RPE Radiation Physics & Engineering
A Peer-Reviewed Journal
Published by K.N. Toosi University of Technology
http://rpe.kntu.ac.ir
Review Article

Radiation Physics and Engineering 2024; ?(?):?–?

Imbalanced classification of antineutrino and cosmic muon in segmented plastic scintillator antineutrino detector

Javad Karimi[a], Faezeh Rahmani[a,*], S. Bijan Jia[b]

[a] *Department of Physics, K.N. Toosi University of Technology, Tehran, Iran*
[b] *Department of Physics, University of Bojnord, Bojnord, Iran*

HIGHLIGHTS

- Inverse beta decay in IRan ANtineutrino Detector as segmented plastic scintillators for antineutrino detection.
- IRAND-Sim Simulation Package for spectra and angular distribution of antineutrinos and muons.
- Memory management techniques to handle the dataset due to the large number of muons.
- Two methods of imbalanced classification for discriminating muon and antineutrino events.

ABSTRACT

Inverse beta decay (IBD) in plastic scintillators is one of the most commonly used methods for detecting reactor antineutrinos. Cosmic muon signals due to the IBD compared to those generated by antineutrinos are still the main challenge in these types of detectors. The IRAND (IRan ANtineutrino Detector) is currently being designed and implemented with the constraint of reducing the required hardware, and at the same time, improving the antineutrino detection efficiency. Imbalanced classification is one of the software methods in machine learning that deals with imbalanced data, such as muon and antineutrino. Using the IRAND-Sim simulation package based on the Geant4 toolkit presented in our previous research, the spectra and angular distribution of antineutrinos and muons can be calculated. However, in this study, the memory management techniques to handle the dataset due to a large number of muons have been used, and also two separate methods have been used in the imbalanced classification for discriminating muon and antineutrino events. The results show that this approach by combining real and simulated data is very efficient, and the imbalanced nature can be reduced to achieve better classifier performance.

KEYWORDS

Segmented plastic scintillator
Antineutrino detector
Machine learning
Geant4 toolkit
Muon reduction
Imbalanced classification

HISTORY

Received: ?
Revised: ?
Accepted: ?
Published: ?

1 Introduction

The use of Inverse Beta Decay (IBD) interaction in plastic scintillator detectors is a commonly used method for identifying reactor antineutrinos. Antineutrino detectors (ADs) are normally comprised plastic scintillator components along with a neutron-capturing material such as gadolinium that can be installed in the vicinity of a reactor (Oguri et al., 2014; Netrakanti et al., 2022; Haghighat et al., 2020; Abreu et al., 2017; Coleman et al., 2019). The IBD reaction can be identified from both prompt and delayed signals, arising from positron and neutron captures, respectively. Since the AD is placed on the ground level, cosmic muons are intense background interference for the system due to their higher flux in comparison to reactor antineutrinos. In addition, some muons produce antineutrino-like signals (prompt and delayed) as well (Lima Jr et al., 2019). The significant inequality in the quantities of muons and antineutrinos causes numerous antineutrino events beundetected, meaning that there exists a high occurrence of false-negatives in the classification process.

Different techniques have been employed, such as machine learning, to categorize these events (Ozturk, 2020; Migliorini et al., 2020; Li et al., 2018; Delgado, 2020; Mulmule et al., 2020; Choma et al., 2018). The difference between the number of antineutrinos and muons makes the classification problem as the imbalanced type. In the imbalanced issue, the effectiveness of the classifier and evaluation of its performance are different from the normal problems. Also, other metrics, such as the confusion matrix, the F1 score, the accuracy, the precision, and the

*Corresponding author: frahmani@kntu.ac.ir

https://doi.org/

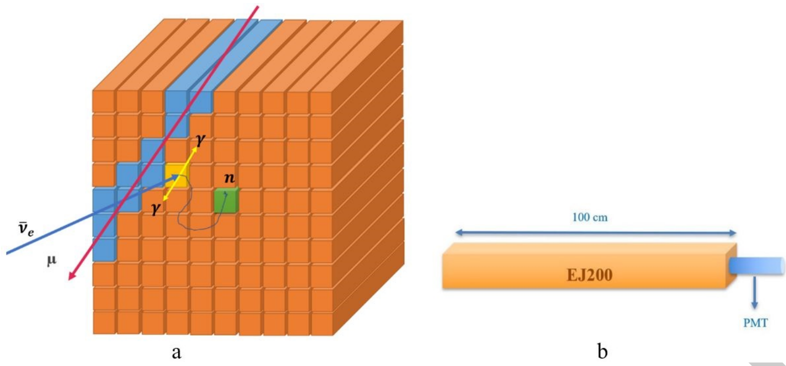


Figure 1: a) Schematic of IRAND detector (100 segments with dimension of $10 \times 10 \times 100$ cm) as well as muon and antineutrino incidence angle, b) A single module of plastic scintillator in IRAND detector along with a PMT (Karimi et al., 2023).

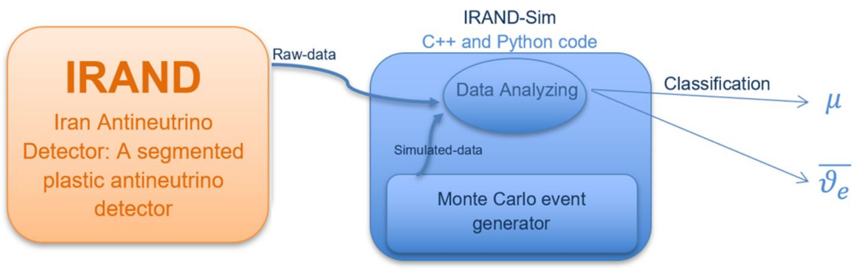


Figure 2: The process of data generation and utilization in the IRAND detector. The antineutrino character must be corrected. Now it is "v" not "nu".

recall needs to be incorporated to assess the success of the classifier.

The proposed technique of the imbalanced classification in machine learning for the efficient detection of antineutrinos can discriminate antineutrino events from muon ones with improved efficiency and minimal cost.

This new approach has been implemented in IRAND (IRan ANtineutrino Detector) and the results of our previous research (Karimi et al., 2023) (i.e., IRAND-Sim (IRan ANtineutrino Detector Simulation package) based on the Geant4 toolkit), has been used in the current research.

It should be noted that this proposed method, which has been employed for the first time, requires no special hardware-intensive setups.

2 Material and Methods

IRAND is the designed segmented plastic scintillator detector for detecting reactor antineutrinos, as shown in Fig. 1.

The detector design aims to reduce the hardware requirements as much as possible and to improve the detector performance through software methods. In our previous project, a method was proposed for storing and analyzing the data in the design where the half number of PMTs was eliminated such that only PMTs on one side of the detectors were used for data acquisition. Here, to enhance the system performance, machine learning techniques, such as imbalanced classification, have been employed, along with the utilization of additional simulated data. In this project, IRAND-sim developed in our previous work, has been used for data generation (Karimi et al., 2023). The process of data generation and utilization in the IRAND detector is depicted in Fig. 2.

2.1 *IRAND-Sim*

IRAND-Sim is a Monte Carlo simulation package based on the Geant4 toolkit that prepares data using Python programming. This package itself is also an event generator for the reactor antineutrinos, in which the most important events such as muon interactions and IBDs can be simulated. This developed code for the IRAND facilitates the simulation cosmic muons and antineutrinos. Furthermore, this package uses Python libraries such as Pandas (McKinney et al., 2011), Numpy (Idris, 2015), and Scipy (Virtanen et al., 2020) to conduct analysis and classification on both simulated and real data retrieved from the detector. More information can be found in (Karimi et al., 2023).

2.2 *Memory management*

The scintillation process in the antineutrino detector is also simulated in IRAND-Sim. According to the simulated scintillation yield in the plastic scintillator, about 10000 optical photons are generated for every 1 MeV deposition energy. All those optical photons are transported, and their information is stored, which requires lots of memory in the simulation process. The simulation process becomes more complex due to the substantial memory requirements, as observed in the case of G4OpticalPhysics (Agostinelli et al., 2003). To generate an appropriate set of events, it is imperative to effectively manage memory while executing the code.

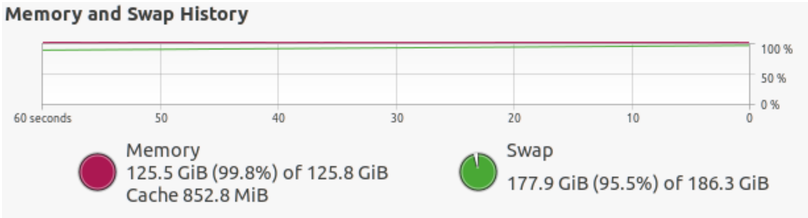


Figure 3: The memory usage during a single run of Monte Carlo sampling of 1000 positive and negative muons.

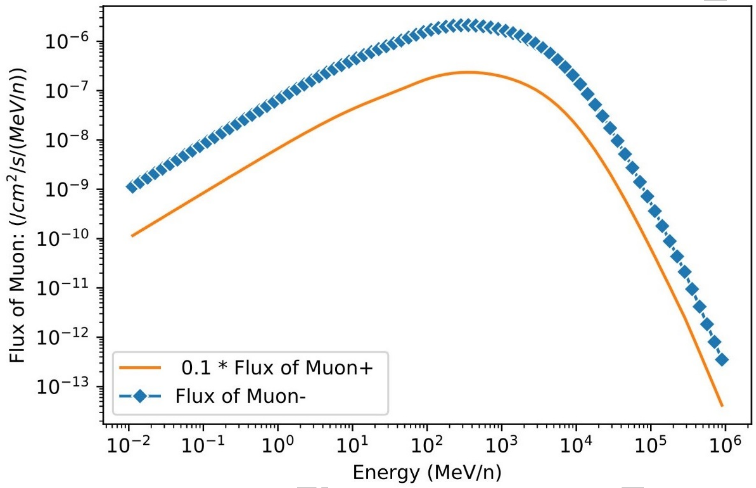


Figure 4: Positive and negative muon near sea level.

It should be noted that optical photons that can reach the photocathode are converted into photoelectrons, according to the quantum efficiency (QE) of the photocathode. So, some of them may produce no electron. When the photons reach the photocathode, the photoelectron production is sampled by the Monte Carlo from the probability density function (PDF) of the QE. If photoelectron production occurs, the related information is preserved in the memory, otherwise, omitted, and the photon history is terminated. This method saves approximately 75% of memory in comparison to the common tracking methods. The G4KillStep method in Geant4 has been used to terminate those unusable photons.

It might be useful to state that in our computing system (Fig. 3), for a computer run of cosmic muons with run/Beam On = 1000, around 300 gigabytes of memory are used, of which 128 gigabytes are supported by RAM, remaining by Swap.

One of the novelties in this work was memory management, which allowed us to generate a dataset with a real size and an acceptable volume. This approach facilitates the use of the resampling technique which will be discussed later.

2.3 EXPACS

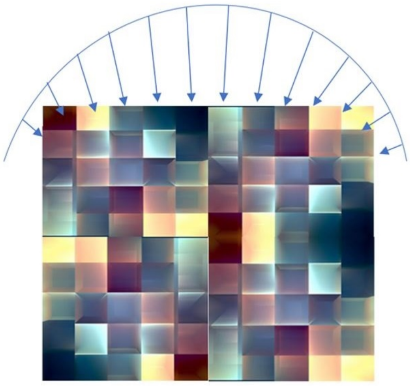
As mentioned before, the effects of a large flux of muons are significant in ADs, which cannot be ignored because of producing IBD-like signals in AD. For considering this effect, the energy spectrum of the PHITS-based Analytical Radiation Model in the Atmosphere (PARMA) has been used for simulating muons. PHITS is a Monte Carlo simulation code that can also be used for modeling cosmic-ray propagation in the atmosphere using the nuclear data library JENDL-High-Energy File (JENDL/HE) (Sato et al., 2008). The accuracy of the simulation has been successfully verified by experimental data taken under various conditions, even near sea level (Sato, 2015). EXPACS is the software for calculating atmospheric cosmic-ray spectra developed for the practical use of PARMA (Sato, 2018). The latest version of the software (Ver 4.13 released on May 3, 2023) has been used to obtain the spectrum of cosmic muons and to implement them as the primary particle in the simulation. Positive and negative muon fluxes have been obtained near sea level, as shown in Fig. 4.

Three primary particles (in two events) have been considered: positron-neutron (in IBD event) and muon events.

Here, particles are created in two different ways. IBDs (i.e., generating neutrons and positrons) are simulated by G4particleGun, and muons using G4GeneralParticleSource. A Messenger has been created to manage the production of particles. By using GeneratePrimaryVertex, two particles, positron and neutron, are created simultaneously with the same details (sampling according to the energy, angular distribution, and position PDFs) as mentioned in (Karimi et al., 2023). Each primary generator has a separate "run beam on".

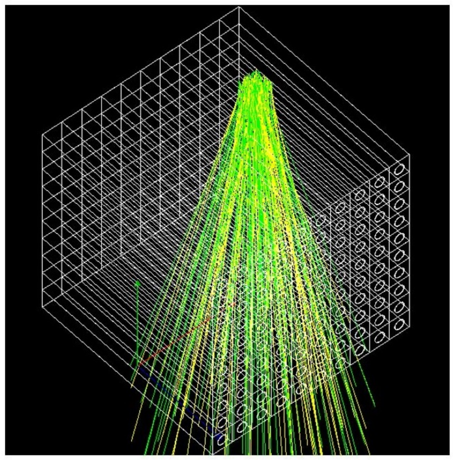
By setting "run beam on", the IBD and muon are sampled according to the relative rate, proportional to the IBD occurrence to muon interaction.

The specific problem for a 3 GWt nuclear power plant,



(a)

(b)



(c)

Figure 5: a) The polar angular distribution of muons incident on the detector. b) The isotropic distribution in the lateral direction. c) Simulation of the incident muons representing an angular symmetry.

e.g., Bushehr nuclear reactor, can be as follows: for a cubic-meter-sized plastic scintillator detector at a standoff of 30 m from a reactor core is expected to measure 2.3×10^3 events per day, and the number of hitting muons to the surface of an AD is about 10^6 per day. So, the sampling ratio of IBD to muon is 0.0023 (2.3×10^3 to 10^6) (Kuroda et al., 2012).

As shown in Fig. 5, the muons hit the detector from various directions, with symmetry in the lateral direction and an angular distribution in the polar direction.

Near the Bushehr reactor, the angular distribution of cosmic muons has been measured (Arneodo et al., 2019; Bahmanabadi, 2019) as shown in Fig. 6, and applied to create a more realistic data set for muons.

Furthermore, muons have been considered from all azimuthal angles to deactivate the impact of the detector positioning with respect to the reactor (regardless of its direction) during the learning process (Fig. 5-B).

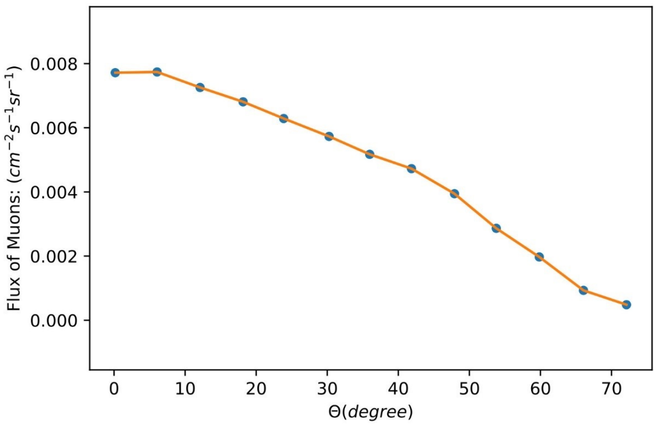


Figure 6: Angular distribution of cosmic muons near Bushehr (Arneodo et al., 2019).

2.4 Imbalanced problem

An imbalanced classification is a problem with an unequal number of samples in the training data set for each class label, due to the biased or skewed distribution of classes instead of equal or nearly equal ones. For example, in case of discriminating IBD events from muons in the AD, this imbalance in the number of events is clearly evident (Fig. 7).

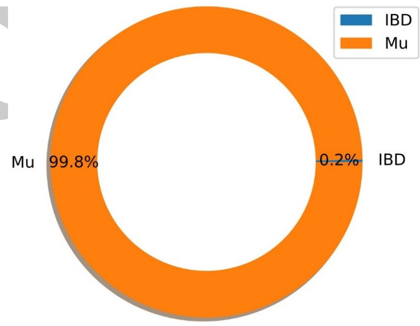


Figure 7: Imbalance in the number of IBD and muon (Mu) events.

2.5 Metrics

The machine learning models are evaluated using a variety of performance evaluation techniques. To assess the effectiveness of a strategy, it can be advantageous to combine various evaluation methods (Lee et al., 2021; Agarwal et al., 2023).Therefore, well-known metrics- accuracy, precision, recall -are employed in this study.

In addition, the confusion matrix is used to show true positive (TP), true negative (TN), false positive (FP), and false negative (FN) which are used to calculate the values for accuracy, precision, and recall. These metrics are calculated using the following equations:

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

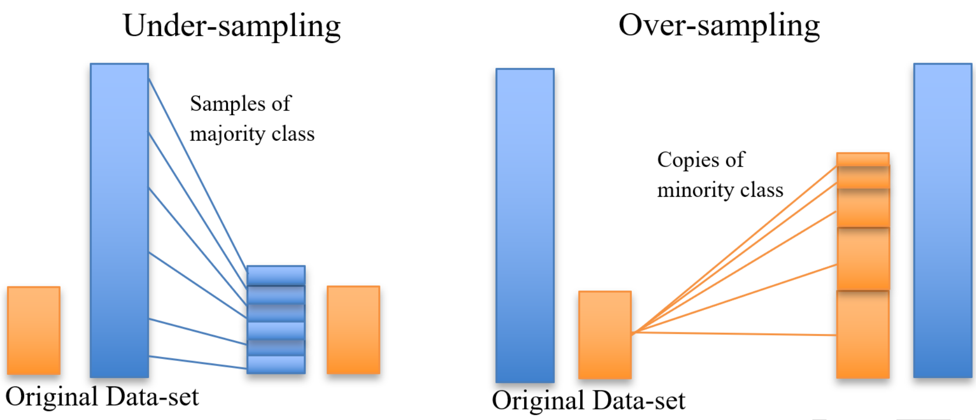


Figure 8: Under-sampling and Over-sampling methods.

$$\text{precision} = \frac{TP}{TP + FP} \tag{2}$$

$$\text{recall} = \frac{TP}{TP + FN} \tag{3}$$

Recall is the most important metric in this research due to the nature of the problem. For example, whether muons are counted or not, the antineutrinos counting is very important such that if they are not counted, the data will be regarded as lost.

2.6 Over-Sampling

Re-sampling is a commonly used method in excessively imbalanced data sets. It involves under-sampling (i.e., removing samples from the majority class) or over-sampling (i.e., increasing samples from the minority class) as shown in Fig. 8.

Although balancing classes have specific advantages, these two strategies also have some drawbacks. Duplicating random data from the minority class is the simplest way to perform over-sampling, however, this might lead to overfitting.

The simplest method of under-sampling includes random data selection from the majority class, which might result in information loss. Under-sampling is not an appropriate option because of data extreme imbalance, therefore, further over-sampling must be explored. There are various over-sampling approaches. One of them is Synthetic Minority Over-sampling Technique (SMOT) , which increases data. This approach is quite popular in structured data. Sometimes Generative Adversarial Network (GAN) are employed. IBD data can be generated independently and can be incorporated into the data set using IRAND-Sim. As a result, the dedicated Monte Carlo simulations of IBD events have been used in our over-sampling strategy.

2.7 Cost-sensitive training

Cost-sensitive training, also known as cost-sensitive learning or cost-sensitive classification, is a machine learning approach that takes into account the variable costs associated with different types of errors in classification problems. In many real-world applications, misclassifying instances into different classes can have different consequences and associated costs. Cost-sensitive training aims to optimize the model performance by considering these costs and minimizing the overall cost of classification errors. This section describes how a typical cost-sensitive training works:

- **Class Imbalance**: It is common in classification problems to have imbalanced class distributions, where one class significantly outnumbers the other(s). In such cases, misclassifying the minority class can be more costly than misclassifying the majority one.
- **Cost Matrix**: To account for these differing costs, a cost matrix is defined. This matrix assigns a cost value for each type of classification error. For example, it might specify a higher cost for false positives (i.e., misclassifying a negative instance as positive) or false negatives (i.e., misclassifying a positive instance as negative) depending on the specific application.
- **Modified Learning Algorithm**: Traditional machine learning algorithms aim to minimize classification error or maximize accuracy. In cost-sensitive learning, the objective function is modified to minimize the total misclassification cost, as defined by the cost matrix. This modification can involve adjusting decision thresholds or altering the training process of the algorithm.
- **Model Selection**: Model selection and hyperparameter tuning consider the cost matrix. Cross-validation techniques are used to choose the best model configuration that minimizes the expected cost of unseen data.

By explicitly accounting for the cost of classification errors, cost-sensitive training helps machine learning models to make the decisions that align with the practical considerations of the problem, ultimately leading to more effective and cost-efficient solutions (Pes and Lai, 2021).

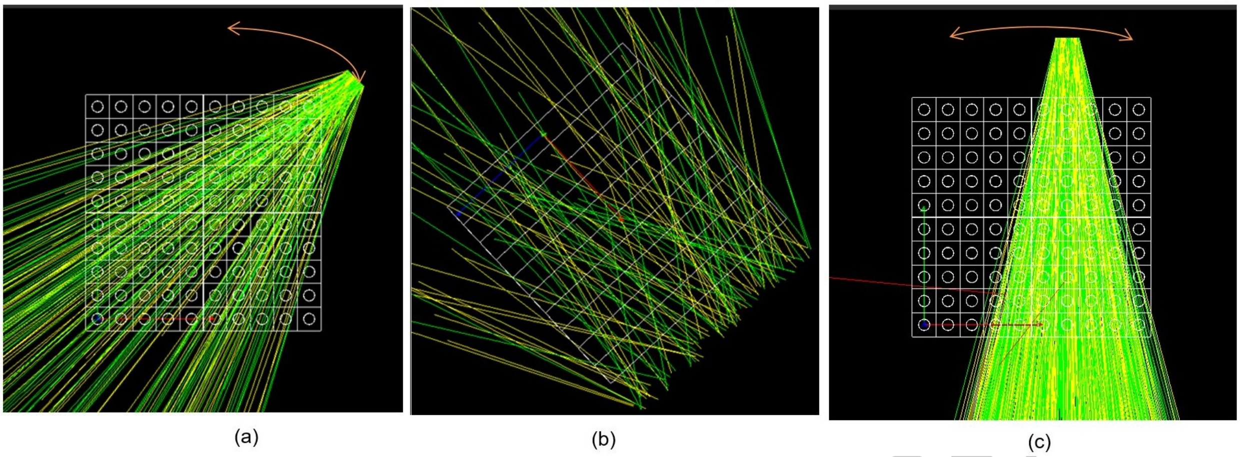


Figure 9: The muon and anti-muon incident on the detector exhibit a distinct angular pattern (as depicted in Fig. 6).

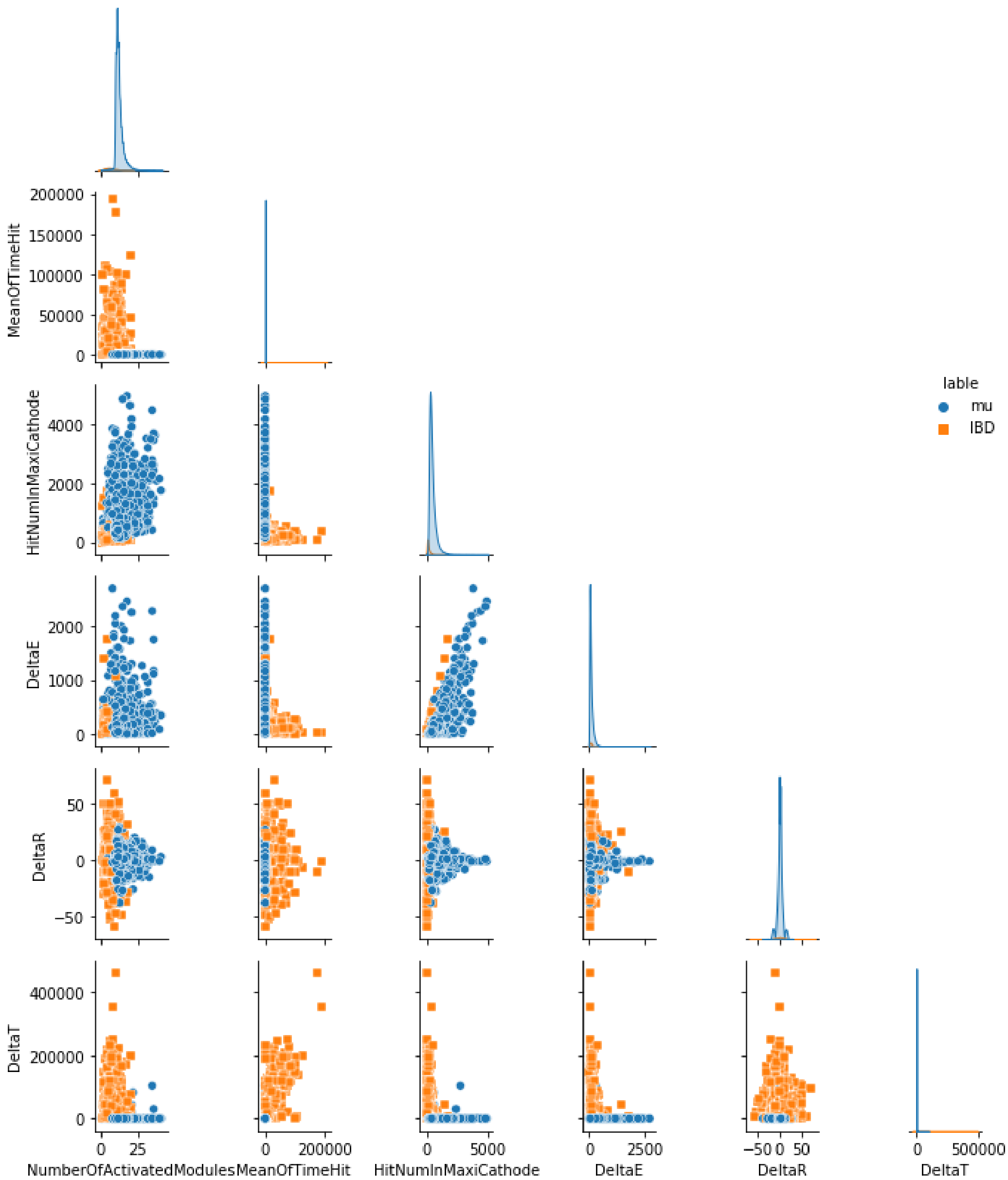


Figure 10: Correlation between the data in the prepared data set.

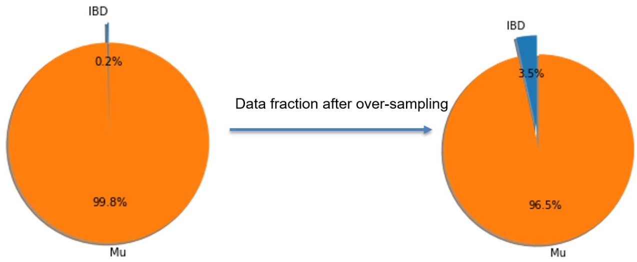
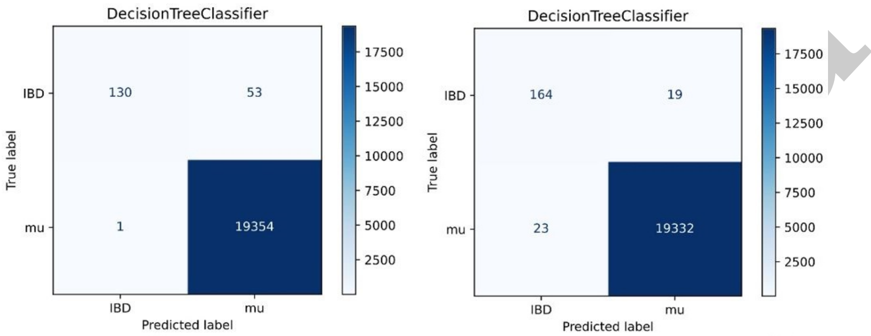


Figure 11: Data fraction before and after over-sampling.



(a) (b)

Figure 12: Confusion matrix (a) before and (b) after the over-sampling.

3 Results

One of the advantages of segmented AD is the ability to generate distinct topologies for each individual particle that enables the categorizing of events based on both the topological characteristics and the temporal patterns. As outlined in (Karimi et al., 2023), each event is meticulously documented using a set of six engineered attributes. Among these attributes, (1) The number of modules activated for each event (i.e., "NumberOfActivatedModules"), (2) The number of photons absorbed in the "First Maximum" (i.e., "HitNumInMaxiCathode"), (3) The difference between the first and Second Maximum" in the number of produced optical photons (i.e, "DeltaE"), and (4) The spatial difference between the First and Second Maximum (i.e, "DeltaR")) are intricately related to the topological attributes of events, while the remaining two (The mean of time hits for each event (i.e., "MeanOfTimeHit") and the time interval between the first and second peak of the signal (i.e., "DeltaT")) pertain to the temporal attributes of events. A thorough examination of their impact on the classification process led to the selection of these features as hyperparameters.

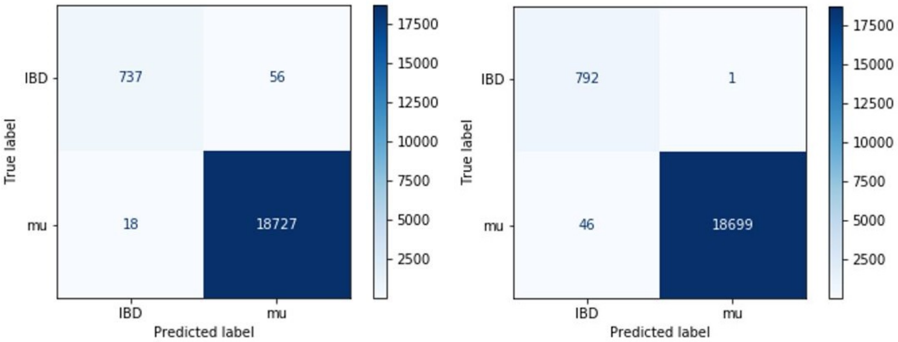
3.1 Preparing data set

A series of calculations have been conducted using IRAND-Sim by implementation memory management as described in the previous section. Due to the impracticality of extracting all data sets in a single run, multiple runs have been conducted, and then the results have been aggregated using Python. To accomplish this, a designed control panel has been employed to generate muons considering the energy and angular distribution of real muons in the detector. This panel scanned the entire detector area systematically and exposed the active volume to both positive and negative cosmic muons. For each run, as stated, the quantity of generated muons and their emission patterns have been set similarly to the angular distribution of real-world muons. About 70 distinct runs have been conducted. Some individual run samples are shown in Figs. 9-a to 9-c. This process is repeated for all angles to create the final data set in accordance with the expected distribution of muon.

The General Particle Source method has been employed to generate muons and fine-tune their angles and quantities. After preparation, two data were integrated to get the final data set, which was subsequently randomized (i.e., the data set with 113570 samples). As far as the reality is concerned, the relative distribution of the data in this data set is also imbalanced. Figure 10 illustrates the correlation between the data.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 256)               1792

 dense_1 (Dense)             (None, 256)               65792

 dropout (Dropout)           (None, 256)               0

 dense_2 (Dense)             (None, 256)               65792

 dropout_1 (Dropout)         (None, 256)               0

 dense_3 (Dense)             (None, 1)                 257

=================================================================
Total params: 133,633
Trainable params: 133,633
Non-trainable params: 0
_________________________________________________________________
```

Figure 13: Layer architecture in the Keras framework.



(a) (b)

Figure 14: Confusion matrix (a) without and (with) cost-sensitive training.

3.2 Over-Sampling

Over-sampling was achieved through a Monte Carlo simulation technique. The IBD events were replicated accurately using IRAND-Sim, thereby increasing the proportion of IBD events as shown in Fig. 11. In various studies (Ozturk, 2020; Mulmule et al., 2020; Karimi et al., 2023), the Random Forest classifier has been identified as one of the top-performing classifiers. In our study, the performance of the classifier in both standard and over-sampling modes was evaluated. It was evident that over-sampling enhanced the classifiers effectiveness (Fig. 12). Based on these findings, we can generate over-sampled data using the Monte Carlo method before linking them to the real data when the detector is positioned in proximity to the active reactor. This approach can be used to enhance overall performance of detector.

Table 1: The recall in different imbalanced classifications.

Classification method	Recall
DT without over-sampling	0.85
DT with over-sampling	0.94
ANN without cost-sensitive training	0.92
ANN with cost-sensitive training	0.99

3.3 Cost-sensitive training

In this proposed method, the Keras framework was used (Cholet, 2020). Neural networks with multiple layers were employed as the classifier, in which the layer architecture is shown in Fig. 13. The weights were inversely correlated with the number of events. Thrity Epochs were taken into account for training. With this strategy, it is possible to achieve the recall in the 30^{th} Epoch with 0.99. The confusion matrix is given for the two cases with/without the cost-sensitive training (Fig. 14).

As seen in Fig. 14, using the cost-sensitive training technique, more than 60 correct from 793 antineutrino events have been detected on the same data set, which is a very significant achievement. Table 1 summarizes the effect of different imbalanced classification methods on the recall metric.

4 Conclusions

In this study, a data set of muons and antineutrinos was created using IRAND-Sim Simulation Package, considering the particle information of the energy spectra and angular distribution, aiming to achieve the closest similarity to the reality. The imbalanced classification method used in this research can reduce the number of false-negative events, which is crucial in antineutrino detection. The cost-sensitive method shows a very good performance. One of the advantages of employing this approach is the ability to integrate the simulated data with the real one from the main IRAND detector. To do this, semi-supervised methods can be used with the distribution of labeled and unlabeled data, which is planned as a future project.

Statement

During the preparation of this manuscript, the authors used DALL-E2/Image Creator for generating Fig. 5-b.

Conflict of Interest

The authors declare no potential conflict of interest regarding the publication of this work.

References

Abreu, Y., Collaboration, S., et al. (2017). SoLid: An innovative anti-neutrino detector for searching oscillations at the SCK. CEN BR2 reactor. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 845:467–470.

Agarwal, C., Queen, O., Lakkaraju, H., et al. (2023). Evaluating explainability for graph neural networks. *Scientific Data*, 10(1):144.

Agostinelli, S., Allison, J., Amako, K. a., et al. (2003). GEANT4a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 506(3):250–303.

Arneodo, F., Benabderrahmane, M., Bruno, G., et al. (2019). Measurement of cosmic muons angular distribution in Abu Dhabi at sea level. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 936:242–243.

Bahmanabadi, M. (2019). A method for determining the angular distribution of atmospheric muons using a cosmic ray telescope. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 916:1–7.

Cholet, F. (2020). *Imbalanced classification (accessed on 2020/04/17)*. available at https://keras.io/examples/structured_data/imbalanced_classification/.

Choma, N., Monti, F., Gerhardt, L., et al. (2018). Graph neural networks for icecube signal classification. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 386–391. IEEE.

Coleman, J., Metelko, C., Murdoch, M., et al. (2019). VIDARR: Aboveground Reactor Monitoring. In *Journal of Physics: Conference Series*, volume 1216, page 012007. IOP Publishing.

Delgado, A. (2020). Machine Learning Applications for Reactor Antineutrino Detection at PROSPECT. In *APS Division of Nuclear Physics Meeting Abstracts*, volume 2020, pages EG–007.

Haghighat, A., Huber, P., Li, S., et al. (2020). Observation of reactor antineutrinos with a rapidly deployable surface-level detector. *Physical Review Applied*, 13(3):034028.

Idris, I. (2015). *NumPy: Beginner's Guide*. Packt Publishing Ltd.

Karimi, J., Rahmani, F., and Jia, S. B. (2023). Improving the detection efficiency of IRAND based on machine learning. *Computer Physics Communications*, 291:108833.

Kuroda, Y., Oguri, S., Kato, Y., et al. (2012). A mobile antineutrino detector with plastic scintillators. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 690:41–47.

Lee, E., Rustam, F., Aljedaani, W., et al. (2021). Predicting pulsars from imbalanced dataset with hybrid resampling approach. *Advances in Astronomy*, 2021:1–13.

Li, R., Zhengyun, Y., and Zhang, Y. (2018). Deep learning for signal and background discrimination in liquid based neutrino experiment. In *Journal of Physics: Conference Series*, volume 1085, page 042037. IOP Publishing.

Lima Jr, H., Alfonzo, J., Anjos, J., et al. (2019). Neutrinos Angra experiment: commissioning and first operational measurements. *Journal of Instrumentation*, 14(06):P06010.

McKinney, W. et al. (2011). pandas: a foundational Python library for data analysis and statistics. *Python for High Performance and Scientific Computing*, 14(9):1–9.

Migliorini, M. L., Cerqueira, A. S., Costa, I. A., et al. (2020). Identification and Classification of Corrupted Signals for the Neutrinos Angra Experiment. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 367–372. IEEE.

Mulmule, D., Netrakanti, P., Pant, L., et al. (2020). Machine learning technique to improve anti-neutrino detection efficiency for the ISMRAN experiment. *Journal of Instrumentation*, 15(04):P04021.

Netrakanti, P., Mulmule, D., Mishra, D., et al. (2022). Measurements using a prototype array of plastic scintillator bars for reactor based electron anti-neutrino detection. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 1024:166126.

Oguri, S., Kuroda, Y., Kato, Y., et al. (2014). Reactor antineutrino monitoring with a plastic scintillator array as a new safeguards method. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 757:33–39.

Ozturk, S. (2020). Nuclear reactor monitoring with gadolinium-loaded plastic scintillator modules. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 955:163314.

Pes, B. and Lai, G. (2021). Cost-sensitive learning strategies for high-dimensional and imbalanced data: a comparative study. *PeerJ Computer Science*, 7:e832.

Sato, T. (2015). Analytical model for estimating terrestrial cosmic ray fluxes nearly anytime and anywhere in the world: Extension of PARMA/EXPACS. *PloS one*, 10(12):e0144679.

Sato, T. (2018). *EXPACS: Excel-based Program for calculating Atmospheric Cosmic-ray Spectrum User's Manual (Last update Dec. 21, 2018)*. User's Manual.

Sato, T., Yasuda, H., Niita, K., et al. (2008). Development of PARMA: PHITS-based analytical radiation model in the atmosphere. *Radiation Research*, 170(2):244–259.

Virtanen, P., Gommers, R., Oliphant, T. E., et al. (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3):261–272.

To cite this article:

J. Karimi, F. Rahmani, S.B. Jia (2024). Imbalanced classification of antineutrino and cosmic muon in segmented plastic scintillator antineutrino detector. *Radiation Physics and Engineering*, In Press

DOI:

To link to this article: