].

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 study their contribution. 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 event selection 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.

Chapter 2

Machine Learning

2.1 Introduction

Machine learning, a subfield of artificial intelligence, is a scientific discipline that focuses on the development of algorithms and statistical models that enable computers to carry out specific tasks by learning patterns from data. It involves the creation of models that can be supervised (learning from labeled data), unsupervised (learning from unlabeled data), semi-supervised, or with reinforcement (learning based on reward/punishment system). Machine learning has a wide range of applications, including but not limited to, natural language processing, image recognition, and recommendation systems [9].

Within the framework of the JUNO experiment, machine learning algorithms are utilized to analyze the data, discerning patterns that represent IBD events, thereby effectively differentiating them from the background.

2.1.1 Supervised Learning

Supervised learning is a fundamental aspect of machine learning, where algorithms are trained using labeled datasets. In this approach, the algorithm learns from examples that are already labeled with the correct answers. The goal is to develop a function that accurately maps input data to corresponding outputs. In the context of this thesis, the focus is on binary classification in the JUNO experiment. The objective is to classify a given event pair as either a correlated inverse event pair or an accidental background event pair, utilizing the event pair's features.

Two different machine learning algorithms, **Boosted Decision Trees** (BDT) and **Neural Network** (NN), are deployed to to perform the classification task.

2.2 Boosted Decision Trees

Gradient Boosting is a machine learning technique that leverages the concept of boosting, combined with the methodology of gradient descent. The objective is to construct a robust predictive model by amalgamating multiple weak learners, which are typically decision trees [10].

The unique aspect of Gradient Boosting, compared to traditional boosting techniques, is its method of error correction. Rather than altering the weights of misclassified instances, the model fits each subsequent tree to the residuals (or the negative gradient) of the loss function

relative to the prediction of the existing ensemble of trees. This implies that each new tree is trained to predict the error of the current model, thereby progressively reducing the overall error.

The process is formalized as follows:

1. **Initialization**: The model is initiated with a constant value, 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 is 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.

2. **Computation of Residuals**: The model iteratively constructs an ensemble of M trees. For each iteration $m = 1$ to M, the residuals are calculated as

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

for each instance $i = 1, 2, ..., N$. These residuals are essentially the negative gradients of the loss function with respect to the model's predictions

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 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

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

Here, ν represents the learning rate, a parameter typically less than 1, which controls the contribution of each tree to the final prediction.

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 correction to the predictions of the previous trees, collaboratively reducing the loss function's value and improving the overall model's performance [10].

XGBoost is an highly efficient implementation of this method, which introduces several improvements such as parallel processing [11].

2.3 Neural Networks

Neural Networks (NNs) are computational models that draw inspiration from the interconnected structure of the human brain [12].

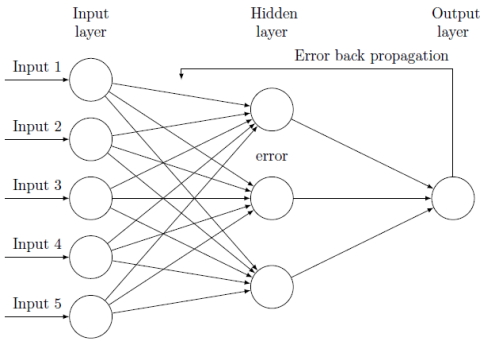
An artificial neuron takes inputs $x = [x_1, x_2, ..., x_n]$, applies weights $w = [w_1, w_2, ..., w_n]$ to the inputs, sums them, and adds a bias term b. Mathematically, this operation can be represented as:

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

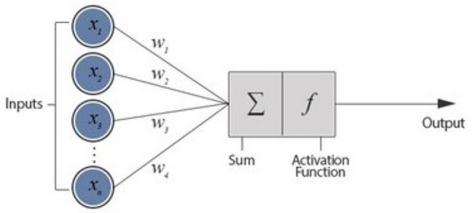
The calculated value, z, is then passed through an *activation function*, f, to generate the neuron's output, $a = f(z)$ [12].

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 [13].

An Artificial Neural Network (ANN) consists of interconnected layers of neurons, including an input layer, one or more hidden layers, and an output layer. Each layer is fully connected to the next layer. A graphical representation of an Artificial Neural Network and a single neuron is presented in Figure 2.1a and Figure 2.1b [13].



(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 [13].

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*, an iterative optimization algorithm for finding the minimum of a function, in this case, the loss function. It is important to note that gradient descent is just one of many optimization techniques that can be used for this purpose, but it is widely used due to its efficiency and simplicity. This process iteratively adjusts the parameters to minimize the loss function [14]:

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

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

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 [15].

IBD Dataset

The primary dataset, known as the IBD dataset, is specifically designed for analyzing potential Inverse Beta Decay events, assuming that the sources of antineutrinos are the reactors (Table 1.2) providing a combined output of 57.4 events/day. Importantly, the dataset does not contain any contribution from background events.

The IBD dataset has the following key features:

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

Upon completion of the comprehensive simulation, which incorporates the modeling of the trigger system to accurately reflect its efficiency, the resulting dataset has a total of 2,977,856 events.

BKG Dataset

The second dataset is exclusively composed of events related to radioactivity. The file structure of the BKG dataset remains identical to that of the IBD dataset. However, due to the uncorrelated nature of events in the BKG dataset, a unique SimID is assigned to each event, ensuring its distinctiveness from all other events.

In Table 3.1 are shown rates that are used as inputs for the simulation process.

Dataset Name	Rates
U238@LS	3.234 Hz
Th232@LS	0.733 Hz
K40@LS	0.53 Hz
Pb210@LS	17.04 Hz
C14@LS	$3.3 \cdot 10^4$ Hz
Kr85@LS	1.163 Hz
U238@Acrylic	98.41 Hz
Th232@Acrylic	22.29 Hz
K40@Acrylic	161.25 Hz

Dataset Name	Rates
U238@node/bar	2102.36 Hz
Th232@node/bar	1428.57 Hz
K40@node/bar	344.5 Hz
Co60@node/bar	97.5 Hz
U238@PMTGlass	$4.90 \cdot 10^6$ Hz
Th232@PMTGlass	$8.64 \cdot 10^5$ Hz
K40@PMTGlass	$4.44 \cdot 10^5$ Hz
Tl208@PMTGlass	$1.39 \cdot 10^5$ Hz
Rn222@WaterRadon	90 Hz

Table 3.1: Table of isotopes, and rates used as simulation inputs

This background represents all the contributions from radioactive decays (α, β, γ) both internal and external to the detector, but that have deposited energy inside the detector.

The rates provided in the Table 3.1 serve as assumptions regarding the occurrence of events from different isotopes and their respective decay rates. However, it is important to consider that the actual event rate in the dataset is influenced by various factors, including trigger thresholds, event multiplicity, energy deposition location, which may extend beyond the detector, and experimental conditions. As a result, the observed event rate in the final dataset deviate from the initially assumed rates.

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

- **Liquid Scintillator (LS):** The main part of the detector, where isotopes U238, Th232, K40, Pb210, C14, and Kr85 are found. It's worth noting that C14 contributes with a high decay rate of 33 kHz.
- **Acrylic:** This constitutes the detector walls, hosting isotopes U238, Th232, and K40.
- **Node/bar:** This is the metallic structure supporting the detector. Here, isotopes U238, Th232, K40, and Co60 are located.
- **PMT Glass:** The glass of the photomultipliers containing isotopes U238, Th232, K40, and Tl208. The isotope U238 stands out with an extremely high decay rate of 4.9 MHz.
- **WaterRadon:** This represents the gas present in the water surrounding the detector and inside the detector during the initial filling stages.

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 produce a signal in 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 MHz. This high rate indicates that U238 within the PMT Glass is highly active and undergoes decay at a very rapid pace. Similarly, Th232@PMTGlass also exhibits a high decay rate, around 864 kHz. Like U238, Th232 in the PMT Glass is highly active. Such high activity will contribute substantially to the background within the detector.

The dataset is made of comprises a total of 8,841,188 recorded events, obtained from a one-day exposure.

3.2 Feature creation

The process of feature engineering plays a pivotal role in the successful development of models for detecting IBD events. It is a critical step where the majority of the model's performance is determined. This crucial process involves loading the two distinct datasets mentioned earlier: one for IBD events and the other for radioactivity background.

3.2.1 IBD Features Table

As we mentioned earlier, an IBD event is characterized by two correlated 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 is the characteristic energy released during ionization and subsequent annihilation of a positron with an electron in the scintillator liquid.

- $\mathbf{E_{delayed}}$: This feature represents the energy of the delayed signal. It is the energy released when a neutron is captured by the scintillator liquid.

- $\mathbf{\Delta t}$: 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.

$$\Delta R = \sqrt{(x_i - x_j)^2 + (y_i - y_i)^2 + (z_i - z_j)^2)}$$

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. An event pair is assigned a label of 1 if it represents a true IBD event generated as part of the same simulation. On the other hand, a label of 0 indicates uncorrelated background events that originate from different simulations. 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.

Efficient Feature Calculation

To enhance computational speed due to the large dataset size and the complexity of feature calculation, a parallel computing approach was implemented. The calculations were carried out on a virtual machine hosted on Cloud Veneto, which had 14 CPU cores available. The feature calculation task was divided into multiple sub-tasks, allowing them to be executed simultaneously by different CPU cores. This parallelization strategy significantly decreased the overall computation time required for the task.

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 only about 0.7% of IBD events that occur outside this time window, but this trade-off is deemed acceptable.

The distribution of the features are presented in the plots of Figure 3.1.

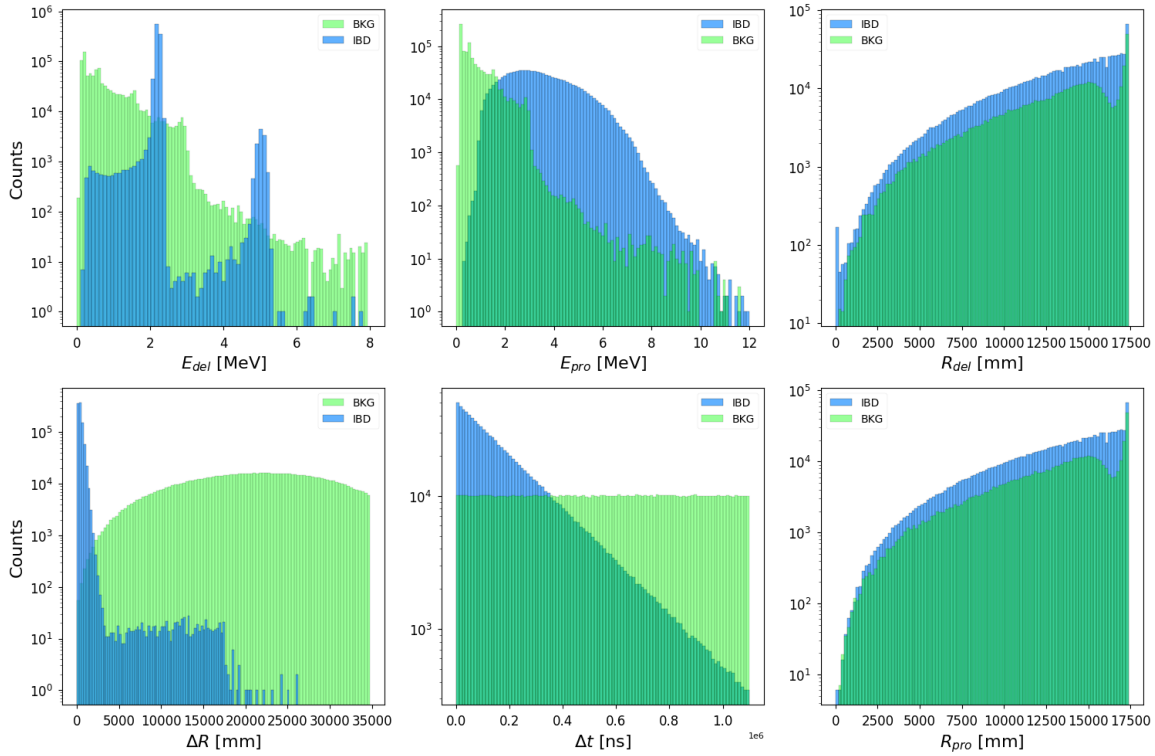


Figure 3.1: Features histograms

Feature distribution

In the graph of Figure 3.1, focusing only on IBD events, it is clearly observable for E_{del} feature, the distinct characteristics of the IBD events, such as the peaks at $2.2MeV$ and $4.9MeV$, and the clearly visible positron spectrum in the E_{pro} feature. The distribution of R_{del} and R_{prompt} follows the expected pattern for events that are uniformly distributed within the detector. This distribution aligns with the spherical shape of the detector, where IBD events are more likely to occur in the outer region due to the greater volume of liquid scintillator in that area. The observed distribution adheres to the expected R^3 distribution characteristic of uniformly distributed events within a spherical detector.

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, as expected for accidental coincidences. This allows to distinguish IBD events from BKG events. For the Δt distribution of BKG events, it is observed to be nearly constant across all possible Δt values. As the energy E_{del} increases, the number of E_{del} events decreases. Despite the presence of well-defined peaks in the data, these peaks do not correspond to the expected peaks from neutron capture.

Another notable characteristic is the remarkable similarity between the spectra of the prompt and delayed signals, providing evidence that they arise from accidental coincidences.

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 $R \approx 16500$ mm 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, in the outer regions of the detector, where the collection of scintillation light is less efficient.

3.3 Models

This chapter introduces several algorithms, starting with a manual cut-based approach, **Manual Cut**, which sets criteria based on event physics and background noise. Manual Cut is considered the current state-of-the-art method and serves as the benchmark against which other methods are compared. Additionally, machine learning-based algorithms, specifically **Boosted Decision Trees** and **Neural Networks**, are discussed.

Dataset pre-processing

The dataset used for model evaluation was pre-processed by combining the feature tables for IBD and BKG events in a random manner. To address class imbalance, surplus BKG event

pairs were removed.

The dataset was divided into three subsets: training, validation, and test datasets. The training dataset, which accounted for 80% of the original dataset, was used for model training. The test dataset, constituting 20% of the original dataset, provided an independent evaluation of the final models. Additionally, a validation dataset was created, representing 10% of the training set, and it allowed for performance evaluation during the training process.

By following this methodology, we ensured reliable comparisons and drew meaningful conclusions from the evaluation of the models.

3.3.1 Manual Cut

The algorithm is designed to suppress various types of background while maintaining high efficiency for 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 employed as outlined in the referenced paper [7].

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 (Figure 3.3) and the radial distance (Figure 3.2) 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 is determined based on the characteristic time scale of neutron thermalization and capture, denoted as τ. This characteristic time scale represents the average time it takes for a neutron to undergo thermalization and subsequently get captured, while the 1.5 m spatial cut considers the short distance neutrons typically travel before capture.

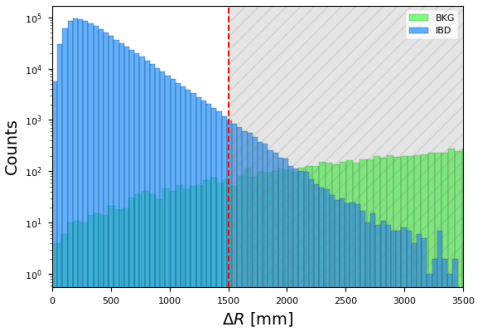


Figure 3.2: ΔR cut

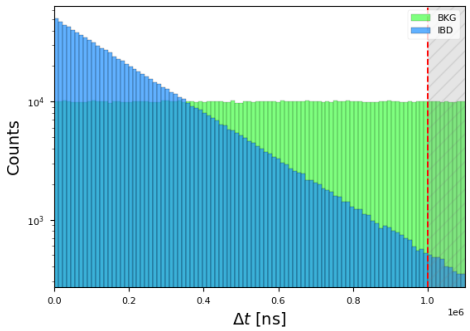


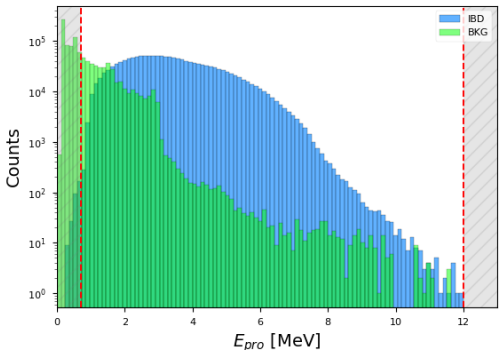
Figure 3.3: Δt cut

2. **Energy of the Prompt Signal (E_{pro}) Cut**: The next cut is applied on the energy of the prompt signal (Figure 3.4), 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 based on the fact that the IBD events predominantly occupy this energy range. The energy of the prompt signal is associated with the energy of the positron coming 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

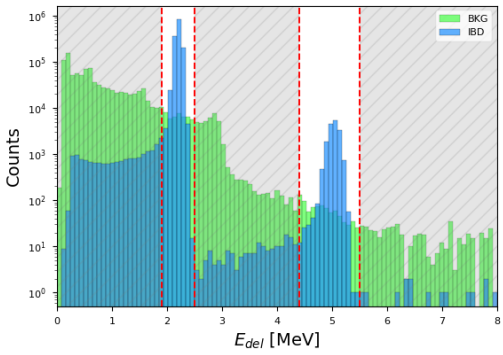


3. **Energy of the Delayed Signal (E_{del}) Cut**: The final cut is applied on the energy of the delayed signal (Figure 3.5), which is the signal produced by the neutron capture that follows the antineutrino interaction. The criteria are:

    - 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) 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 the test dataset.

	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 a 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 configured with specific settings and hyperparameters:

- **Random seed (`seed`)** : Set to 1 for reproducibility, ensuring consistent random number generation.
- **Number of estimators (`n_estimators`)** : Configured with an initial value of 10,000 decision trees, controlling model complexity. However, during training, the number of estimators was automatically determined using the early stop condition, which monitors the model's performance on a validation set.
- **Learning rate (`learning_rate`)** : Set at 0.05, dictating each tree's contribution to the final prediction.
- **Maximum tree depth (`max_depth`)** : Limited to 3, controlling the complexity of each tree.

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 evaluated various settings and hyperparameter combinations to identify the optimal configuration that maximizes model accuracy.

Results

The algorithm exhibited higher 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 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 the test dataset.

	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) [16], 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 shapley value, ϕ_i, for feature i is calculated using:

$$\phi_j(x) = \sum_{S \subseteq M \setminus \{j\}} \frac{|S|!(M - |S| - 1)!}{M!} [f(x_S \cup \{j\}) - f(x_S)] \tag{3.1}$$

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 $|S|!(|M| - |S| - 1)!/|M|!$ 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 IBD events and background events.

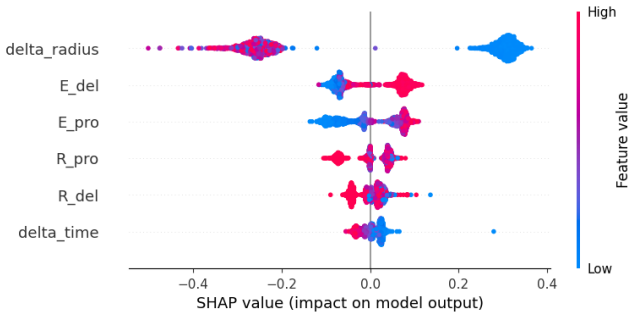


Figure 3.6: XGBoost Summary Plot

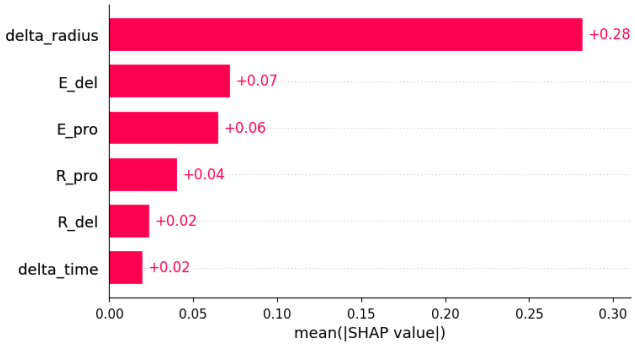


Figure 3.7: XGBoost feature importance

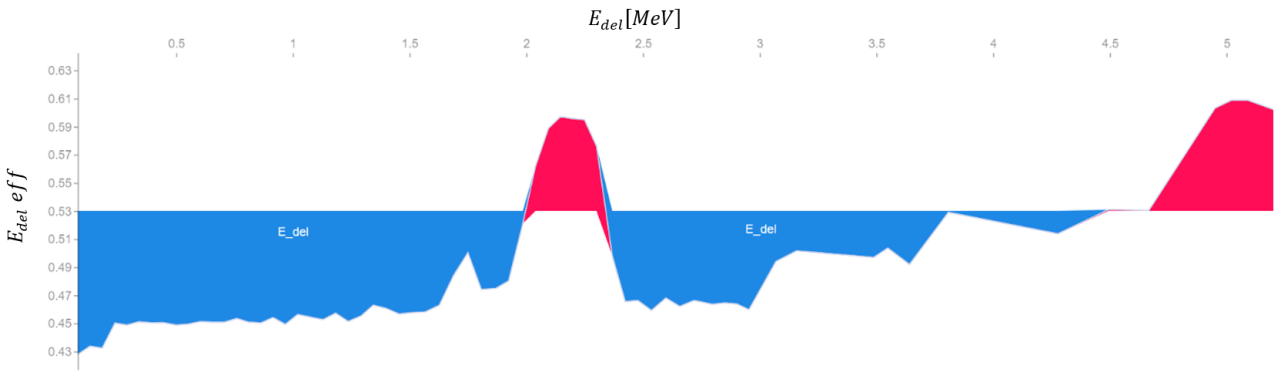
The presented graphs depict the importance of each feature used by the algorithm for learning, measured by calculating the SHAP values. On Figure 3.7, the x-axis represents the mean absolute SHAP value for each feature. The first key characteristic of the model is evident here: the most importance feature in classification is ΔR. Here, the importance 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 of Figure 3.1, it was already observable that ΔR is the feature that separates the IBD class most distinctly from the BKG class.

In the left summary plot of Figure 3.6, the x-axis represents the SHAP value and the y-axis represents the 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, indicating events accurately identified as IBD. 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 in 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.

Exploring in-depth the analysis of E_{del} 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, the $E_{del}eff$, starting from a '$base\ value = 0.53$' that is the average of the model's predictions across the entire training dataset. It is observed that around the characteristic peaks of neutron capture at approximately $2.2MeV$ and $4.9MeV$, the algorithm has successfully learned to distinguish them from background events with high accuracy.



Regarding the E_{pro} feature, of Figure 3.1, for values below 1 MeV, the histogram is predominantly occupied by BKG events, and as seen from the summary plot (Figure 3.6), 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 as E_{pro}, E_{del} and ΔR to the algorithm's ability to discern between the two classes from their distribution, as there are no clear differences between the feature distribution for IBD and BKG.

Based on the aforementioned information, Figure 3.9 and Figure 3.8, show waterfall plots. Two typical events have been chosen, one for IBD and one for BKG. These plots visually display the individual contributions of each feature to the model's final prediction, which is 1 for IBD events and 0 for background events. The starting point of these plots is the '*base value*', mentioned before. 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.

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`,

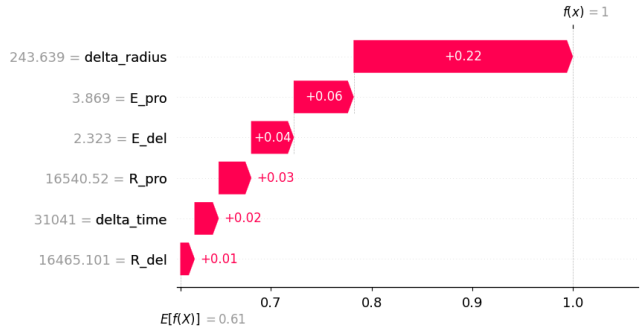


Figure 3.8: Waterfall IBD

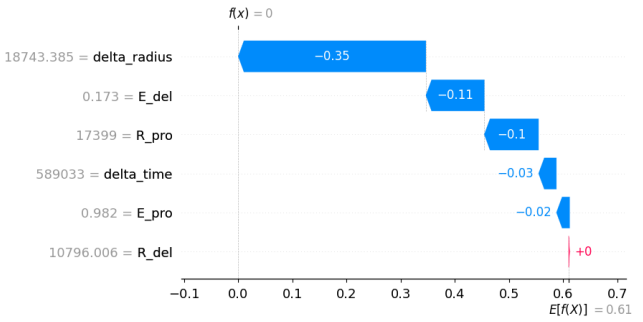


Figure 3.9: Waterfall BKG

`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* (Figure 3.10) and *False Negative* (Figure 3.11) predictions:

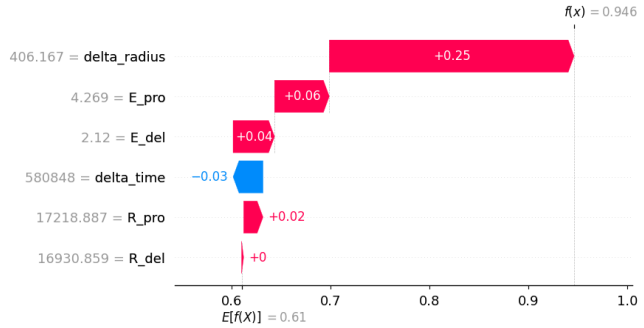


Figure 3.10: Waterfall FP

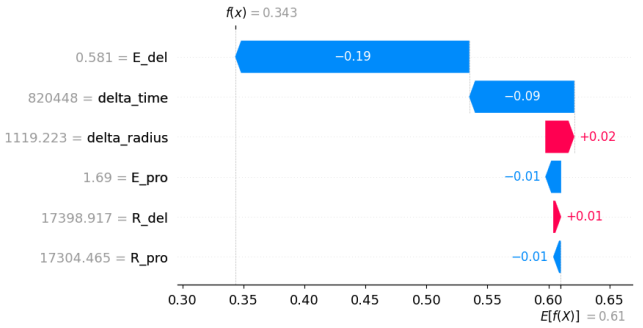


Figure 3.11: Waterfall FN

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 the event falls into a region where background events are 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 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 made 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 six 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. It is not defined an explicit activation function in the output layer of the network, opting instead for 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, equipped with *NVIDIA T4 Tensor Core GPU*, thereby enhancing computational efficiency. The *Cross-Entropy* Loss is chosen as the *loss* function. 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 showed 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 created, and is shown in Table 3.7. 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 high 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

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

Table 3.7: Confusion Matrix

3.4 Model Comparison

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

The *XGBoost* model demonstrated the lowest 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. Nonetheless, this model approach ffers effective background rejection at the expense of a slight decrease in efficiency.

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.

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 modelling, 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.

XGBoost and 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 of Figure 3.12 and Figure 3.13 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.

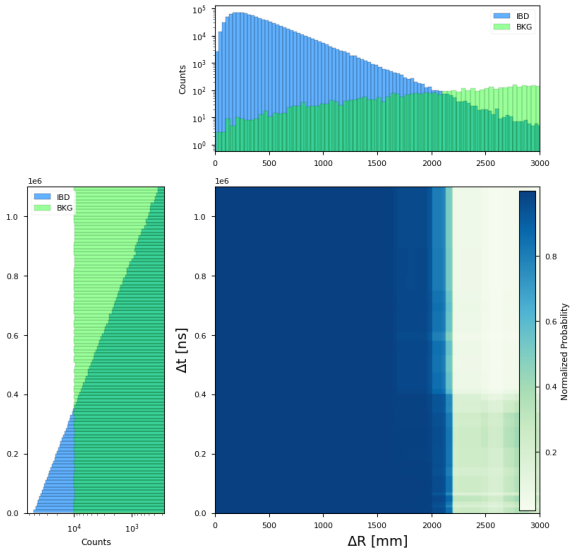


Figure 3.12: (ΔR, Δt) plot

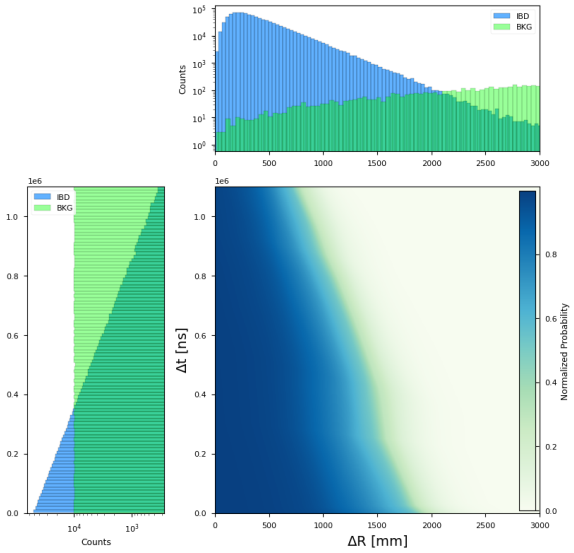


Figure 3.13: (ΔR, Δt) plot

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. 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, 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.

The graphs in Figure 3.14 and Figure 3.15 demonstrate the evaluation of XGBoost and PyTorch in determining IBD events compared to BKG events, using the feature values of ΔR and R_{pro}.

Both algorithms, XGBoost and PyTorch, have effectively learned to distinguish 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 attributable 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 in the count of background events, followed by a resurgence. Interestingly, the (R_{pro}, ΔR) PyTorch graph (Figure 3.15) shows that the neural network has successfully captured and utilized this behaviour to effectively distinguish IBD events that fall within this range.

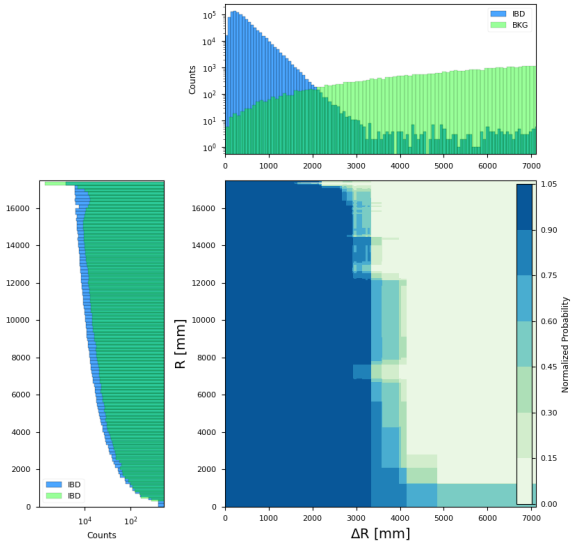


Figure 3.14: (ΔR, R_{pro}) plot

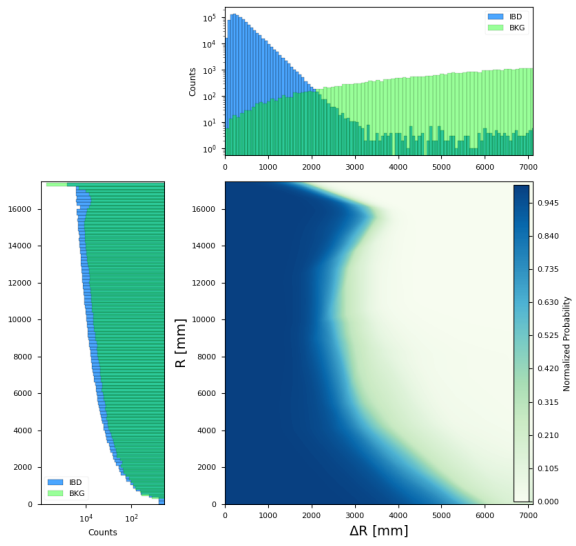
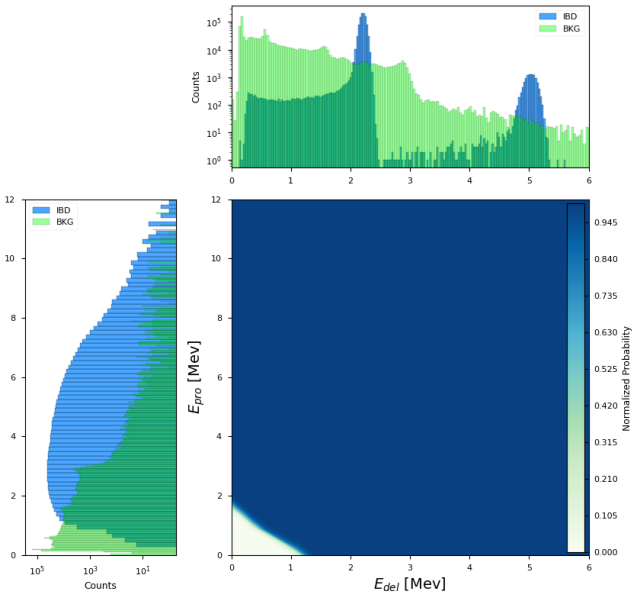


Figure 3.15: (ΔR, R_{pro}) plot

This is evident as a peak in probability in the (R_{pro}, ΔR) plot along the R_{pro} axis. It is important to analyze the behavior of the neural network with respect to the E_{del} and E_{pro} features. As illustrated in Figure 3.16, the Neural Network (NN) encounters challenges in effectively distinguishing Inverse Beta Decay events that occur within the characteristic peaks of E_{del}.

Figure 3.16: (E_{del}, E_{pro}) PyTorch



This limitation can be primarily attributed to the insufficient training of the Neural Network (NN), as demonstrated by the observed plot of E_{pro} and E_{del} and the subsequent outcomes, which clearly demonstrate a notable discrepancy when compared to the performance achieved by the Manual Cut approach. The precise reasons for this inadequacy remain uncertain, and it is plausible that a combination of factors contributes to this phenomenon. Potential factors may comprise various training constraints, or challenges in optimizing the network's hyperparameters. Additionally, the simplistic architecture of the NN, with limited layer. Furthermore, the presence of outliers within the energy values used for training, along with subsequent scaling operations, might introduce further complexities and adversely affect the NN's discriminative capabilities. A comprehensive analysis of these factors, accompanied by rigorous experimentation and fine-tuning, is necessary to elucidate the underlying causes and devise potential solutions for enhancing the NN's performance in this context.

Efficiency and purity

To better evaluate the models' selection capabilities, the selection efficiency and purity are calculated based on the number of events per day from source neutrinos, *Accidental Background*, and *Correlated Background*. These calculations involve a combination of factors, including Event/Day and a muon cut, applied to a variety of event types. These include Reactor events, our primary source of antineutrinos, and Correlated Background events, such as Geo-U, Geo-Th, Li9, He8, World Reactors, Atmospheric Neutrinos, Fast Neutrons, and $C(\alpha, n)^{16}O$. The results are presented to in Table 3.8. The expected IBD events from Accidental background are also evaluated, as detailed in Table 3.9.

For each event type, the selection efficiency for IBD (shown in Tables 3.3, 3.5, and 3.7) is multiplied by the `muon cut` and the events per day to derive the number of IBD events expected when applying the model to the true events generated each day. The calculation for expected IBD events from Accidentals is performed also by multiplying the `muon cut`, the events per day, and the background efficiency reported in the table, minus 1.

	ev/day	muon cut	Manual Cut	XGBoost	PyTorch
Reactor	57.4	0.916	51.4	52.6	52.6
Geo-U	1.155	0.916	1.03	1.06	1.06
Geo-Th	0.345	0.916	0.31	0.32	0.32
Li9	0.81	1	0.79	0.81	0.81
He8	0.09	1	0.09	0.09	0.09
World Reactors	1.22	0.916	1.09	1.12	1.12
Atmospheric ν	0.2	0.916	0.18	0.18	0.18
Fast neutron	0.12	0.916	0.11	0.11	0.11
$C(\alpha, n)^{16}O$	0.06	0.916	0.05	0.05	0.05
Total	–	–	55.02	56.32	56.31

Table 3.8: IBD expected from *Reactors* and *Correlated Background*

	ev/day	muon cut	Manual Cut	XGBoost	PyTorch
Accidentals	134124.0	0.916	22.69	18.22	209.01

Table 3.9: IBD expected from *Accidental Background*

Finally, the efficiency and purity were calculated as the number of selected IBDs divided by the overall IBDs and the number of selected IBDs divided by the total selected events, respectively. This method provides an estimation of the selection algorithm's efficiency and purity, reported in Table 3.10.

	Manual Cut	XGBoost	Neural Network
Purity	0.6610	0.7054	0.1981
Efficiency	0.8949	0.9160	0.9158

Table 3.10: Performance Evaluation

Chapter 4

Conclusions

In this thesis, we have conducted an investigation of models for event classification in the JUNO experiment. Our goal was to compare the performance of different models and evaluate their effectiveness in distinguishing between Inverse Beta Decay events and background events.

The Manual Cut model, XGBoost model, and PyTorch Neural Network were explored as potential solutions for event classification.

Based on the evaluation of overall accuracy and purity, the XGBoost model emerges as the optimal choice for the identification of both Inverse Beta Decay and background events. It consistently demonstrates high accuracy and purity in distinguishing between the two event types. The PyTorch Neural Network model also exhibits strong accuracy in recognizing IBD events, although its purity is slightly lower compared to XGBoost. On the other hand, the Manual Cut model, while having high background efficiency, shows lower overall accuracy when compared to the machine learning models.

However, there are still several areas that deserve further exploration and hold potential for future advancements:

- Feature Enhancement: Incorporating additional features, such as reconstructed energy, vertex position, and spatial distribution of hits, could potentially improve the performance of the models and enhance event classification accuracy.

- Model Optimization: Fine-tuning the hyperparameters of the machine learning models and exploring different architectures could further enhance their performance and robustness.

Bibliography

[1]A. Abusleme et al., "Juno physics and detector", Progress in Particle and Nuclear Physics **123**, 103927 (2022), https://www.sciencedirect.com/science/article/pii/S0146641021000880.

[2]T. Kajita, "Nobel lecture: discovery of atmospheric neutrino oscillations", Reviews of Modern Physics **88**, 10.1103/revmodphys.88.030501 (2016), https://doi.org/10.1103/revmodphys.88.030501.

[3]Z. Maki, M. Nakagawa, and S. Sakata, "Remarks on the unified model of elementary particles", Progress of Theoretical Physics **28**, 870 (1962).

[4]S. Bilenky, "Bruno pontecorvo and the neutrino", Physics-Uspekhi **184**, 531 (2014).

[5]B. Pontecorvo, "Neutrino experiments and the problem of conservation of leptonic charge", Zh. Exsp. Teor. Fiz. **53**, 1717 (1967).

[6]R. L. Workman et al. (Particle Data Group), "Review of Particle Physics", PTEP **2022**, 083C01 (2022).

[7]A. Abusleme et al., "Sub-percent precision measurement of neutrino oscillation parameters with juno", Chinese Physics C **46**, 123001 (2022), https://doi.org/10.1088/1674-1137/ac8bc9.

[8]*Iaea reactor database (rrdb)*, Online, https://nucleus.iaea.org/rrdb/.

[9]T. Mitchell, *Machine learning* (McGraw Hill, 1997).

[10]J. H. Friedman, "Gradient boosting decision trees explained", Annals of Statistics, 1189 (2001), https://www.mdpi.com/2072-4292/12/12/1973/pdf?version=1592572050.

[11]T. Chen and C. Guestrin, "Xgboost: a scalable tree boosting system", in Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (ACM, 2016), p. 785, https://dl.acm.org/doi/10.1145/2939672.2939785.

[12]S. Haykin, *Neural networks and learning machines* (Pearson Upper Saddle River, NJ, USA: 2009).

[13]I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning* (MIT press, 2016).

[14]S. Ruder, "An overview of gradient descent optimization algorithms", arXiv:1609.04747 (2016).

[15]T. Lin et al., "Simulation software of the JUNO experiment", The European Physical Journal C **83**, 10.1140/epjc/s10052-023-11514-x (2023), https://doi.org/10.1140/epjc/s10052-023-11514-x.

[16]S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions", in Proceedings of the 31st international conference on neural information processing systems (2017), p. 4765.