Jurnal Ilmu Komputer dan Informasi (Journal of Computer Science and Information)
18/2 (2025), 181-205. DOI: http://dx.doi.org/10.21609/jiki.v18i2.1489

Application of Machine Learning Methods for Classification of Gamma and Hadron Signals in High Energy Particle Detection

Firdaus Andi Wibowo[1], Tomi Yulianto[1], Nicholaus Ola Malun, Rizqy Rionaldy[1], Verdi Yasin[2], Ruben Cornelius Siagian[3]

[1]Data Science Business Informatics, Swiss German University, Banten, Indonesia
[2]Department of Informatics Engineering, Jayakarta College of Information and Computer Management, Jakarta, Indonesia
[3]Faculty of Mathematics and Natural Sciences, Universitas Negeri Medan, Medan, Indonesia

E-mail: firdaus.wibowo@student.sgu.ac.id

Abstract

A major challenge in particle physics is the binary classification of high-energy gamma signals against a complex hadron background. Accurate identification of these gamma signals is critical for particle detection, especially as the volume and complexity of data increases as technology advances. The research developed a machine learning-based classification model to efficiently and accurately distinguish gamma signals from hadrons. Logistic Regression, Decision Trees, Random Forests, and Artificial Neural Networks are used for classification. Principal Component Analysis (PCA) and correlation analysis identified dominant features, while Monte Carlo simulations validated the distribution of gamma and hadron spectra. This study focuses on geometric parameters such as fLength, fWidth, fAlpha, as well as photon distribution and distance effects (fDist) in gamma signal identification using K-Means clustering. The Random Forest algorithm achieved the highest accuracy of 87.96%, with an F1-score of 0.91, which defines its robustness in the classification task. PCA and correlation analysis showed fSize, fLength, and fWidth as the most influential factors in classification. Monte Carlo simulations successfully replicated the spectral distribution pattern with high experimental validation. The research presents a novel integration of geometric analysis, clustering techniques, and simulation validation in the classification of high-energy particles. Machine learning methods, in particular Random Forest, effectively distinguish the gamma signal from the hadron background. The combination of PCA and Monte Carlo simulations improves the understanding of data distribution patterns and key classification factors. This research contributes to the development of a more reliable astrophysical signal classification system with potential applications in large-scale astronomical data management.

Keywords: *Gamma signal detection, particle physics classification, machine learning algorithms, monte carlo simulation, geometric parameter analysis*

1. Introduction

In particle physics, one of the main challenges is distinguishing between very high-energy gamma signals and the more complex and more numerous hadron background [1], [2]. This process is crucial for particle detection and identification in physics experiments, especially in particle telescopes and astrophysical observatories. Gamma, which is a high-energy photon, is often the signal to look for, but hadrons, which are made up of particles such as protons and neutrons, often produce a distracting background [3], [4], [5], [6]. With the development of detection and data acquisition technologies, such as particle telescopes and gamma-ray detectors, the data generated is becoming increasingly large and complex [7]. Identifying and separating the gamma signal from the hadron background requires efficient and accurate methods. One promising solution is the use of machine learning techniques to develop predictive models that can distinguish the two types of signals based on available features. Along with advances in data analysis and machine learning, various algorithms can be used to improve accuracy in gamma and hadron classification. Methods such as Logistic Regression, Decision Trees, Random Forests, and Neural Networks have proven effective in various classification applications. Evaluation of the model using metrics such as accuracy, precision, recall,

and AUC-ROC is important to ensure that the model built is reliable in real conditions [8], [9].

Research in this field has been conducted for a long time, with various approaches that continue to evolve along with advances in detection and data processing technology. Previously, gamma and hadron identification techniques relied more on classical statistical methods and simple algorithms. However, as data complexity increases, especially in large experiments such as those conducted at astrophysical and particle physics observatories, machine learning approaches become more relevant. One important aspect of this research is to understand the dominant factors that distinguish gamma and hadrons. Using principal component analysis (PCA) and correlation analysis techniques, it is possible to identify which features are most influential in distinguishing the two types of signals. This research will examine the relationship between geometric parameters and photon distribution in the telescope. It illustrates how these variables relate to the detected photon distribution. Visualizing the distribution will provide a clearer picture of the patterns that can be used to distinguish gamma from hadron.

Research conducted by [10] explored the classification of image and non-image parameters in atmospheric Cherenkov events, which also highlighted the role of particle characteristics in distinguishing gamma rays from the hadron background, while still relying on conventional statistical methods. Further advances introduced machine learning to improve the gamma/hadron separation. The research conducted [1] applies a tree-based classification method, while [11] Integrating a reinforced decision tree (BDT) to improve detection of faint gamma rays. [12] and [13] further utilizes deep learning for event classification with data from the CERN LHC and ultrahigh-energy cosmic rays, but falls short of analyzing key physical parameters that affect classification accuracy. Other research, such as [14] and [15], highlighted the importance of understanding hadronic and electromagnetic cascades and the need for accurate classification methods. However, the integration of principal component analysis (PCA) and correlation analysis in gamma/hadron classification is rarely explored. This research extends previous approaches by combining machine learning, PCA, and clustering to improve classification accuracy. By considering geometric parameters such as fLength, fWidth, and fAlpha, as well as Monte Carlo simulation validation, this study offers a comprehensive approach for gamma signal identification in high-energy astrophysics.

This research develops a machine learning-based model to distinguish gamma signals from hadron background using Logistic Regression, Decision Tree, Random Forest, and Neural Network. The model performance is evaluated by accuracy, precision, recall, and AUC-ROC metrics. Principal Component Analysis (PCA) and correlation analysis identified key features that affect classification, especially geometric parameters (fLength, fWidth, fAlpha) and photon distribution. Clustering techniques analyzed the impact of distance (fDist), while Monte Carlo simulations validated the spectrum distribution patterns against real data.

This study hypothesizes that machine learning models can achieve high accuracy in classification and that geometric parameters significantly affect gamma identification. PCA and correlation analysis are expected to reveal the dominant patterns that distinguish gamma from hadrons. Clustering methods, such as K-Means, explore the role of fDist in classification, and Monte Carlo simulations assess the influence of fSize, fAlpha, and fDist on spectrum formation.

This research is limited to the selected machine learning algorithm and focuses on feature-based classification, without considering external environmental factors. This research emphasizes on statistical analysis and visualization to understand the distribution of features. Although previous studies have explored gamma-hadron classification, many have relied on conventional methods with limited efficiency for large datasets. This research integrates machine learning with PCA and correlation analysis to improve classification accuracy and efficiency.

This research presents a novel approach to gamma and hadron classification by evaluating three machine learning models-Decision Tree, Random Forest, and Neural Network-based on geometric and statistical features. Its main uniqueness lies in the in-depth analysis of the contribution of features in classification, which improves accuracy and also identifies the dominant factors that influence model decisions. The innovative approach in the research includes a combination of Principal Component Analysis (PCA) and SHAP analysis methods to assess the role of features such as fLength, fWidth, and fSize in distinguishing gamma and hadron. This research integrates correlation analysis and clustering using the K-means method to explore the geometric distribution of photons as well as the patterns of interrelationships between variables. Another advantage is the simulation of gamma and hadron spectra using the Monte Carlo method, which compares the distribution of the main features with the original data.

2. Methods

2.1 Gamma or Hadron Prediction Data from NASA's Fermi Gamma-ray Space Telescope

The dataset used in this study was obtained from NASA's Fermi Gamma-ray Space Telescope, which aims to classify particles as gamma (signal) or hadron (background) [16]. This dataset is a public dataset that can be accessed by the scientific community for research and development of astrophysical data analysis methods.

The dataset contains 19,020 samples with 11 numerical attributes that interpret the geometric characteristics of the detected particles, such as size, shape, and intensity distribution. The main attributes include the length and width of the ellipse, the total pixel size, the ratio of the highest intensities, the distance from the center, and the angles and statistical moments that describe the particle pattern. Particle classes are divided into two categories, namely gamma (g) and hadron (h).

In the implementation of machine learning, this dataset can be broken down into two subsets, where the Training data is 80% of the total data, namely 15,216 samples, and the Test Data is 20% of the total data, namely 3,804 samples. From the initial exploration results, this dataset has two classes, namely gamma and hadron. To determine the balance of the dataset, it is necessary to analyze the class distribution by calculating the number of samples in each class. If the number of classes is unbalanced (More gamma samples than hadrons or vice versa), then imbalance handling methods such as undersampling, oversampling, or algorithmic methods such as weighted loss function in machine learning models are required.

2.2 Geometrical and Intensity Features for Gamma and Hadron Particle Classification

The dataset used for particle type prediction, i.e. gamma or hadron, is based on geometric features and intensity measured from images or particle detection data. In physics, the data is used to distinguish between two different types of particles, namely gamma (particles typically associated with electromagnetic radiation) and hadrons (particles composed of quarks, such as protons and neutrons) [17], [18].

The first features contained in the dataset are fLength and fWidth, which describe the length and width of the ellipse axis representing the energy or intensity distribution of the particle at the detector. In physics, it indicates how distributed the particle energy is in the detection space. fSize is the logarithm of the sum of the contents of all pixels, which gives information about the total energy or intensity detected in the area, measured in photons. The larger the fSize value, the greater the energy or number of photons detected.

The fConc and fConc1 features measure the ratio of the highest pixel intensity compared to the total intensity (fSize). This ratio provides information about the energy concentration at a particular point, which can help in distinguishing between particles that have a more concentrated energy distribution (such as gamma) and those that are more dispersed (such as hadrons). fAsym measures the distance between the highest pixel and the center of the ellipse, projected on the major axis. It can describe the asymmetry of the energy distribution, which can be different between gamma and hadrons.

The fM3Long and fM3Trans features measure the third moment along the major and minor axes, which provides information on the energy distribution in the major and minor directions. In physics, these moments give an idea of the "roundness" or "roughness" of the energy distribution, which can be used to distinguish between types of particles that have different energy distributions.

fAlpha is the angle between the major axis of the ellipse and the vector pointing to the origin, which indicates the orientation of the energy distribution in the detection space. fDist measures the distance between the center of the ellipse and the origin, which can indicate how far the particle is from the detection point or measurement center.

Class is a label that indicates the type of particle, i.e. gamma (signal) or hadron (background), which is the main goal of prediction in this dataset. By analyzing these geometric and intensity features, we can distinguish between these two types of particles, which have different energy distribution patterns.

2.3 Mathematical Model in Gamma Classification of Hadrons

2.3.1 Decision Tree Analysis

The decision tree divides the data based on certain features with threshold values [19], [20]. The feature `fAlpha` is used to separate the data into two classes, namely Gamma (g) and Hadron (h). If $f_\alpha <$ threshold, then the data is classified as Gamma (g). If $f_\alpha \geq$ threshold, then the data is classified as Hadron (h). Mathematically, this division can be written as:

$$\phi_j(f) = \sum_{S \subseteq N \setminus j} \frac{|S|!(|N|-|S|-1)!}{|N|!}[f(S \cup j) - f(S)] \qquad (1)$$

The probability of a class $P(\text{Class})$ is calculated based on the ratio of the number of observations in a particular class to the total observations in that node [21]. The formula is:

$$P(C) = \frac{n_C}{n} \quad (2)$$

Mathematically, the probability of a class at the final node is calculated as the ratio of the number of observations in that class to the total observations at the node [22]. Mathematically, the Gamma probability is:

$$P(\Gamma) = \frac{n_\Gamma}{n} \quad (3)$$

while the Hadron probability is:

$$P(\text{Hadron}) = \frac{n_H}{n} \quad (4)$$

In the final node, the class prediction is made based on the highest probability. The decision tree selects the feature with the highest information gain, which is calculated by the reduction in entropy before and after the split [23]. Entropy (H) is computed using the formula:

$$H = -\sum_k P_k \, log_2 \, P_k \quad (5)$$

where (P_k) is the probability of class (k) [24]. Decision tree models separate data based on a threshold value of a particular feature, such as f_α. The classification probability at the final node is calculated by the ratio of observations in a particular class to the total observations in the node [22]. The gain information is used to select the best feature for data division.

2.3.2 Random Forest Analysis

The ensemble function in Random Forest is used to combine predictions from many decision trees to produce a final prediction [25]. The ensemble function can be written as follows:

$$RF(x) = \frac{1}{N}\sum_{i=1}^{N} T_i \, (x) \quad (6)$$

In the Random Forest model, the final prediction RF(x) is the average of the predictions from each decision tree [26]. Each tree $T_i(x)$ provides a prediction, and the final prediction is the average of all (N) tree predictions. To assess the model's accuracy. Accuracy, which measures the proportion of correct predictions, and Kappa, which evaluates the agreement between observed and predicted values [27]:

$$Accuracy = \frac{TP+TN}{T_0} \quad (7)$$

In model accuracy evaluation, True Positives (TP) are the correctly classified positive data, and True Negatives (TN) are the correctly classified negative data [28], [29]. Total Observations refer to all data tested. The Kappa coefficient measures the agreement between predictions and actual data, adjusting for chance agreement [30]. The Kappa formula is:

$$\kappa = \frac{P_o - P_e}{1 - P_e} \quad (8)$$

P_o is the actual level of agreement between the model predictions and the actual data, calculated as the proportion of correct predictions. Whereas P_e is the randomly expected level of agreement, which is calculated based on the distribution of classes in the data. In other words, P_o measures how well the model works, while P_e indicates how likely the agreement is to occur by chance. P_o is calculated by the formula:

$$P_o = \frac{TP+TN}{T_0} \quad (9)$$

where (TP) is True Positives and (TN) is True Negatives. The P_e (expected agreement) is calculated using the formula:

$$P_e = \frac{(TP+FP)(TP+FN)+(TN+FN)(TN+FP)}{T_0} \quad (10)$$

In Random Forest, feature contribution to model accuracy is measured using two metrics: Mean Decrease Accuracy and Mean Decrease Gini [31]. Mean Decrease Accuracy quantifies a feature's impact on accuracy by evaluating how much accuracy decreases when the feature is shuffled [32]. Features with greater impact cause a larger decrease in accuracy:

$$\text{MDA} = \Delta \text{ Accuracy afters huffling} \quad (11)$$

where (Δ Accuracy) is the difference in model accuracy before and after shuffling a specific feature. The larger the Mean Decrease Accuracy value, the more important the feature is in the model [33]. Mean Decrease Gini measures the reduction in Gini impurity caused by a feature in the decision tree. Gini impurity is a measure of impurity at each node in the decision tree [34]. The larger the reduction in Gini impurity, the more important the feature is in making decisions at each node [35]. The formula is:

$$\text{MDG} = \Delta\text{Gini} \quad (12)$$

At each node, Gini impurity is calculated using the formula:

$$Gini = 1 - \sum_{i=1}^{C} p_i^2 \quad (13)$$

where p_i is the proportion of class (i) at that node and (C) is the number of classes.

2.3.3 Neural Network Analysis

In neural networks, forward propagation is used to calculate the output of the model based on the inputs and weights [36]. For each layer (l), the output ($a^{(l)}$) is calculated as follows:

$$a^{(l)} = f\left(W^{(l)} a^{(l-1)} + b^{(l)}\right) \quad (14)$$

In a neural network, the output of each layer ($a^{(l)}$) is calculated by multiplying the output of the previous layer ($a^{(l-1)}$) by the weights ($W^{(l)}$), adding a bias ($b^{(l)}$), and then applying an activation function (f). These activation functions, such as sigmoid or ReLU, provide non-linearity to the model, which allows the neural network to learn and model more complex relationships in the data [37]. In the first layer, the input data ($a^{(0)}$) is directly used as input for this calculation, while in subsequent layers, the output of the previous layer is used to generate the output of the next layer [38]. The loss function is used to measure how well the model predicts compared to the actual label. In the case of binary classification, the loss function used is log loss or binary cross-entropy:

$$L = -\frac{1}{N}\sum_{i=1}^{N}\{y_i \, log(\hat{y}_i) + (1 - y_i) \, log(1 - \hat{y}_i)\} \quad (15)$$

Log loss is a function used to measure the error between the probabilistic prediction of the model and the actual target value [39]. (N) refers to the number of samples in the dataset, (y_i) is the original label for the (i)th sample (0 or 1), and ($\hat{y}_i$) is the model prediction for that sample. Log loss calculates the difference between the prediction ($\hat{y}_i$) and the label (y_i), with the goal of minimizing the log loss value so that the model can produce more accurate predictions [40]. The smaller the log loss value, the better the model is at predicting the correct label, indicating more optimal model performance in classification. Backward propagation is used to update the weights and bias of the model based on the gradient of the loss function [41]. The gradient is calculated for each layer and used to update the weights ($W^{(l)}$) on each layer. The weight update is done with the formula:

$$W^{(l)} \leftarrow W^{(l)} - \eta \frac{\partial L}{\partial W^{(l)}} \quad (16)$$

where (η) is the learning rate, which controls how much the weights change at each update step. The gradient of $\left(\partial L/\partial W^{(l)}\right)$ is the partial derivative of the loss function (L) against the weights ($W^{(l)}$), which is calculated using the chain rule. The gradient $\partial L/\partial W^{(l)}$ shows how much the loss function changes with respect to the weights ($W^{(l)}$). The weight update is done by subtracting the calculated gradient, multiplied by the learning rate (η), to reduce the error in prediction. To calculate the gradient $\partial L/\partial W^{(l)}$, we use the chain rule in backpropagation. The gradient for each layer is calculated from the last layer towards the first layer. The last layer (output layer) is the gradient calculated based on the difference between the prediction and the actual label. Gradient at the output layer for log loss:

$$\frac{\partial L}{\partial a^{(L)}} = \hat{y} - y \quad (17)$$

where ($\hat{y}$) is the output prediction and (y) is the target label. The previous layer which is the Gradient is calculated for each previous layer using the gradient calculated for the layer after it, multiplying by the derivative of the activation function [42]. The gradient for the weights at the (l)th layer:

$$\frac{\partial L}{\partial W^{(l)}} = \frac{\partial L}{\partial a^{(l)}} \cdot \frac{\partial a^{(l)}}{\partial W^{(l)}} \quad (18)$$

where ($\partial a^{(l)}/\partial W^{(l)}$) is the derivative of the output of the lth layer against the weights ($W^{(l)}$), which is calculated based on the activation function used. The bias is also updated during backward propagation in the same way as the weights:

$$b^{(l)} \leftarrow b^{(l)} - \eta \frac{\partial L}{\partial b^{(l)}} \quad (19)$$

where $\left(\partial L/\partial b^{(l)}\right)$ is the gradient of the loss function against the bias at layer (l). The bias gradient is calculated in the same way as the weights, and the bias is updated by subtracting the calculated bias gradient, multiplied by the learning rate (η).

2.3.4 Analysis of the Curve ROC

The ROC (Receiver Operating Characteristic) curve evaluates classification model performance by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) [43]. TPR, or sensitivity, is the ratio of correctly predicted positives to the total number of positive data [44]. Mathematically, it is expressed as:

$$TPR(x) = \frac{TP(x)}{TP(x)+FN(x)} \quad (20)$$

On the ROC Curve, the False Positive Rate (FPR) is calculated by the formula

$$FPR(x) = \frac{FP(x)}{FP(x)+TN(x)} \quad (21)$$

FP(x) represents the number of False Positives, and TN(x) represents the number of True Negatives at a given threshold value (x). The False Positive Rate (FPR) measures how often the model incorrectly classifies negative data as positive at that threshold [45]. The Area Under the Curve (AUC) evaluates the model's ability to distinguish between positive and negative classes [46]. AUC is calculated by integrating the True Positive Rate (TPR) over the threshold values (x), which range from 0 to 1, representing the area under the ROC Curve [47]:

$$AUC = \int_0^1 TPR(x)dx \quad (22)$$

Each point on the ROC curve represents the True Positive Rate (TPR) at a specific decision threshold (x). By varying (x), the ROC curve shows the relationship between TPR and the False Positive Rate (FPR). The area under the ROC curve is the integral of TPR(x) over the interval ($x \in [0,1]$), reflecting the accumulated contribution of TPR(x) at each threshold:

$$\text{AUC} = \int_{\text{FPR}=0}^{\text{FPR}=1} \text{TPR}(\text{FPR})d\text{F PR} \quad (23)$$

This integral gives us the area of the area bounded by the ROC curve, which effectively gives the AUC.

2.3.5 Shapley Analysis

For each subset S that does not include feature j, we calculate two predictions. First, the model prediction using only the features in S, and second, the model prediction by adding feature j to the subset S. The difference between these two predictions, which is calculated as:

$$\Delta f(S) = f(S \cup j) - f(S) \quad (24)$$

describes how much influence the addition of feature j has on the model prediction results. Each subset S has a weight that depends on the size of the subset and the number of features in the total set N. These weights are calculated using factorials:

$$\frac{|S|!(|N|-|S|-1)!}{|N|!} \quad (25)$$

|S|! counts the number of possible sequences for feature subset S, (|N| - |S| - 1)! counts the sequences for features that are not in subset S as well as feature j being analyzed, and |N|! is the number of sequences for all features in set N. These three elements are used to calculate the contribution weight of each subset in the Shapley value assessment. After calculating the prediction differences and weights for each subset, the contributions of each subset are summed up in equation (1).

Shapley's formula incorporates two main elements: the prediction difference, which measures the change in model prediction when feature j is added, and the subset weight, which calculates the contribution of each subset based on the order of possible features [48], [49].

2.4 Correlation Analysis

As for the correlation between variables in the dataset, the Pearson correlation coefficient formula is used:

$$r_{xy} = \frac{\sum(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum(x_i-\bar{x})^2 \sum(y_i-\bar{y})^2}} \quad (26)$$

The correlation coefficient r_{xy} measures the relationship between two variables (x) and (y), calculated by comparing the difference of individual values (x_i) and (y_i) from the mean of each variable ($\bar{x}$) and ($\bar{y}$).

2.5 PCA Analysis

In PCA analysis, principal components are calculated using eigenvalue decomposition of the data covariance matrix [50]. The process will result in principal components that maximize the variance of the data in a particular direction. As for the Covariance Matrix (Σ):

$$\Sigma = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})\,(x_i - \bar{x})^T \quad (27)$$

In the analysis, (x_i) refers to the data vector for the i-th observation, while ($\bar{x}$) is a vector containing the average of all data. To find the principal components, for the eigenvalues (λ) and eigenvectors (v) of the covariance matrix (Σ):

$$\Sigma v = \lambda v \quad (28)$$

The eigenvalue (λ) indicates how much variance is explained by the principal component, while the eigenvector (v) describes the direction of the principal component. After calculating the

eigenvalues and eigenvectors, we can calculate the proportion of variance explained by each principal component with the formula:

$$\text{Varians Proporsion} = \frac{\lambda_i}{\sum_{i=1}^{k} \lambda_i} \quad (39)$$

The eigenvalue (λ_i) represents the variance contribution of the i-th principal component, while (k) is the number of principal components selected in the analysis. PCA is also often visualized with a two-dimensional plot using the first two principal components (PC1 and PC2), which can be described by the formula:

$$z_1 = v_1^T x \quad \text{and} \quad z_2 = v_2^T x \quad (30)$$

In PCA analysis, the scores for the first (z_1) and second (z_2) principal components are calculated by multiplying the standardized data (x) by the eigenvectors of each principal component (v_1) and (v_2). (x) is the standardized data.

PCA (Principal Component Analysis) in this study serves as a method for dimension reduction as well as feature analysis. As a dimension reduction technique, PCA identifies the principal components that capture most of the variability in the data [51]. In this study, PC1 to PC6 already cover more than 90% of the data variation, allowing for a reduction in the dataset dimension without losing too much important information. This process helps machine learning models such as Decision Tree, Random Forest, and Neural Network work more efficiently, especially in avoiding the curse of dimensionality that can lead to overfitting.

PCA also plays a role in feature analysis by revealing the contribution of variables to data variance. Features such as fLength, fWidth, and fSize have the greatest influence in PC1, while fAsym and fM3Long are more dominant in PC2, indicating different patterns in data characteristics. The results from this analysis are used for visualization, by mapping the data in a two-dimensional space based on PC1 and PC2 to observe whether there is a clear separation between the gamma and hadron classes.

2.6 Distributed Monte Carlo

Suppose (X) is a random variable representing a particular feature such as (fsize), (falpha), or(fdist), which we want to simulate. The probability distribution function (PDF) of (X) is a function that describes the probability of occurrence of a value within the range of (X) values. The PDF gives us information about how the values of (X) are spread along the sample space:

$$f_X(x) = P(X = x) \quad (31)$$

where $f_X(x)$ is the PDF of the random variable (X) and (x) is the possible values taken by (X). Monte Carlo simulation involves taking random samples from a known or estimated PDF [52]. The goal of simulation is to obtain the distribution of a random variable (X) by generating a set of simulated values, which approximates the desired distribution. The value of (X) is simulated based on the PDF $f_X(x)$. The first step in Monte Carlo simulation is to generate random samples $(x_1, x_2, \dots, x_n)$ that follow the probability distribution $f_X(x)$. These values are obtained using sampling techniques, such as Inverse Transform Sampling or Rejection Sampling.

The Inverse Transform Sampling method is used to simulate the distribution of a random variable (X). The first step is to determine the cumulative distribution function (CDF) $f_X(x)$, which is the integral of the PDF $f_X(x)$. This CDF describes the cumulative probability that (X) is less than or equal to the value (x). The second step is to take random samples $(u_1, u_2, \dots, u_n)$ from the uniform distribution (U(0, 1)). The third step is to transform those (u_i) values into simulated (x_i) values using the inverse function of the CDF, $F_X^{-1}(u_i)$. As a result, we obtain a series of values $(u_1, u_2, \dots, u_n)$ that follow the distribution $f_X(x)$. After sampling and transformation, a set of values $(X_{\text{simulation}} = \{x_1, x_2, \dots, x_n\})$can be obtained which is expected to have a distribution very similar to the original distribution. Mathematically, as:

$$X_{\text{simulation}} \sim \text{PDF}(X) \quad (32)$$

That is, the distribution of the simulated value ($X_{\text{simulation}}$) follows the same probability distribution as the PDF $f_X(x)$. To prove that the distribution of these simulated values is really close to the desired distribution, we can use the Monte Carlo Convergence Theorem. This theorem states that if we take many random samples (n) large enough, then the distribution of those samples will be close to the desired distribution. Mathematically, this theorem states that:

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} g(x_i) \approx \mathbb{E}[g(X)] \quad (33)$$

where g(x) is the measurement or observation function applied to the sample value (x_i), and ($\mathbb{E}[g(X)]$) is the expected value of that function based on the distribution $f_X(x)$.

3 Results and Discussion

3.5 Gamma Classification of Hadrons with Machine Learning Methods

3.5.1 Decision Tree Analysis

Research has classified data between gamma (signal) and hadron (background), a set of geometric and statistical features measured on objects in an image. Models are built using various features, such as the length and width of the ellipse axis, total pixel size, pixel concentration ratio, and various other parameters.

In the decision tree analysis, the model has separated the data into nodes based on certain threshold values of the features `fAlpha` and `fLength`. If the value of `fAlpha` is less than 20.8755, the data is more likely to be gamma, with a probability of 84.12%. However, when the value of `fAlpha` is greater than the threshold, the data is more likely to be hadron, especially if the value of `fLength` is smaller than 38.65345 or larger than 119.5457, leading to further classification decisions.

The prediction of the decision tree model showed that of the total data tested, 2,809 data were classified as gamma (g), while 994 data were classified as hadron (h). This indicates that the model has a tendency to identify more data as gamma, with a considerable ratio between gamma and hadron. The number of data classified as hadrons is also quite significant, which implies that the model manages to separate the two classes quite well in the context of prediction based on the given features

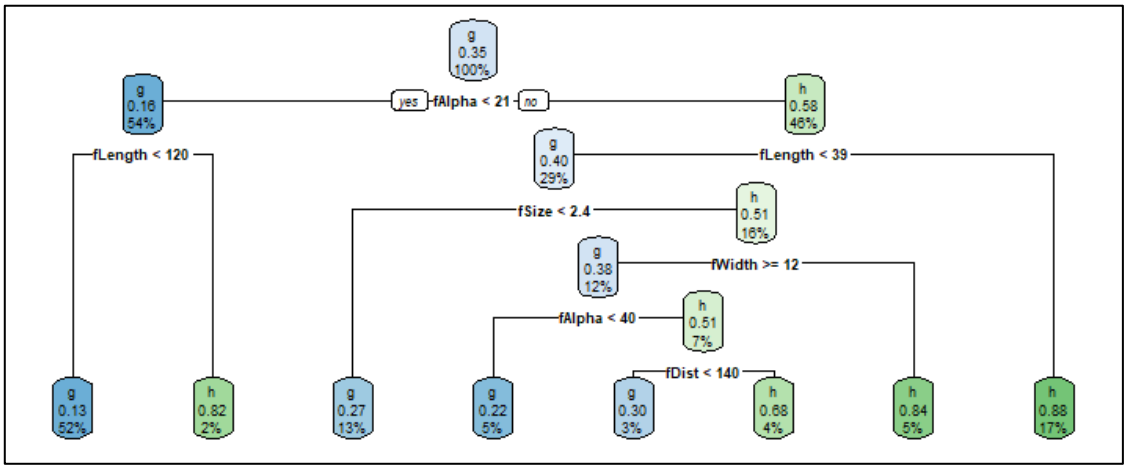

Figure 1. Classification model structure for separating data between Gamma (signal) and Hadron (background) classes.

Table 1. Key evaluation metrics for the Decision Tree model in classifying Gamma (g) and Hadron (h) data.

Metric	Value
Accuracy	**82.41%**
95% Confidence Interval	(81.16%, 83.61%)
No Information Rate	64.84%
P-Value [Acc > NIR]	**< 2.2e-16**
Kappa	0.5901
Mcnemar's Test P-Value	**< 2.2e-16**
Sensitivity (Recall)	**93.39%**
Specificity	62.15%
Positive Predictive Value (Precision)	81.99%
Negative Predictive Value	83.60%
Prevalence	64.84%
Detection Rate	60.56%
Detection Prevalence	73.86%
Balanced Accuracy	77.77%

Figure 1 is the structure of the classification model used to separate data between gamma (signal) and hadron (background) classes. Each node in the tree represents a decision based on the threshold value of a particular feature, such as `fAlpha`, `fLength`, and `fWidth`. The terminal node represents the final classification, with the probability of each class (gamma or hadron).

Shown in Table 1 are the main evaluation metrics for the decision tree model used in the classification between gamma (g) and hadron (h) data. With an accuracy of 82.41%, the model showed a good ability to classify the data, above the average guess. A Kappa value of 0.5901 indicates the model has moderate agreement between prediction and reality, after correcting for random agreement. The very high sensitivity metric (93.39%) indicates the model is very good at detecting the gamma (g) class, although the specificity for hadrons (h) is lower (62.15%). Positive predictive value (81.99%) indicates that when the model predicts gamma, it is mostly correct, while negative predictive value (83.60%) indicates that the model is quite accurate in predicting hadrons as hadrons. Balanced accuracy of 77.77% indicates that the model can classify both classes fairly equally, although there is a slight imbalance in the amount of data (prevalence

64.84%). The very small P-value indicates that the model is statistically better than a random guess.

Based on the evaluation results of the confusion matrix of the decision tree model, it can be seen that the model has a fairly good overall performance. The model successfully classified 2,303 data as gamma correctly, and 831 data as hadron correctly as well. However, there are also a number of misclassifications, namely 506 hadron data incorrectly classified as gamma, and 163 gamma data classified as hadron. The accuracy of the model reached 82.41%, which means that more than 8 out of 10 predictions of the model match the actual labels. This value is quite high and is supported by a very small p-value (<2.2e-16), which means that the performance of the model is statistically much better than a random guess based on the data distribution (No Information Rate). The sensitivity of the model, which describes the ability of the model to correctly recognize the gamma class, is very high at 93.39%. This means that most of the gamma data is successfully detected correctly. In contrast, the specificity, which shows the ability of the model to recognize hadron data, is 62.15%, which although lower, still shows that the model is able to distinguish between the two classes quite well. The Kappa value of 0.5901 indicates a moderate level of agreement between the model predictions and the actual labels, after correcting for possible agreement due to chance. This indicates that the model has a fairly reliable performance, although it could still be improved, especially in distinguishing the hadron class.

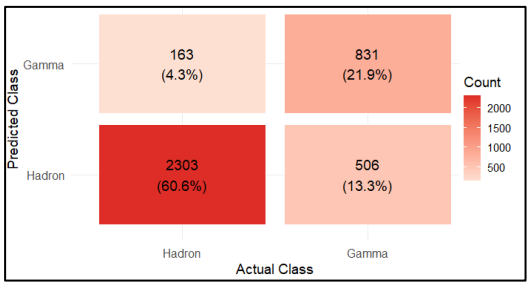

Figure 2. Confusion matrix of Gamma vs Hadron classification using Decision Tree (rpart).

It can be noticed in Figure 2, which is the resulting confusion matrix, that the decision tree model for the classification of gamma (g) and hadron (h) has a clear prediction distribution between the two classes. On the X-axis, which represents the actual (reference) class, there are two categories, including Gamma (g) and Hadron (h). Meanwhile, the Y-axis, which represents the classes predicted by the model, also includes the same two categories. This matrix shows that the model correctly predicted 2303 data as gamma (g) that were actually gamma, and 831 data as hadron (h) that were actually hadron. However, there were misclassifications, where 506 hadron (h) data were mispredicted as gamma, and 163 gamma (g) data were mispredicted as hadron.

3.5.2 Random Forest Analysis

Based on the confusion matrix, the model successfully classified 2326 gamma data correctly, but there were 318 gamma data that were misclassified as hadrons. The model successfully classified 1019 hadron data correctly, although there were 140 hadron data that were misclassified as gamma. The model has good accuracy in predicting both classes, although there are some misclassifications that need to be noted for further improvement.

The prediction model for gamma or hadrons has an accuracy of 87.96%, with a 95% confidence interval between 86.88% and 88.97%. The "No Information Rate" (NIR) value used as a reference is 64.84%, and the p-value for the comparison of the model accuracy against the NIR shows highly significant results, with a p-value < 2.2e-16. The Kappa coefficient of 0.7276 indicates a good level of agreement between model predictions and actual data. The McNemar test also showed significant results, with a very small p-value (< 2.2e-16), indicating that the model provides better predictions than random predictions.

Table 2. Gamma or hadron prediction model evaluation results.

Metric	Value
Accuracy	**87.96%**
95% Confidence Interval (CI)	(86.88%, 88.97%)
No Information Rate (NIR)	64.84%
P-Value [Acc > NIR]	**< 2.2e-16**
Kappa	0.7276
McNemar's Test P-Value	**< 2.2e-16**

Table 2 is the evaluation result of the gamma or hadron prediction model, where the model shows significant performance compared to the reference value (NIR) and has a Kappa coefficient of 0.7276, indicating good agreement. The McNemar test results also support the significance of the model being better than random prediction.

The prediction model for 'Positive' class detection (g) has an excellent sensitivity of 0.9432, which means that the model can correctly identify most positive cases. The specificity of the model reached 0.7622, indicating its fairly good ability to identify negative cases. The positive predictive value (Pos Pred Value) and negative predictive value (Neg Pred Value) were 0.8797 and 0.8792 respectively, indicating that the model was quite reliable in predicting both classes. With a prevalence of 0.6484, the model successfully detected about 61.16% of the actual positive cases (Detection Rate), while the detection prevalence

rate reached 69.52%. The balanced accuracy of the model is 0.8527, which confirms that the model performs quite well in distinguishing between positive and negative classes.

Based on the Feature Importance analysis, some features have a significant contribution to model accuracy. Among the most influential features is `fAlpha`, with a MeanDecreaseAccuracy value of 148.01141 and a very high MeanDecreaseGini of 1663.8876, meaning that the feature has a large impact in determining the prediction results. Other features that also show important contributions are `fSize`, with a MeanDecreaseAccuracy of 60.70807 and a MeanDecreaseGini of 784.2242, and `fWidth` which has a MeanDecreaseAccuracy of 52.70045 and a MeanDecreaseGini of 890.3852.

Features such as `fLength`, `fDist`, and `fM3Long` show an important role in the model, but with a slightly lower contribution rate compared to `fAlpha` and `fSize`. In contrast, features such as `fConc` and `fConc1` have a smaller contribution to the accuracy of the model, with lower MeanDecreaseAccuracy and less significant MeanDecreaseGini values.

Table 3. Importance of each feature in enhancing performance in the Random Forest model (MDA and MDG values).

Feature	MDA	MDG
fAlpha	**148.0114**	**1663.888**
fSize	60.70807	784.2242
fWidth	52.70045	890.3852
fDist	46.17297	465.3962
fLength	44.73865	1034.56
fM3Long	39.25734	628.0802
fAsym	27.73382	369.198
fConc	30.05116	399.0421
fM3Trans	26.78662	314.8018
fConc1	26.27733	382.7869

Table 3 shows the contribution of each feature in improving accuracy and its importance in the Random Forest model, with higher MeanDecreaseAccuracy values showing greater influence on the prediction result

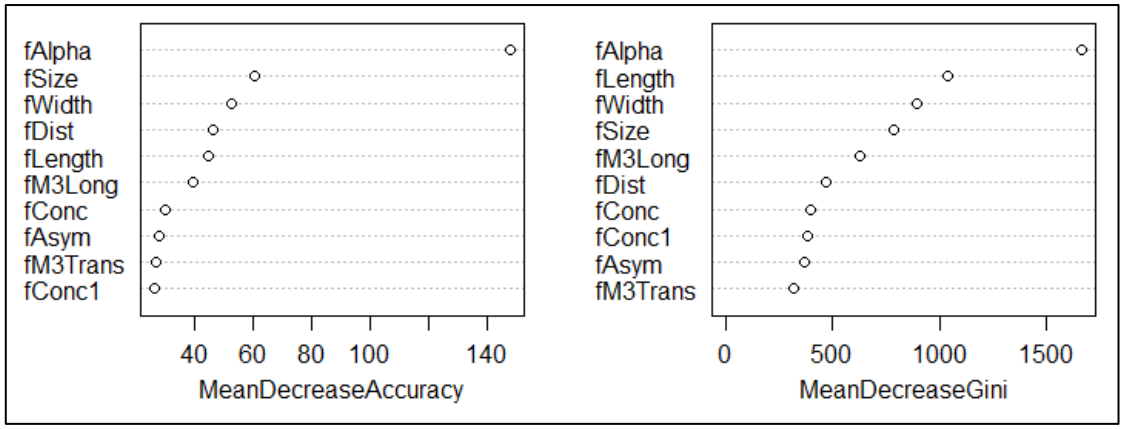

Figure 3. Features that have the most influence on prediction results.

Figure 3 displays the features that have the most influence on the prediction results, such as fAlpha, fSize, and fWidth, which appear to have higher importance compared to other features. The graph shows that fAlpha has the greatest influence, followed by fSize and fWidth, which each show significant contributions to the model's accuracy. In contrast, features such as fConc and fConc1 have lower importance values, indicating that their influence on gamma or hadron prediction is relatively small.

Shapley's interpretation method was used to analyze a prediction model built using random forest (RF) on data consisting of 15,217 rows and 10 columns. The model was used to predict classes based on various features measured in the dataset. Predictions on the test data showed two main results: a prediction value of 1.000000 with a difference of 0.351646 from the average prediction (0.648354), and a prediction value of 0.000000 with a difference of -0.351646 from the average prediction. Shapley analysis revealed the contribution of each feature to the prediction. Features such as length (fLength), width (fWidth), size (fSize), and concentration (fConc) have a significant influence on the predicted class. As seen, the fLength feature shows a positive contribution to the 'g' class with a value of 0.02, fWidth shows a negative contribution with a value of -0.13.

Table 4. The contribution of each feature in improving accuracy and its importance in the Random Forest model.

Feature	Class	Phi	Phi.var	Feature Value
fLength	g	0.02	0.080404	23.8172
fWidth	g	-0.13	0.235455	9.5728
fSize	g	**0.39**	**0.280707**	2.3385
fConc	g	-0.08	0.134949	0.6147
fConc1	g	-0.01	0.030202	0.3922
fAsym	g	-0.02	0.04	27.2107

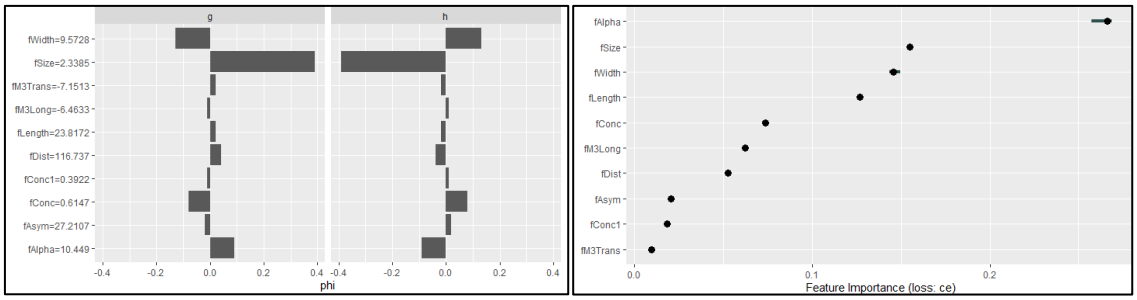

Figure 4. Shapley analysis and its important features.

Table 4 shows the contribution of each feature to the prediction of class 'g' using the Shapley value, with positive or negative values reflecting its influence on the prediction results. The variance of the Shapley value is also included to show the extent to which the contribution of the feature can vary.

Figure 4 is an interpretation of the model using two types of images resulting from Shapley analysis and feature importance. The left image depicts the contribution of each feature to the model's prediction, as measured by the Shapley value, where it can be seen how each feature influences the prediction outcome, with positive or negative values indicating an influence that leads to the prediction of a particular class. As features with higher Shapley values imply greater influence in influencing the final prediction. The right image is the result of the feature importance analysis, which is calculated using the cross-entropy (ce) loss metric. The features with the highest values in this figure are the most influential to the model's predictions, helping in understanding which features make the most significant contribution to the model's decisions.

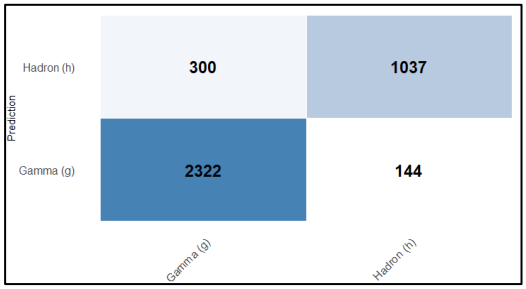

Figure 5. The confusion matrix obtained from the model illustrates the performance of the Random Forest model.

Figure 5 shows the Confusion matrix obtained from the performance of the Random Forest model in classifying data based on two classes, namely Gamma (g) and Hadron (h). The matrix consists of four main elements that show how the model predicts each class. In the first part, it can be seen that there are 2322 Gamma cases that were correctly predicted as Gamma (g). This shows that the model has a very good ability in identifying Gamma data. However, there are 300 Gamma cases that were incorrectly predicted as Hadron (h), which indicates a misclassification, but the number is relatively small compared to the total number of Gamma cases. On the other hand, in the second row, there are 144 cases of Hadrons mispredicted as Gamma (g), indicating a misclassification from Hadrons to Gamma. While this is significant, it is much smaller than the number of errors in the Gamma(g) category. However, there were 1037 Hadron cases that were correctly predicted as Hadron (h), indicating that the model was also quite effective in identifying the Hadron data.

3.5.3 Neural Network Analysis

The neural network model was trained using training data with a hidden layer size of 10 and a maximum number of iterations of 200. After the model was applied to the test data, predictions were made to classify the data into the two categories. Evaluation results using its confusion matrix showed that the model produced 348 correct predictions for the gamma (g) category and 392 correct predictions for the hadron (h) category. However, there were 945 misclassifications for gamma predicted as hadron, and 2118 misclassifications for hadron predicted as gamma. The neural network model can perform the classification, but there are challenges in prediction accuracy for both categories.

The prediction model for gamma or hadron classification has an accuracy rate of 0.1946, with a 95% confidence interval between 0.1821 and 0.2075. The value performs poorly compared to the zero information rate (NIR) accuracy of 0.6484. The p-value for accuracies greater than NIR is 1, indicating the model does not provide statistically significant results. Kappa, which measures the agreement between predictions and observations, has a negative value of -0.4711, indicating that the model is not even better than a random guess. The McNemar test showed highly significant results with a p-value of less than 2e-16, indicating a significant difference between the predicted and actual values. The sensitivity of the model is 0.14112, which shows that the ability of the model

to detect the positive class (gamma or hadron) is quite low. The specificity of the model is 0.29319, which indicates its ability to detect the negative class is also low. The Pos Pred Value (PPV) was 0.26914, showing that only about 26.9% of the positive predictions were actually positive. The Neg Pred Value (NPV) was 0.15618, showing that only 15.6% of the negative predictions were actually negative. The prevalence of the positive class (gamma or hadron) is 0.64844, which shows that this class appears more frequently in the dataset. Detection Rate was 0.09151, indicating that only 9.15% of the positive events were successfully detected by the model. Detection Prevalence is 0.33999, which shows that 33.99% of the model predictions are positive classes. Balanced Accuracy, which considers both sensitivity and specificity, is 0.21716, indicating that the model has a very limited ability to distinguish between positive and negative classes.

Figure 6 is the resulting confusion matrix showing the frequency distribution between the predicted and actual classes in the neural network model. Darker colors on the heatmap represent higher predicted frequencies, lighter colors represent lower frequencies.

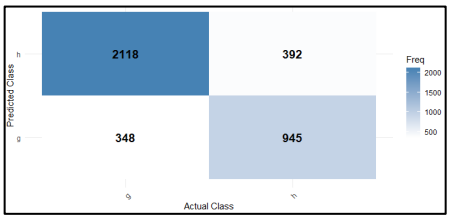

Fig 6. The resulting confusion matrix shows the frequency distribution between the predicted and actual classes in the neural network model.

3.5.4 Comparative Evaluation of Machine Learning Methods in Gamma and Hadron Particle Classification

We have evaluated and compared the performance of three machine learning methods-Decision Tree, Random Forest, and Neural Network-in classifying two types of particles, namely gamma particles and hadron particles, based on features available in the dataset. The main focus of the analysis is to see how effective each model is in distinguishing between the two types of particles, through evaluation metrics that include accuracy, sensitivity, specificity, positive and negative predictive values, and balanced accuracy. Table 5 shows a comparison of the evaluation metrics for the three gamma vs hadron classification methods, using Decision Tree, Random Forest, and Neural Network models. The Table 5 an equal and comprehensive evaluation based on key performance metrics.

Table 5. Comparison of Gamma vs Hadron classification model evaluation metrics.

Metric	Decision Tree	Random Forest	Neural Network
Accuracy	82.41%	**87.96%**	19.46%
95% Confidence Interval	(81.16%, 83.61%)	**(86.88%, 88.97%)**	(18.21%, 20.75%)
No Information Rate (NIR)	64.84%	64.84%	64.84%
P-Value [Acc > NIR]	< 2.2e-16	< 2.2e-16	1
Kappa	0.5901	**0.7276**	-0.4711
McNemar's Test P-Value	< 2.2e-16	< 2.2e-16	< 2e-16
Sensitivity (Recall)	93.39%	**94.32%**	14.11%
Specificity	62.15%	**76.22%**	29.32%
Precision (PPV)	81.99%	**87.97%**	26.91%
Negative Predictive Value (NPV)	83.60%	**87.92%**	15.62%
Prevalence	64.84%	64.84%	64.84%
Detection Rate	60.56%	**61.16%**	9.15%
Detection Prevalence	**73.86%**	69.52%	33.99%
Balanced Accuracy	77.77%	**85.27%**	21.72%

The evaluation results showed that Random Forest was the best performing model. The model achieved an accuracy of 87.96%, with a 95% confidence interval between 86.88% and 88.97%, and a Kappa value of 0.7276, indicating a substantial level of agreement between the model's predictions and the actual labels. This performance far exceeds the random value indicated by the No Information Rate (NIR) of 64.84%, with a highly significant p value ($p < 2.2e-16$), indicating that the results obtained are statistically better than random guesses. In terms of the ability to detect gamma particles, Random Forest shows a sensitivity of 94.32% and a specificity of 76.22%, which means that this model is able to recognize gamma particles very well while being relatively good at avoiding misclassifying hadron particles as gamma. Precision (PPV) of 87.97% and NPV of 87.92% indicate that the model's predictions are highly reliable. The Decision Tree model, while not performing as well as Random Forest, still performed reasonably well with an accuracy of 82.41%, and a Kappa value of 0.5901, indicating a medium level of agreement. The sensitivity of this model to gamma particles is also high, at 93.39%, but the specificity is only 62.15%, indicating that

the model is more likely to classify the data as gamma even though it is actually hadron. This is also reflected in the Detection Prevalence value which is quite high (73.86%), indicating that the model tends to overpredict gamma. However, the balanced accuracy is still relatively good, at 77.77%. In contrast, the Neural Network performed very poorly in this context, with an accuracy of only 19.46%, far below the NIR of 64.84%, and a negative Kappa value (-0.4711), indicating that the model's prediction was even worse than a random guess. The model is only able to detect about 14.11% of the actual gamma particles (sensitivity), with a specificity of 29.32%, and very low predictive values, both PPV (26.91%) and NPV (15.62%). Its balanced accuracy of only 21.72% reinforces the conclusion that this model fails to perform a decent classification.

3.5.5 Model comparison with ROC Curve

Figure 7 shows a comparison of the performance of three prediction models-Decision Tree, Random Forest, and Neural Network-in classifying data related to Gamma or Hadron predictions. The analysis is performed with the ROC (Receiver Operating Characteristic) curve for each model, which measures the model's ability to distinguish between the control class and the case class. The Decision Tree model produces an ROC curve depicted in red, Random Forest and Neural Network are represented in blue and green, respectively.

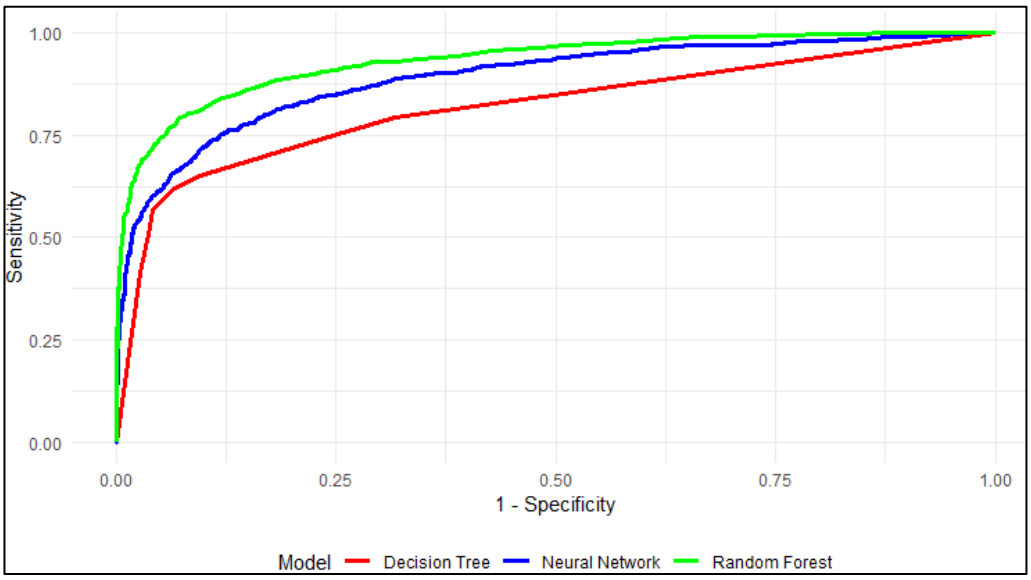

Figure 7. Performance comparison of three machine learning prediction models.

The three models show comparable performance in classifying the data, with the area under the curve (AUC) providing further insight into the accuracy of each model. From the ROC visualization results, it can be seen that each model has different characteristics in terms of sensitivity and specificity. Decision Tree tends to be more sensitive to the case class, whereas Random Forest and Neural Network provide a better balance between the two classes.

The ROC curve in Figure 7 on the right with the X-axis representing 1 - specificity, ensures that the range is between 0 and 1 with the origin at (0,0). This improvement allows for a more intuitive interpretation, as the further to the right of the X-axis, the greater the false positive rate, while the Y-axis still represents the sensitivity or true positive rate. With this view, the performance comparison of the Decision Tree, Random Forest, and Neural Network models becomes clearer. Models with curves closer to the upper left corner show better performance in distinguishing classes, with an optimal balance between sensitivity and specificity.

The model used 10 features, with a maximum decision tree depth of 6 and a learning rate (eta) of 0.3. The training process was conducted for 100 iterations, with the aim of optimizing the prediction using the binary logistic function. During the training phase, the measured log loss value showed a significant decrease from 0.5526 in the first iteration to 0.1196 in the last iteration, indicating the model successfully learned the patterns in the data. Feature contribution analysis using the SHAP method shows the influence of each feature on model prediction. The SHAP results show which features have the greatest contribution to the model's decision, which can be used for further interpretation.

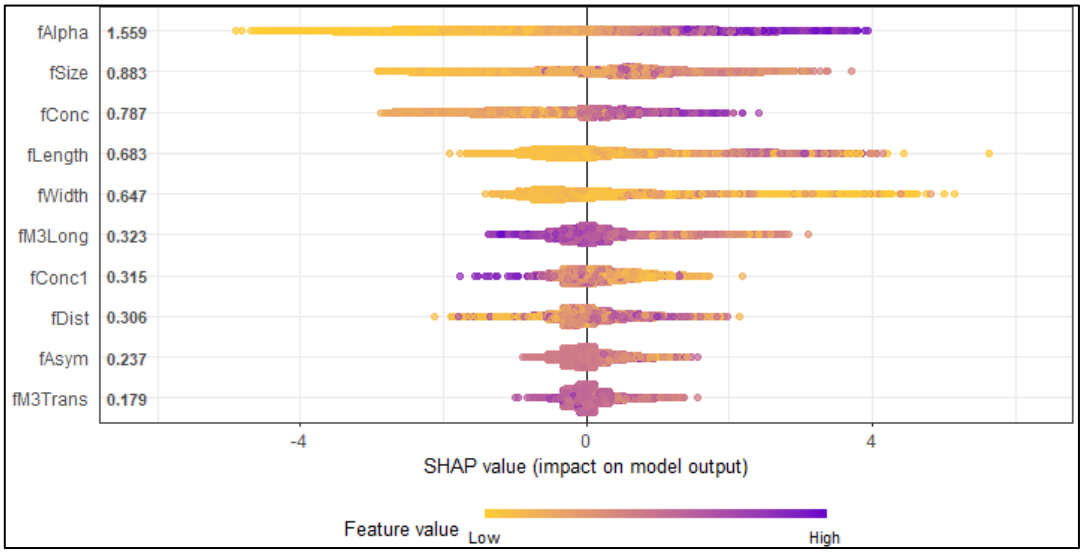

Figure 8. Contribution of a feature to model decisions using the SHAP method.

In Figure 8, each point shows the contribution of a feature to the model's decision, with the color indicating the value of the feature (red for high values and blue for low values). Features that have a greater contribution to the model prediction will appear more clearly on the left or right side of the axis, depending on whether their contribution is positive or negative to the predicted class. From the graph, it is possible to identify the most influential features in determining the prediction results. if a feature appears frequently with a large contribution, this means that it has a significant influence on the model. Conversely, features with small contributions or scattered around the zero axis can be considered less important in this prediction model.

3.6 Dominant Factor Analysis for Gamma Identification

3.6.1 Correlation Analysis

There are significant relationships between variables in the "telescope" dataset. The correlation matrix shows that some features have strong positive and negative correlations. fLength has significant positive correlations with fWidth (0.77) and fSize (0.70), while fConc and fConc1 have very high correlations (0.98). Strong negative correlations were found between fSize and fConc (-0.85) and fConc1 (-0.81), indicating an opposite relationship between size and concentration in gamma measurements. Variables such as fDist indicated moderate positive correlations with fLength (0.42) and fSize (0.44), meaning that the distribution distance can have an effect on the size and length of features in the data. Variables such as fM3Long and fAlpha show weaker correlations with other variables, meaning that they have less influence on gamma identification.

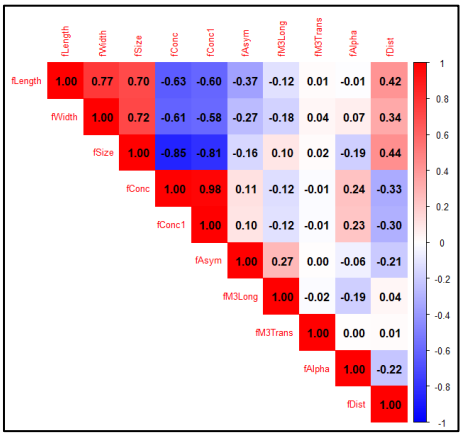

Figure 9. Relationship between variables in the dataset used for Gamma identification.

Figure 9 interprets the relationship between variables in the dataset used for gamma identification. The degree of correlation strength between the variables, with blue showing a strong negative correlation, white for a weak correlation or no correlation, and red for a strong positive correlation. Some variables, such as fLength and fWidth, indicate a significant positive correlation, as seen by the intense red color. And, variables such as fSize and fConc have a strong negative correlation, as seen by the dark blue color.

3.6.2 PCA Analysis

Based on the PCA results, the first principal component (PC1) has the largest contribution to the variability of the data, with a standard deviation of 2.055, which interprets the influence of the main factors such as length (fLength), width (fWidth),

and size (fSize) on the variation of the data. The second principal component (PC2) also implies a significant contribution, mainly influenced by factors such as asymmetry (fAsym) and M3 length (fM3Long).

Table 6. The contribution of each feature in improving accuracy and its importance in the Random Forest model.

Component	SD	PV	PV Kumulatif
PC1	**2.0552**	**42.24%**	**42.24%**
PC2	1.2551	15.75%	57.99%
PC3	1.0059	10.12%	68.11%
PC4	0.997	9.94%	78.05%
PC5	0.8614	7.42%	85.47%
PC6	0.8063	6.50%	91.97%
PC7	0.6384	4.08%	96.05%
PC8	0.4694	2.20%	98.25%
PC9	0.3934	1.55%	99.80%
PC10	0.1423	0.20%	100.00%

Principal component analysis (PCA) on gamma or hadron prediction data for gamma identification reveals that the first component (PC1) has the largest contribution to data variation, with a variance proportion of 42.24%. This component is followed by the second component (PC2) which contributes 15.75% and the third component (PC3) with a contribution of 10.12%. The first to third components (PC1, PC2, and PC3) explained about 68.11% of the total variance in the dataset. PCA also reveals further components, such as PC4 to PC10, which interpret a smaller contribution to the variation in the data, with a decreasing proportion of the variance. The PC4 component, for example, contributed only 9.94%, and the PC5 component contributed 7.42%. Cumulatively, the principal components up to PC6 explain more than 90% of the data variance.

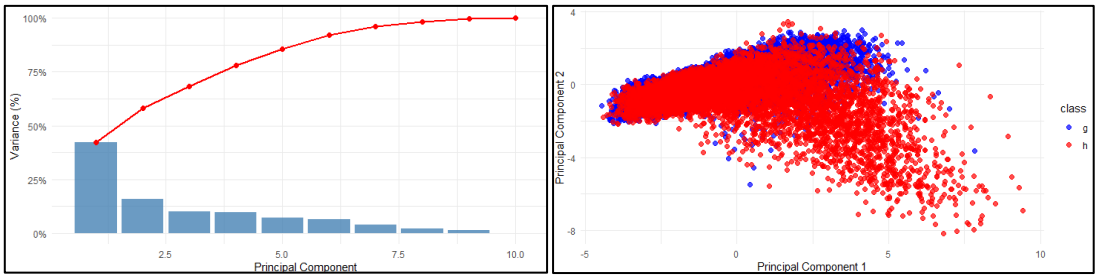

Figure 10. Contribution of components to total variance and 2D separation of data classes based on principal component scores.

Figure 10 displays two important aspects of principal component analysis (PCA). The left image visualizes the explained variance showing the contribution of each component to the total variance of the data. The first component (PC1) has the largest contribution, followed by the second component (PC2), which overall explains most of the variation in the dataset. This indicates that most of the information required for analysis can be found in the first few principal components. On the right is a visualization of PCA in two dimensions (2D) showing a clear separation between the classes of data based on the principal component scores. Two principal components (PC1 and PC2), the data is well clustered, which can identify patterns or structures in the data based on the classes present.

Table 7. Contribution of key features to principal components (PC1 and PC2) in dominant factor analysis for Gamma vs. Hadron prediction.

Feature	PC1	PC2
fSize	**0.4489**	-
fConc	-0.4416	-
fConc1	-0.4292	-
fLength	0.4083	-0.2257
fWidth	0.3951	-0.2453
fM3Long	-	**0.5996**
fAsym	-	0.4869
fAlpha	-	-0.4567

Table 7 is the contribution of the main features in the first (PC1) and second (PC2) principal components in the dominant factor analysis for the identification of Gamma or Hadron prediction data.

3.7 Geometric and Dynamical Studies on Photon Distribution

Based on the calculated correlation matrix, there is a significant positive correlation between fLength and fWidth, with a correlation value of 0.77. This indicates that the greater the focal length, the greater the focal width. However, the correlation between fAlpha and the other two features (fLength and fWidth) is very weak, with correlation values of -0.009 and 0.066 respectively, indicating that alpha angle does not have a strong relationship with focal length and width in this dataset.

As can be seen in Figure 11 on the left, the histograms and boxplots for the three geometric parameters-fLength, fWidth, and fAlpha-show the asymmetrical distribution pattern of the data as well as the presence of notable extreme values. In general, the histogram shown in the left image below reveals that most of the fLength values are concentrated in the lower range, with a long right tail indicating a right-skewed distribution. This

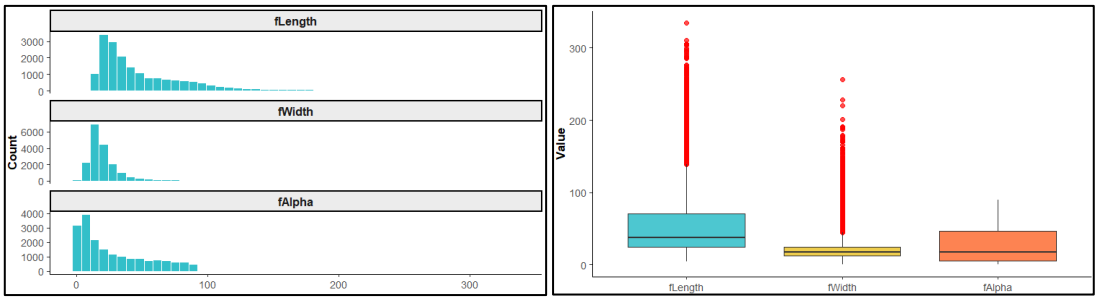

Figure 11. Geometric distribution of fLength, fWidth, and fAlpha with outlier detection on Telescope dataset.

Table 8. Descriptive statistics and geometric distribution of fLength, fWidth, and fAlpha features on Telescope dataset.

Parameter	Min	Q1	Median	Mean	Q3	Max	Interpreting
fLength	4.28	24.34	37.15	**53.25**	70.12	334.18	Widespread distribution with large outliers; data dominant in the mid-range.
fWidth	0	11.86	17.14	22.18	24.74	256.38	A similar pattern to fLength; the majority of the data is of medium value with extreme scatter.
fAlpha	0	5.55	17.68	27.65	45.88	90	More evenly distributed; tends to be stable with some high values.

pattern is seen in the maximum value of fLength that far exceeds the third quartile (Q3), where the maximum value reaches 334.18 while the median is only 37.15. The same can be seen for fWidth and fAlpha, albeit with a different range of values. fWidth has a narrower distribution and is relatively concentrated between Q1 (11.86) to Q3 (24.74), but still shows positive skewness with a maximum value of 256.38. Meanwhile, fAlpha can be seen to have a more spread out and varied distribution pattern, from 0 to 90, with a median value (median= 17.68) that is much lower than the maximum value, reinforcing the indication of distribution imbalance.

Figure 11 on the right is a Boxplot representing for visual confirmation the presence of significant outliers in all three features, especially in fLength and fWidth, where the outliers appear as red dots outside the whiskers. These outliers are important to note as they can affect further analysis such as correlation and predictive modeling.

Table 8 interprets the three main geometric parameters in the telescope dataset, namely fLength, fWidth, and fAlpha. In general, the distributions of fLength and fWidth show a tendency to be widely spread with maximum values far above the mean, indicating the presence of significant outliers. In contrast, fAlpha shows a more even distribution with a predominance of low to medium values, meaning that the alpha angle is stable in the measurement. All three features are not normally distributed and have a high range of variability, which is important to note in further analysis.

Figure 12 is a correlation matrix that shows the relationship between the geometric parameters visually and quantitatively. It can be seen that there is a strong positive correlation between fLength

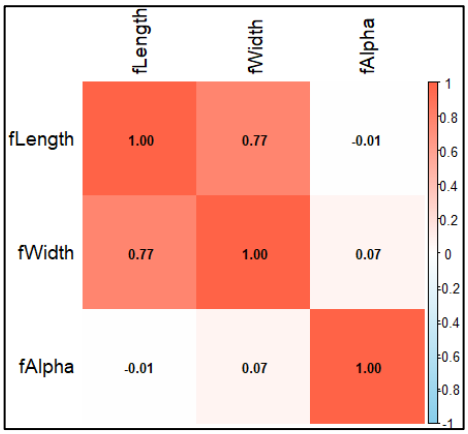

Figure 12. Geometric parameter correlation matrix.

and fWidth with a value of 0.771, indicating that a larger focal length tends to be accompanied by a larger focal width. In contrast, the relationship between fAlpha and the other two parameters is very weak, with an almost zero correlation to fLength (-0.009) and a very low correlation to fWidth (0.066), indicating that the alpha angle has no significant relationship with the focal dimensions in this data. The colors in figure 12 emphasize the strength and direction of the relationship, from light blue (negative/weak) to bright red (positive/strong).

Linear regression analysis reinforced these findings. When fLength was used as a predictor for fWidth, a model was obtained with a positive coefficient of 0.3337, which was highly significant ($p < 0.001$), with an R^2 value of 0.5937. This means that approximately 59.4% of the variation in fWidth can be explained by the variation in fLength. This model is very robust and shows a good linear relationship. In contrast, when fLength was used to predict fAlpha, the results were not significant ($p = 0.226$) and the R^2 value was very

low (0.000077), indicating that fLength has no significant influence on fAlpha. A slightly different result was found when fWidth was used as a predictor for fAlpha, where the regression yielded a small positive coefficient (0.094), yet statistically significant ($p < 0.001$). However, the R^2 value was only 0.00436, indicating that this relationship was weak and only about 0.4% of the variation in fAlpha could be explained by fWidth.

What has been explained is supported in Figure 13, which shows a relationship pattern that matches the numerical results. At the top of the graph, the correlation values between the pairs are shown, where only the fLength-fWidth relationship appears strong. At the bottom of the graph, the scatterplot with the regression line shows that the fLength-fWidth relationship forms a clear linear pattern, while the other two relationships are scattered and do not follow a strong linear pattern. The density curves on the diagonal of the plot interpret the distribution of each variable, and reinforce the descriptive results that fLength and fAlpha tend to have a wider distribution than fWidth.

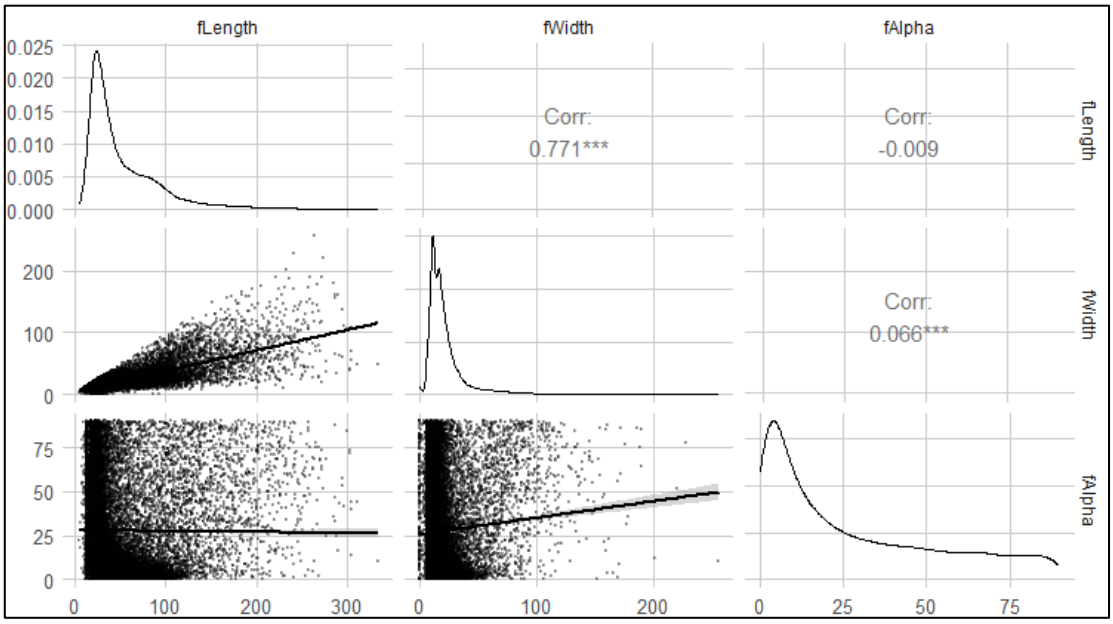

Figure 13. Geometric parameter pair relationship (fLength, fWidth, fAlpha).

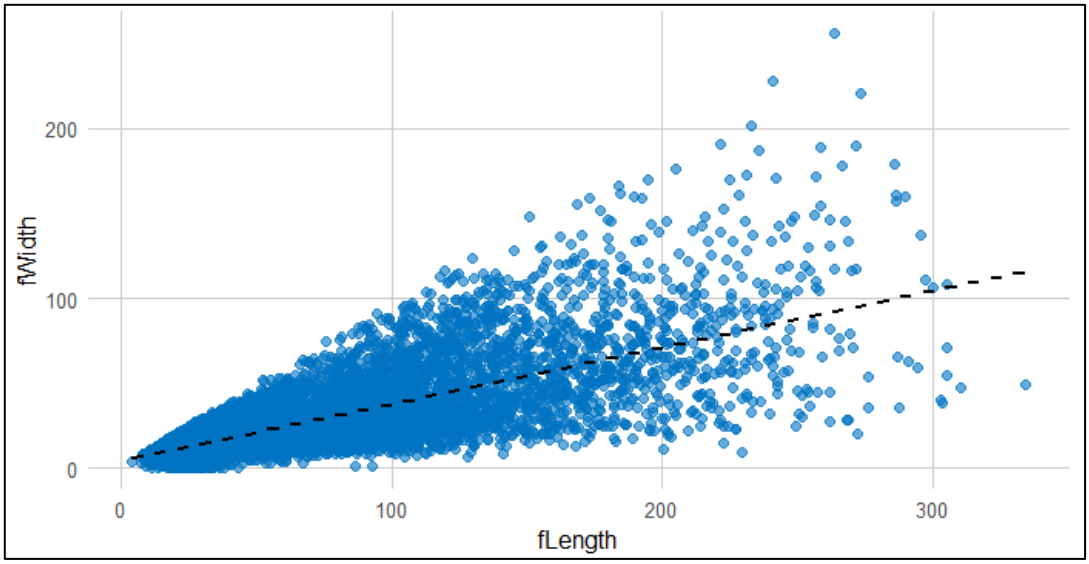

Figure 14. Relationship between fLength and fWidth.

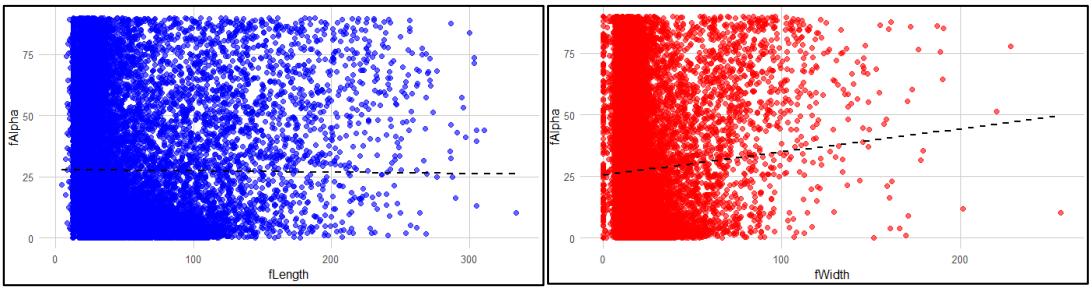

Figure 15. The relationship between two geometric parameters, fLength and fWidth, and the variable fAlpha.

Table 9. Results of simple linear regression analysis between fAlpha and two predictor variables (fLength and fWidth).

Relationship	Coefficient of Intercept	Coefficient X	Std. Error	R-squared	p-value
fAlpha ~ fLength	27.9337	**-0.0054**	0.0045	0.000077	0.226
fAlpha ~ fWidth	25.5608	**0.094**	0.0103	**0.004364**	**< 2e-16**

Figure 14 shows the relationship between fLength and fWidth. The figure shows that there is a significant and positive relationship between the two variables. The linear regression model yields a coefficient for fLength of 0.334, which means that every 1-unit increase in fLength will be followed by a 0.334-unit increase in fWidth, with all other conditions held constant. This coefficient is highly significant, with a very small p-value ($p < 2e\text{-}16$), meaning that this relationship does not occur by chance. The standardized residual of 11.69 shows the variation around the predicted value generated by the model, indicating that while the model does a good job of describing the data, there is still considerable deviation between the predicted and actual observed values. The Multiple R-squared value of 0.5937 indicates that about 59% of the variation in fWidth can be explained by this regression model with fLength as the main predictor.

Figure 15 presents the relationship between fLength and fAlpha, where the linear regression results show that there is no statistically significant relationship between the two variables. This insignificance is also reflected in the high p value and the very low coefficient of determination (R^2). For completeness, the Table 9 presents the results of a simple linear regression analysis that evaluates the relationship between the two main geometric parameters, fLength and fWidth, and the variable fAlpha, which clarifies the level of contribution of each variable to the change in the parameter fAlpha.

In the relationship between fLength and fAlpha, the regression analysis results as shown in the table 9 show that there is no statistically significant relationship between the two variables This is reflected in the p value for the fLength coefficient of 0.226, which is far above the significance threshold of 0.05. In addition, the very small and negative value of the regression coefficient (-0.0054) indicates that an increase in focal length (fLength) hardly contributes to the change in alpha angle (fAlpha). The very low R-squared value (0.000077) confirms that the variation in fAlpha cannot be explained by fLength. Thus, scientifically, there is no meaningful linear relationship between the focal length and angular parameters in this optical system. Meanwhile, the relationship between fWidth and fAlpha shows a statistically significant relationship, as seen from the very small p value (<2e-16) for the coefficient of fWidth. The positive coefficient of 0.094 indicates that an increase in focal width (fWidth) is positively correlated to an increase in fAlpha. While this relationship is significant, the R-squared value of only 0.0043 indicates that the strength of the relationship is very weak and only explains about 0.43% of the variation in fAlpha. Although there are indications that an increase in focal width tends to be followed by an increase in alpha angle, this effect is very small and is likely to be influenced by other variables not included in the model.

3.8 Effect of Distance (fDist) on Gamma Identification

Researchers have conducted research so that the distribution of signal distances (fDist) based on two classes, namely Gamma (g) and Hadron (h), is depicted through histograms and density curves as shown in the figure 16. The graph in Figure 16 shows a comparison of the distribution of these two classes with a focus on important statistical characteristics such as the mean and median.

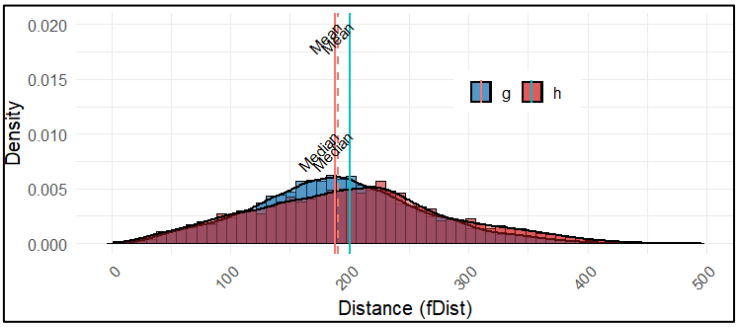

Figure 16. Comparison of the distribution of the two classes with their mean and median against the 2 classes of gamma and hadron.

The results of the statistical calculations can be seen in Table 9 where the mean of the signal distance in the Gamma class is about 190, while that of the Hadron class is slightly higher at 200. The median values for both classes are almost similar, 188 for Gamma and 200 for Hadron respectively, which means that the distribution of signal distance in both classes has almost the same center. However, the difference between the mean and median values of the two classes indicates that the data distribution is not completely symmetrical, although they are not too skewed (low skewness values, 0.194 for Gamma and 0.205 for Hadron). Furthermore, the standard deviation (SD) shows the variation or spread of the data around the mean. The Hadron class has a larger standard deviation (81.9) compared to the Gamma class (70.3), indicating that the signal distance in Hadron is more dispersed or has greater variation. The kurtosis analysis shows that the signal distance distribution for both classes is relatively flat, with negative kurtosis values (-0.117 for Gamma and -0.272 for Hadron). This means that the distributions of both classes have lower peaks and thinner tails compared to a normal distribution. Table 10 shows the statistical results for the fDist distribution by class (Gamma and Hadron) which includes the mean, median, standard deviation (SD), skewness, kurtosis, minimum value (Min), and maximum value (Max).

As can be seen in Figure 17 which displays the density graph showing the distribution of the variable fDist based on the two clusters generated,

Table 10. fDist distribution statistics by classes (Gamma and Hadron)

Statistics	Gamma (g)	Hadron (h)
Mean	**190**	**200**
Median	188	200
SD	70.3	81.9
Skewness	0.194	0.205
Kurtosis	-0.117	-0.272
Min	5.75	1.28
Max	**450**	**496**

two clearly separated distributions can be observed, each representing two different clusters. The first cluster, represented by the blue color, shows a narrower distribution and is more centered on the lower values of fDist. While the second cluster, represented by the orange color shows a wider distribution with peaks at higher values of fDist. This means that the first cluster may be more associated with data that has low fDist values, while the second cluster covers more of the higher fDist values, which implies a difference in the type of object or phenomenon observed, such as the difference between gamma and hadron signals. The right side of Figure 17 is a boxplot that shows a comparison of the fDist distribution between the two clusters, further clarifying the differences. The first cluster has a narrower range with relatively few outlier values, indicating the fDist in the cluster is more centered and consistent. On the other hand, the second cluster has a wider range, with some more noticeable outliers, indicating a greater variation in the fDist values in this cluster.

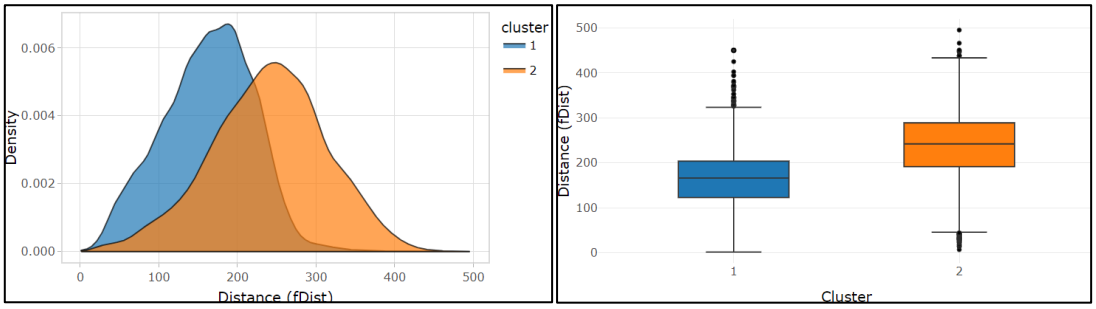

Figure 17. Density curve and boxplot of fDist variable based on K-Means clustering results.

Table 11. fDist feature statistics by cluster.

Cluster	Mean fDist	Median fDist	SD fDist	Min fDist	Max fDist	Skewness fDist	Kurtosis fDist
Cluster 1	**162**	**166**	57.7	1.28	450	-0.0791	-0.141
Cluster 2	**239**	**242**	73.2	6.56	496	-0.175	-0.0856

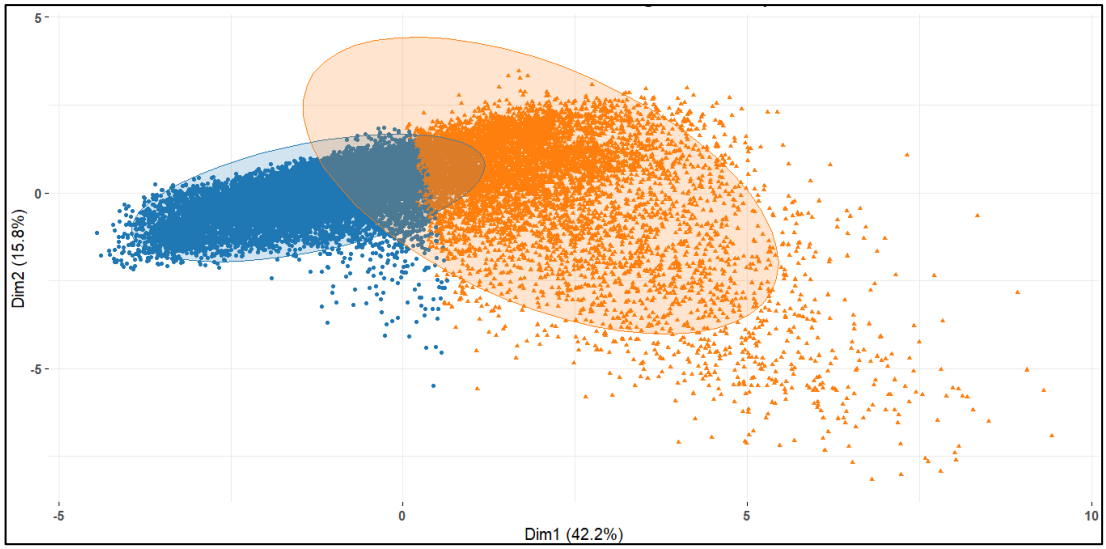

Figure 18. PCA-based clustering using K-means method of ten main features of telescope signals.

It can be seen in Figure 17 that the first cluster has a mean fDist of 162, which is lower than the second cluster which has a mean fDist of 239. Furthermore, the standard deviation in the second cluster is higher, at 73.2, compared to the first cluster which is only 57.7, which indicates the fDist values in the second cluster are more dispersed. Both clusters show skewness values close to zero (-0.0791 for the first cluster and -0.175 for the second cluster), indicating a relatively symmetrical distribution in both clusters. The slightly negative kurtosis in both clusters indicates that their distributions have lighter tails compared to the normal distribution.

Table 11 shows the descriptive statistics of the fDist (signal distance) variable in the two clusters obtained through K-means clustering analysis, namely Cluster 1 and Cluster 2. It can be seen that Cluster 2 shows an overall higher fDist value compared to Cluster 1, with a larger mean and median, and a wider variation (SD). Cluster 1 has a narrower distribution with a lower minimum value. The skewness and kurtosis values that are close to zero in both clusters indicate a distribution of fDist that is relatively symmetrical and slightly flatter than the normal distribution.

As can be seen in Figure 18, which is a PCA-based clustering using the K-means method of the ten main features of the telescope signal, the data can be grouped into two fairly separate clusters, which are assumed to represent two different types of particles, namely gamma and hadrons. PCA shows the distribution of the data in the two main dimensions that best explain the variation between observations. The two normal ellipses surrounding each cluster represent a relatively compact internal distribution, indicating that the separation between clusters is stable and representative. The blue and orange colors used emphasize the visual contrast between the groups, making the cluster structure easily recognizable to the naked eye.

Table 12. K-means clustering and its relationship with Gamma vs Hadron classes.

Cluster	Number of Observations	Gamma (%)	Hadron (%)
1	11.25	**66.61**	33.39
2	7.77	**62.27**	37.73

Quantitative cross-tabulation between the clustering results as seen in Table 12 and the actual class labels shows that in the first cluster, about 66.6% are from the gamma class and 33.3% from hadron, while in the second cluster, the proportion of gamma is 62.3% and hadron is 37.7%. Although there is no perfect separation, the dominance of one class in each cluster shows that this unsupervised approach is quite effective in revealing the natural structure of the data based on the morphological features of the signals. Further statistical analysis of the variable `fDist' (distance to center of the camera) reveals that the average value of fDist in the hadron class (200.04) is higher than that in the gamma class (189.61), with the scatter (standard deviation) also greater in hadron (81.91 vs. 70.26).

The skewness values are positive and close to zero (0.194 for gamma, 0.205 for hadron) meaning that the distribution of fDist tends to be symmetrical in both classes, although the hadron class is slightly more skewed to the right. The negative kurtosis values in both classes (-0.117 and -0.272) indicate that the fDist distribution tends to be flatter than a normal distribution (platykurtic), with the hadron class having a slightly flatter distribution.

Based on the results of the Welch Two Sample t-test conducted to compare the average fDist values between the Gamma (g) and Hadron (h) classes, the statistical value of t = -8.6154 with degrees of freedom (df) of 12.059, and a p-value of <2.2e-16. This very small p-value means that there is a statistically significant difference between the average fDist in the two classes, with a 95% confidence level. Specifically, the average fDist value of the Gamma class is 190.23, while that of the Hadron class is 200.43, indicating that on average, the Hadron signal has a larger fDist distance than the Gamma signal. The 95% confidence interval for the mean difference is in the range [-12.53, -7.88], which is completely below zero. This further strengthens the evidence that the difference is not due to sample chance, but due to real differences in the population. The larger average distance in the Hadron class means that Hadron events tend to produce Cherenkov light distribution patterns that are more scattered or away from the center than Gamma events, which could be related to the different physical properties of the particles.

3.9 Gamma and Hadron Spectrum Simulation

Researchers conducted Monte Carlo simulations applied to the telescope data successfully replicating the distributions of the analyzed features, namely `fSize`, `fAlpha`, and `fDist`, for both particle classes, namely gamma and hadron. For each feature, a Kolmogorov-Smirnov (KS) test was performed to compare the cumulative distributions between the original data and the simulated results. The KS test results show that there is no significant difference between the original and simulated data, with very high p-values for both particle classes. Table 13 summarizes the Kolmogorov-Smirnov (KS) test results for the fSize, fAlpha, and fDist features, as well as the D statistic and p-value for both gamma and hadron classes:

Table 13. Kolmogorov-Smirnov (KS) test results for the fSize, fAlpha, and fDist features, as well as the D statistic and p-value.

Feature	Class	D-statistic	P-value
fSize	Gamma	**0.008352**	0.782894
	Hadron	**0.008523**	0.968337
fAlpha	Gamma	**0.007704**	0.857713
	Hadron	**0.011663**	0.753284
fDist	Gamma	**0.006001**	0.979456
	Hadron	**0.009121**	0.943649

Table 13 shows the D statistic and p-value generated from the Kolmogorov-Smirnov test for each feature in both classes. The very high p-value (more than 0.75) indicates that there is no significant difference between the original data distribution and the simulated data for each feature tested. As seen for the `fSize` feature, the KS test yields a p-value of 0.7829 for the gamma class and 0.9683 for the hadron class, meaning that the original and simulated data distributions are not significantly different. Similarly, for the `fAlpha` and `fDist` features, the p-values for both particle classes are above 0.75, indicating no significant difference in distribution between the original and simulated data.

Based on the distribution of features performed through Monte Carlo simulation, it can be seen that for each of the analyzed features, namely fSize, fAlpha, and fDist, the distribution of the original data (observation data) and simulated data show very similar patterns, with statistically insignificant differences.

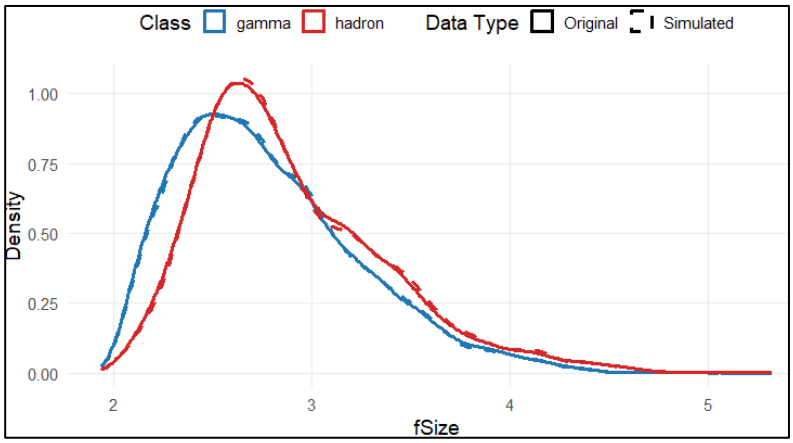

Figure 19. Comparison between original and simulated data for Gamma and Hadron classes of the fSize distribution.

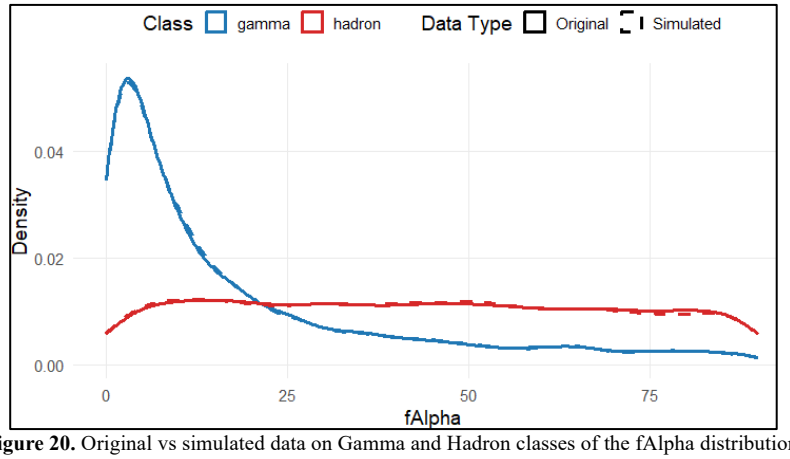

Figure 20. Original vs simulated data on Gamma and Hadron classes of the fAlpha distribution.

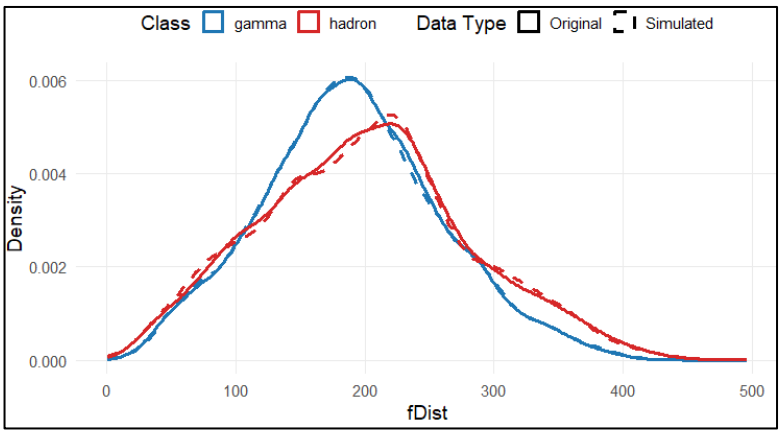

Figure 21. Original vs simulated data on Gamma and Hadron classes of the fDist distribution.

In the fSize feature (Figure 19), both the original data distributions for the gamma and hadron classes, as well as the simulated distributions, appear to have almost identical shapes. The Kolmogorov-Smirnov (KS) test shows very high p-values (0.7828942 for gamma and 0.9683368 for hadron), indicating that there is no significant difference between the original and simulated data on this feature. This indicates that the simulation model can well represent the size distribution of the fSize feature for both the gamma and hadron classes.

Similarly, for the fAlpha feature (Figure 20), the distributions of the original and simulated data for both classes also show very clear similarities. The KS tests for gamma (0.8577132) and hadron (0.7532842) show p-values much greater than 0.05, which further confirms that there is no significant difference between the original and simulated data, implying the accuracy of the simulation in replicating the distribution of fAlpha values in both classes.

As seen in the fDist feature (Figure 21), the distributions of the original data and the simulated results show a very good match, both for gamma and hadron. The very high p-values (0.9794560 for gamma and 0.9436487 for hadron) in the Kolmogorov-Smirnov test indicate that the distributions between the original data and the simulated data are not significantly different, further reinforcing the finding that the simulation successfully mimics the distribution pattern of the distance between points in the fDist feature.

4 Conclussion

The study evaluated the classification of gamma and hadron signals using machine learning methods, namely Decision Tree, Random Forest, and Neural Network. The analysis showed that Random Forest performed best with an accuracy of 87.96%, and identified fAlpha, fSize, and fWidth as the most influential features, as confirmed by SHAP analysis. In contrast, Neural Network performed poorly with an accuracy of only 19.46% and a negative kappa value, indicating that either the model architecture is not optimal or the available features are not informative enough to distinguish patterns in the dataset. Correlation

analysis revealed significant relationships between variables, such as a positive correlation between fLength and fWidth and a negative correlation between fConc and fSize. Principal Component Analysis (PCA) identified that most of the information in the dataset can be represented by a few principal components, especially PC1 which explains 42.24% of the data variability and is influenced by fLength, fWidth, and fSize. This means that dimensionality reduction can improve classification efficiency without losing important information. The geometric parameter distribution means that the length and width of the gamma signals are more concentrated than the hadrons, while the distance (fDist) plays a role in classification, with the gamma signals tending to be closer than the hadrons. Clustering using K-means reinforces these findings, noting a more focused distribution pattern of gamma signals. As an additional validation, Monte Carlo simulations compared the distribution of key features between the original data and the simulated results, which showed a good fit of the patterns, confirming the accuracy of the classification method used.

Practically, the research shows that Random Forest can be implemented in an automatic classification system to detect gamma signals more accurately. Feature optimization based on PCA and SHAP analysis can significantly improve model performance. The integration of this system in telescope observatories will help astronomers identify gamma signals more efficiently, reducing the need for manual analysis. The application of Monte Carlo simulation as a validation method ensures that the model predictions match the observational data patterns. The results of this study also have broad implications in astrophysics, especially in improving the precision of gamma radiation source identification and hadrons in future studies.

Further research can be focused on developing a more optimal Neural Network architecture to improve the accuracy of gamma and hadron classification, for example with hyperparameter tuning techniques or the use of more complex models such as Convolutional Neural Network (CNN). Exploration of additional features or feature engineering techniques can be done to improve data representation. The use of other classification methods, such as Gradient Boosting or Support Vector Machine (SVM), can also be compared to see potential performance improvements. In terms of application, the implementation of this classification system in a telescope observatory can be tested directly to assess its effectiveness in real conditions. Further validation using datasets from various observatories or simulations with more varied parameters can strengthen the generalizability of the model in astrophysical research.

Acknowledgement

The authors gratefully acknowledge the NASA Fermi Gamma-ray Space Telescope (Fermi LAT) team for providing open access to the gamma-ray and hadron signal datasets via the NASA Open Data Portal, which were vital to this study. We appreciate the support from Swiss German University, Jayakarta College of Information and Computer Management, and Universitas Negeri Medan. Special thanks to all colleagues and reviewers for their valuable feedback.

References

[1] S. Ohm, C. van Eldik, and K. Egberts, "γ/hadron separation in very-high-energy γ-ray astronomy using a multivariate analysis method," *Astroparticle Physics*, vol. 31, no. 5, pp. 383–391, 2009.

[2] F. A. Aharonian, *Very high energy cosmic gamma radiation: a crucial window on the extreme Universe*. World Scientific, 2004.

[3] R. A. Ong, "Very high-energy gamma-ray astronomy," *Physics Reports*, vol. 305, no. 3–4, pp. 93–202, 1998.

[4] F. A. Aharonian, D. Khangulyan, and L. Costamante, "Formation of hard very high energy gamma-ray spectra of blazars due to internal photon–photon absorption," *Monthly Notices of the Royal Astronomical Society*, vol. 387, no. 3, pp. 1206–1214, 2008.

[5] L. Evans, "The large hadron collider," *Annual review of nuclear and particle science*, vol. 61, no. 1, pp. 435–466, 2011.

[6] S. Myers, "The large hadron collider 2008–2013," *International Journal of Modern Physics A*, vol. 28, no. 25, p. 1330035, 2013.

[7] J. Eberth and J. Simpson, "From Ge (Li) detectors to gamma-ray tracking arrays–50 years of gamma spectroscopy with germanium detectors," *Progress in Particle and Nuclear Physics*, vol. 60, no. 2, pp. 283–337, 2008.

[8] G. Naidu, T. Zuva, and E. M. Sibanda, "A review of evaluation metrics in machine learning algorithms," presented at the Computer Science On-line Conference, Springer, 2023, pp. 15–25.

[9] A. M. Carrington *et al.*, "Deep ROC analysis and AUC as balanced average accuracy to improve model selection, understanding and interpretation," *arXiv preprint arXiv:2103.11357*, 2021.

[10] A. Razdan, A. Haungs, H. Rebel, and C. Bhat, "Image and non-image parameters of atmospheric Cherenkov events: a comparative study of their γ-ray/hadron classification potential in ultrahigh energy regime," *Astroparticle Physics*, vol. 17, no. 4, pp. 497–508, 2002.

[11] M. Krause, E. Pueschel, and G. Maier, "Improved γ/hadron separation for the detection of faint γ-ray sources using boosted decision trees," *Astroparticle Physics*, vol. 89, pp. 1–9, 2017.

[12] M. Andrews, M. Paulini, S. Gleyzer, and B. Poczos, "End-to-end physics event classification with CMS open data: Applying image-based deep learning to detector data for the direct classification of collision events at the LHC," *Computing and Software for Big Science*, vol. 4, pp. 1–14, 2020.

[13] F. Carrillo-Perez, L. J. Herrera, J. M. Carceller, and A. Guillén, "Deep learning to classify ultra-high-energy cosmic rays by means of PMT signals," *Neural Computing and Applications*, vol. 33, pp. 9153–9169, 2021.

[14] H. Kolanoski and N. Wermes, *Particle Detectors: Fundamentals and Applications*. Oxford University Press, USA, 2020.

[15] D. Bose, V. Chitnis, P. Majumdar, and A. Shukla, "Galactic and extragalactic sources of very high energy gamma rays," *The European Physical Journal Special Topics*, vol. 231, no. 1, pp. 27–66, 2022.

[16] D. J. Thompson and C. A. Wilson-Hodge, "Fermi gamma-ray space telescope," in *Handbook of X-ray and Gamma-ray Astrophysics*, Springer, 2022, pp. 1–31.

[17] A. M. Bagher, "Advantages of gamma radiation in science and industry," *Journal of Advanced Physics*, vol. 3, no. 2, pp. 97–103, 2014.

[18] C. Amsler, *The quark structure of hadrons*. Springer, 2018.

[19] K. K. Rana, "A survey on decision tree algorithm for classification," *International journal of Engineering development and research*, vol. 2, no. 1, pp. 1–5, 2014.

[20] B. Charbuty and A. Abdulazeez, "Classification based on decision tree algorithm for machine learning," *Journal of Applied Science and Technology Trends*, vol. 2, no. 01, pp. 20–28, 2021.

[21] S. A. Macskassy and F. Provost, "Classification in networked data: A toolkit and a univariate case study.," *Journal of machine learning research*, vol. 8, no. 5, 2007.

[22] N. Meinshausen, "Node harvest," *The Annals of Applied Statistics*, pp. 2049–2072, 2010.

[23] S. Nowozin, "Improved information gain estimates for decision tree induction," *arXiv preprint arXiv:1206.4620*, 2012.

[24] D. Alpay, "Entropy: Discrete Case," in *Exercises in Applied Mathematics: With a View toward Information Theory, Machine Learning, Wavelets, and Statistical Physics*, Springer, 2024, pp. 597–633.

[25] D. Kocev, C. Vens, J. Struyf, and S. Džeroski, "Tree ensembles for predicting structured outputs," *Pattern Recognition*, vol. 46, no. 3, pp. 817–833, 2013.

[26] X. Zhou, P. Lu, Z. Zheng, D. Tolliver, and A. Keramati, "Accident prediction accuracy assessment for highway-rail grade crossings using random forest algorithm compared with decision tree," *Reliability Engineering & System Safety*, vol. 200, p. 106931, 2020.

[27] O. Allouche, A. Tsoar, and R. Kadmon, "Assessing the accuracy of species distribution models: prevalence, kappa and the true skill statistic (TSS)," *Journal of applied ecology*, vol. 43, no. 6, pp. 1223–1232, 2006.

[28] Ž. Vujović, "Classification model evaluation metrics," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 6, pp. 599–606, 2021.

[29] A. Tharwat, "Classification assessment methods," *Applied computing and informatics*, vol. 17, no. 1, pp. 168–192, 2021.

[30] M. Banerjee, M. Capozzoli, L. McSweeney, and D. Sinha, "Beyond kappa: A review of interrater agreement measures," *Canadian journal of statistics*, vol. 27, no. 1, pp. 3–23, 1999.

[31] B. H. Menze *et al.*, "A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data," *BMC bioinformatics*, vol. 10, pp. 1–16, 2009.

[32] D. Rengasamy, B. C. Rothwell, and G. P. Figueredo, "Towards a more reliable interpretation of machine learning outputs for safety-critical systems using feature importance fusion," *Applied Sciences*, vol. 11, no. 24, p. 11854, 2021.

[33] H. Han, X. Guo, and H. Yu, "Variable selection using mean decrease accuracy and mean decrease gini based on random forest," presented at the 2016 7th ieee international conference on software engineering and service science (icsess), IEEE, 2016, pp. 219–224.

[34] E. Muchai and L. Odongo, "Comparison of crisp and fuzzy classification trees using gini index impurity measure on simulated data," *European Scientific Journal*, vol. 10, no. 18,

2014.
[35] S. Tangirala, "Evaluating the impact of GINI index and information gain on classification using decision tree classifier algorithm," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 2, pp. 612–619, 2020.
[36] M. Buscema, "Back propagation neural networks," *Substance use & misuse*, vol. 33, no. 2, pp. 233–270, 1998.
[37] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, 2022.
[38] M.-C. Popescu, V. E. Balas, L. Perescu-Popescu, and N. Mastorakis, "Multilayer perceptron and neural networks," *WSEAS Transactions on Circuits and Systems*, vol. 8, no. 7, pp. 579–588, 2009.
[39] D. Filimonov and V. Poroikov, "Probabilistic approaches in activity prediction," 2008.
[40] H. Jain, Y. Prabhu, and M. Varma, "Extreme multi-label loss functions for recommendation, tagging, ranking & other missing label applications," presented at the Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 935–944.
[41] H. Qin *et al.*, "Forward and backward information retention for accurate binary neural networks," presented at the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 2250–2259.
[42] E. Trentin, "Networks with trainable amplitude of activation functions," *Neural Networks*, vol. 14, no. 4–5, pp. 471–493, 2001.
[43] J. Li, "Area under the ROC Curve has the most consistent evaluation for binary classification," *PloS one*, vol. 19, no. 12, p. e0316019, 2024.
[44] A. S. Jadhav, "A novel weighted TPR-TNR measure to assess performance of the classifiers," *Expert systems with applications*, vol. 152, p. 113391, 2020.
[45] J. Davis and M. Goadrich, "The relationship between Precision-Recall and ROC curves," presented at the Proceedings of the 23rd international conference on Machine learning, 2006, pp. 233–240.
[46] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.
[47] S. D. Walter, "The partial area under the summary ROC curve," *Statistics in medicine*, vol. 24, no. 13, pp. 2025–2040, 2005.
[48] E. Algaba, V. Fragnelli, and J. Sánchez-Soriano, *Handbook of the Shapley value*. CRC Press, 2019.
[49] M. Sastre and A. Trannoy, "Shapley inequality decomposition by factor components: Some methodological issues," *Journal of Economics*, vol. 77, pp. 51–89, 2002.
[50] S. P. Mishra *et al.*, "Multivariate statistical data analysis-principal component analysis (PCA)," *International Journal of Livestock Research*, vol. 7, no. 5, pp. 60–78, 2017.
[51] B. M. S. Hasan and A. M. Abdulazeez, "A review of principal component analysis algorithm for dimensionality reduction," *Journal of Soft Computing and Data Mining*, vol. 2, no. 1, pp. 20–30, 2021.
[52] R. Y. Rubinstein and D. P. Kroese, *Simulation and the Monte Carlo method*. John Wiley & Sons, 2016.