# Classification of Gamma-Ray Bursts using Machine Learning

A Project Report Submitted

in Fulfilment of the Requirements

for the

## MINOR DEGREE

in

**Data Science**

*by*

**Nishil Mehta**

**(Roll No. IMS18180)**


*to*

**SCHOOL OF PHYSICS**

**INDIAN INSTITUTE OF SCIENCE EDUCATION AND RESEARCH**

**THIRUVANANTHAPURAM - 695 551, INDIA**

*July, 2022*

# DECLARATION

I, **Nishil Mehta (Roll No: IMS18180)**, hereby declare that, this report entitled **"Classification of Gamma-Ray Bursts using Machine Learning"** submitted to Indian Institute of Science Education and Research Thiruvananthapuram towards partial requirement of **Minor Degree** in **Data Science**, is an original work carried out by me under the supervision of Dr. Shabnam Iyyani and has not formed the basis for the award of any degree or diploma, in this or any other institution or university. I have sincerely tried to uphold the academic ethics and honesty. Whenever an external information or statement or result is used then, that have been duly acknowledged and cited.

Thiruvananthapuram - 695 551 **Nishil Mehta**

July, 2022

# CERTIFICATE

This is to certify that the work contained in this project report entitled **"Classification of Gamma-Ray Bursts using Machine Learning"** submitted by **Nishil Mehta** (**Roll No: IMS18180**) to Indian Institute of Science Education and Research, Thiruvananthapuram towards the partial requirement of **Minor Degree** in **Data Science** has been carried out by him under my supervision and that it has not been submitted elsewhere for the award of any degree.

Thiruvananthapuram - 695 551

July, 2022

Dr. Shabnam Iyyani

Project Supervisor

# ACKNOWLEDGEMENT

I want to extend a sincere and heartfelt obligation towards all the personages without whom the completion of the project was not possible. I express my profound gratitude and deep regard to Dr. Shabnam Iyyani, IISER Thiruvananthapuram for her guidance, valuable feedback, and constant encouragement throughout the project. Her valuable suggestions were of immense help. I sincerely acknowledge her constant support and guidance during the project.

I am also grateful to the Indian Institute of Science Education and Research, Thiruvananthapuram, for allowing me to do this project and providing all the required facilities.

Thiruvananthapuram - 695 551 **Nishil Mehta**

July, 2022

# ABSTRACT

Gamma-ray bursts (GRBs), the brightest explosions, are known to occur in the Universe. Ground and space-based observatories have been studying GRBs for the last several decades. Categorizing GRBs according to a variety of factors can aid in determining progenitors. Clustering methods are essential for enumerating and describing the different Gamma-Ray Bursts (GRBs). However, their performance can be affected by several factors, such as the choice of clustering algorithm and inherently associated assumptions, the inclusion of variables in clustering, the nature of initialization methods used, the iterative algorithm, or the criterion used to judge the optimal number of groups supported by the data. Analysis of GRBs from the Fermi catalog was done using unsupervised methods. After initial analysis and testing algorithms and methods, all the variables used in the literature in different subsets - namely, the flux duration variables, fluence, and Band function parameters contain information on the clustering. Various models were explored for the classification of Fermi Catalog Data. Further analysis was done on the GRB catalog from Swift and ASTROSAT. The analysis here suggests that GRBs should be categorized into 2 classes using T90 and Alpha.

**Keywords:** Machine learning — Data Science — Gamma-Ray Bursts — Unsupervised learning — Clustering

# Contents

# List of Figures

# List of Tables

# Chapter 1
# Introduction

Gamma-ray bursts have always been exciting topic of research. The bimodal Gamma-ray Burst (GRB) duration distribution shows that GRBs can be divided into short/hard and long/soft classes at a T90 of around 2s [1].

The collapsar hypothesis [2] is supported by the observational link between long GRBs and the deaths of massive stars, which is provided by the association of long GRBs with star-forming galaxies [3] and Type Ic supernovae [4].

Strong evidence suggests compact-object mergers (neutron star-neutron star or neutron star-black hole) as the origins of short GRBs [5] [6]. Other origin hypotheses for short GRBs include neutron star-white dwarf mergers, double white dwarf mergers, and accretion-induced collapse of white dwarfs, which could result in an unstable magnetar remnant. The short-merger/long-collapsar paradigm has some notable exceptions, including the short-collapsar event [7] and long GRB without a supernova [8]. Numerous short-duration high-redshift GRBs have been proposed to originate from collapsars [9]. It is difficult to distinguish between GRBs based solely on duration because of the variation in GRB phenomenology (Also, due to the fact that different satellites measure different T90. This has been shown in Appendix A). Therefore, categorizing GRBs according to a variety of factors can aid in determining the progenitors.

Table 1 of [10] summarises the prior research and the number of components found in various GRB datasets. Depending on the sample, conditions, and techniques employed, between two and five kinds of GRBs are discovered. An initial study on the trial dataset was done in this study for 2, 3, and 4-dimensional trial datasets. Clusters were created using a gaussian distribution, and the distance and standard deviation were varied individually and combined. Different metric algorithms were tested to check which algorithm performed better in different conditions. All these metric algorithms (to determine the number of clusters) are described in the next chapter, along with GMM and KMeans clustering algorithms. In Chapter 3, the results of the tests are discussed. Results obtained from the Fermi analysis using 2 different methods are discussed in Chapter 4, along with the classification and properties of each class of GRB. The study is concluded in the last chapter. Finally, analysis of Swift and Astrosat GRB Data has been discussed in the Appendix.

# Chapter 2
# Machine Learning

Machine learning has become an integral part of science and research. Modern science, particularly physics, is characterized by the availability of large datasets. Data analysis has become crucial in fields like experimental particle physics, observational astronomy and cosmology, condensed matter physics, biophysics, and quantum computing.

K-Means Clustering scales effectively to large numbers of samples and has been implemented in a wide variety of application areas in numerous fields. K-Means is highly adaptable to new applications and guarantees convergence and scalability for big data sets.

A Gaussian mixture model is a probabilistic model that assumes that all data points were produced by combining a limited number of Gaussian distributions with unidentified parameters. Mixture models can be seen as a generalization of k-means clustering to include details of the covariance structure of the data as well as the locations of the latent Gaussian centers.

Hence, for this study, K-Means and GMM were chosen as the clustering algorithms. The code written for this study has been done with the help of the python package Scikit-learn [11].

## 2.1 Supervised learning

The data is labeled, i.e., the mapping of the input to output is known. The corrective algorithm makes appropriate changes during the iterative process if a mistake occurs in the model. Regression and classification are typical examples of supervised learning. The goal is to generate a model that can map the input to the output. It learns from a given set of input-output pairs and tries to predict the given input.

## 2.2 Reinforcement learning

This enables the machine or agent to adapt its behavior based on the environment's feedback. In this learning, the agent independently makes a series of choices, and as a result, either a +1 or a -1 will be given as the reward. The agent reevaluates its course in light of the final reward. In contrast to frequently used machine learning techniques, reinforcement problems are more closely related to the artificial intelligence approach.

## 2.3 Unsupervised learning

In unsupervised learning, the algorithm learns by itself without supervision, i.e., no target/output variable is provided. Finding hidden patterns and relationships in the data is the key. Unsupervised learning has no correct answers; instead, the algorithm seeks to find similarities between inputs so that they can be grouped and given a common category. Density estimation is a statistical method used in unsupervised learning. The focus would be on unsupervised learning, which is the main topic of this study.

The first question was how many classes of GRBs could be classified. To determine the number of clusters, the following algorithms were used:

1. AIC-BIC

   The "Akaike Information Criterion" helps choose a model from a limited number of models by providing a relative assessment of a model's quality for a particular data set. The information lost in the model is estimated using the number of parameters and the maximal likelihood estimate. The AIC metric provides a trade-off between model complexity and accuracy. It helps to prevent overfitting.

   $$AIC = -2(log - likelihood) + 2K$$

   where,

K is the number of estimated parameters.

Log-likelihood is a measure of model fit.

Bayesian Information Criterion (BIC) is also used for model selection. It includes an additional parameter of the number of data points over AIC. The penalty for extra parameters in BIC is more significant than in AIC. The BIC penalizes free parameters more severely than the AIC.

$$BIC = -2(log - likelihood) + log(N) * k$$

where,

N is number of sample data.

2. Silhouette Coefficient

   The separation distance between the generated clusters can be investigated using silhouette analysis. By showing how close each point in one cluster is to points in the neighboring clusters, the silhouette plot provides a visual method to assess characteristics like the number of clusters. The range of this metric is [-1, 1].

   $$S = \frac{b-a}{max(a,b)}$$

   where,

   a = mean intra-cluster distance

   b = mean nearest-cluster distance

   The sample is remote from the surrounding clusters if the silhouette coefficients (as these values are known) are close to +1. Indicated by a value of 0, a sample is on or near the boundary between two neighboring clusters. In contrast, negative values suggest that the sample may have been mistakenly assigned to another cluster.

3. Calinski-Harabasz Index [12]

   The Variance Ratio Criterion, commonly known as the Calinski-Harabasz index, is determined by dividing the sum of the dispersion between each cluster by the sum of the dispersion

inside each cluster (where the dispersion is the sum of squared distances).

$$CH = \frac{\sum_{k=1}^{K} n_k \times ||C_k - C||^2}{\sum_{k=1}^{K} \sum_{i=1}^{n_k} \times ||X_{ik} - C_k||^2} \times \frac{N-K}{K-1}$$

4. Davies–Bouldin index [13]

   The score is determined by averaging the within-cluster to between-cluster distances for each cluster and its most similar cluster, where similarity is defined as the ratio. Therefore, groups that are more evenly spaced apart will score higher.

   $$DB = \frac{1}{N} \sum_{i=1}^{N} \max_{i \neq j} \left(\frac{S_i + S_j}{d(c_i, c_j)}\right)$$

   With a minimum score of 0, better clustering is indicated by lower values.

5. Gap Statistics [14]

   Let $C_1, C_2, ..., C_k$, with $C_r$ denoting the indices of observations in $r$th cluster and $n_r = |C_r|$.

   $D_r = \sum_{i,j \in C_r} d_{ij}$

   be sum of pairwise distances for all points in cluster r and,

   $W_k = \sum_{r=1}^{k} \frac{1}{2n_r} D_r$.

   So, if the distance d is the squared Euclidean distance, then $W_k$ is the pooled within-cluster sum of squares around the cluster means. The sample size n is suppressed in this notation.

   The graph of $log(W_k)$ is standardized by comparing it with its expectation under an appropriate null reference distribution of the data. The estimate of the optimal number of clusters is then the value of k for which $logW_k$ falls the farthest below this reference curve.

   $Gap_n(k) = E_n^* log(W_k) - log(W_k)$

   where $E_n^*$ denotes expectation under a sample of size n from the reference distribution. Our estimate will be the value-maximizing $Gap_n(k)$ after considering the sampling distribution. Note that this estimate is very general, applicable to any clustering method and distance measure $d_{ij}$.

6. Elbow Method

   By fitting the model with a variety of values for "k," the elbow technique aids in selecting the best value for "k" (number of clusters). Distortion is determined by averaging the squared distances between each cluster's cluster centers. The Euclidean distance measure is

commonly utilized. Their inertia is the sum of samples' squared distances from the nearest cluster center.

The value of the k must be chosen at the "elbow," or the point at which the distortion/inertia starts dropping linearly, to calculate the ideal number of clusters. The concept is that a minimal SSE is required, but that as k is increased, the SSE tends to drop near 0. Therefore, our objective is to select a small value of k that still has a low SSE, and the elbow typically denotes the point at which the benefits of raising k begin to decrease.

### 2.3.1 K-Means Clustering

Kmeans is an iterative technique that divides the dataset into K unique non-overlapping clusters, each containing only one group to which each data point belongs. The clusters are kept as diverse (far) from one another as feasible while attempting to make the intra-cluster data points as comparable as possible. It distributes data points to clusters to minimize the sum of the squared distances between the data points and the cluster centroid, which is the average value of all the data points in the cluster. The homogeneity (similarity) of the data points within a cluster increases as the variance within the cluster decreases.

**Algorithm**:

1. The number of the cluster is K.
2. The dataset is shuffled, and then K data points are randomly chosen for centroids without replacement.
3. The Euclidean distance is calculated for each point in the dataset with the identified K points (cluster centroids).
4. Each data point is assigned to the closest centroid using the distance discovered in the previous step.
5. The average distance to the points is taken for each cluster to determine the new centroid.
6. This process is iterated till the centroids remain the same.

Expectation-Maximization is the method that kmeans uses to tackle the issue. The data points are assigned to the closest cluster in the E-step. The centroid of each cluster is calculated in the M-step.

$J = \sum_{i=1}^{m} \sum_{k=1}^{K} w_{ik} ||x^i - \mu_k||^2$

The algorithm is divided into 2 parts. First, J is minimized w.r.t $w_{ik}$ where $\mu_k$ is fixed. This leads to updating the cluster assignment (E-step). Then, J is minimized w.r.t $\mu_k$ and the centroids are updated (M-step).

E-step:

$$w_{ik} = \begin{cases} 1, & \text{if } k = argmin_j ||x^i - \mu_j||^2 \\ 0, & \text{otherwise} \end{cases}$$

M-step:

$\mu_k = \frac{\sum_{i=1}^{m} w_{ik} x^i}{\sum_{i=1}^{m} w_{ik}}$

The resulting clusters are all round in shape. This is due to the cluster centroids being updated iteratively using the mean value.

If clusters are spherically shaped, the Kmeans algorithm does a decent job capturing the data's structure. This suggests that K-means performs poorly at clustering the data when the clusters have complex geometric characteristics. It assumes that the clusters are made of similar size.

The K-means algorithm does not let data points far away from each other share the same cluster even though they belong to the same cluster.

K-Means also performs poorly when the data contains multivariate normal distributions with different means and standard deviations. Hence Gaussian Mixture Models are also considered.

### 2.3.2 Gaussian Mixture Model

A weighted sum of the component Gaussian component densities represents a Gaussian Mixture Model (GMM), a parametric probability density function representing a weighted sum of Gaussian component $(C_i, i = 1, 2, ..., k)$ densities. Each component generates data from a Gaussian distribution with mean $\mu_i$ and covariance matrix $\sum_i$. It gives the probability density function p(x) of data x as

$$p(x) = \sum_{i=1}^{k} w_i N(x|\mu_i, \sum_i)$$

for the whole data set $X = x_j(j = 1, 2, ..., N)$ the likelihood function will be

$$P(X|\omega, \mu, \sum) = \sum_{j=1}^{N}(\sum_{i=1}^{k} \omega_i N(x_j|\mu_i, \sum_i))$$

where $\omega_i$ stands for the weight of each component $C_i, \sum_i$ is the covariance matrix of the ith component and

$$\sum_{i=1}^{k} \omega_i = 1,$$

the distribution of each individual component $C_i$ is
$N(x|\mu_i, \sum_i) = \frac{1}{2\pi}\frac{1}{\sqrt{|\sum_i|}} exp\{-\frac{1}{2}(x - \mu_i)^T \sum_i{}^{-1}(x - \mu_i)\}$

The GMM method uses the EM algorithm to determine the parameters. Its iteration alternates between performing E steps and M steps. Assuming that on the $t$th iteration, the parameters or the estimates be

$$\theta_t = \{w_i(t), \mu_i(t), (i = 1, 2, ..., k)\}$$

the whole process can be described as follows.

i. The input parameter $\theta_{init}$, like k, $\mu, \sum$ is initialized. k is initialized. $\mu$ will be initialized by the parameter mention during the execution (random, in this case). 1/k is the value set for all components for $\omega$.

ii. Expectation step (E step): The expectation value of log-likelihood function with the parameters and the 'responsibility' of each Gaussian component $C_i$ is computed

$\tau_{ij}(t) = p(C_i|x_j, \theta_t) = \frac{p(x_j|C_i,\theta_t)p(C_i|\theta_t)}{p(x_j|\theta_t)}$

where,

$p(C_i|\theta_t) = \omega_i$

$p(x_j|\theta_t) = \sum_{i=1}^{k} p(x_j|C_i, \mu_i(t), \sum_i(t))\omega_i(t)$

GMM parameters are estimated from training data using the iterative Expectation-Maximization (EM) technique or Maximum A Posteriori (MAP) estimation using a suitably trained prior model.

iii. Maximize: Updating the parameters:

$\omega_i(t+1) = \frac{1}{N}\sum_{j=1}^{N} p(C_i|x_j, \theta_t)$
$\mu_i(t+1) = \frac{\sum_{j=1}^{N} p(C_i|x_j,\theta_t)x_j}{\sum_{j=1}^{N} p(C_i|x_j,\theta_t)} = \frac{1}{Nw_i(t+1)}\sum_{j=1}^{N} \tau_{ij}(t)x_j$

$$\sum_i(t+1) = \frac{\sum_{j=1}^{N} p(C_i|x_j, \theta_t)[x_j - \mu_i(t+1)][x_j - \mu_i(t+1)]^T}{\sum_{j=1}^{N} p(C_i|x_j, \theta_t)}$$
$$= \frac{\sum_{j=1}^{N} \tau_{ij}(t)[x_j - \mu_i(t+1)][x_j - \mu_i(t+1)]^T}{N\omega_i(t+1)}$$

iv. Step ii. and iii. are repeated until the total log-likelihood $log(P(X|\omega, \mu, \sum))$ converges.

# Chapter 3
# Metric Algorithm Benchmarking

Data sets containing 3 clusters were created using Multivariate Gaussian distribution. Analysis was done with 2D, 3D, and 4D data by varying the distance, standard deviation, and both simultaneously. Metric algorithms were applied to each step.
The results obtained are given in Fig. 3.1.

These graphs can be used as an assistance to determine the number of clusters when the actual data is analyzed. It can be useful to make appropriate decisions when different algorithms give different outcomes.

For 2D, equivalent performance for all metrics is observed for KMeans, and the DB index performs well in the case of GMM. Gap statistics for 3D Kmeans and AIC-BIC for GMM based perform well in all three scenarios. Finally, for 4D (sample case for multivariate data), Gap statistics and AIC-BIC perform well. All other KMeans metrics, as well as GMM metrics, perform well till a certain point. Hence consensus of all metrics can also be used for determining the number of clusters.

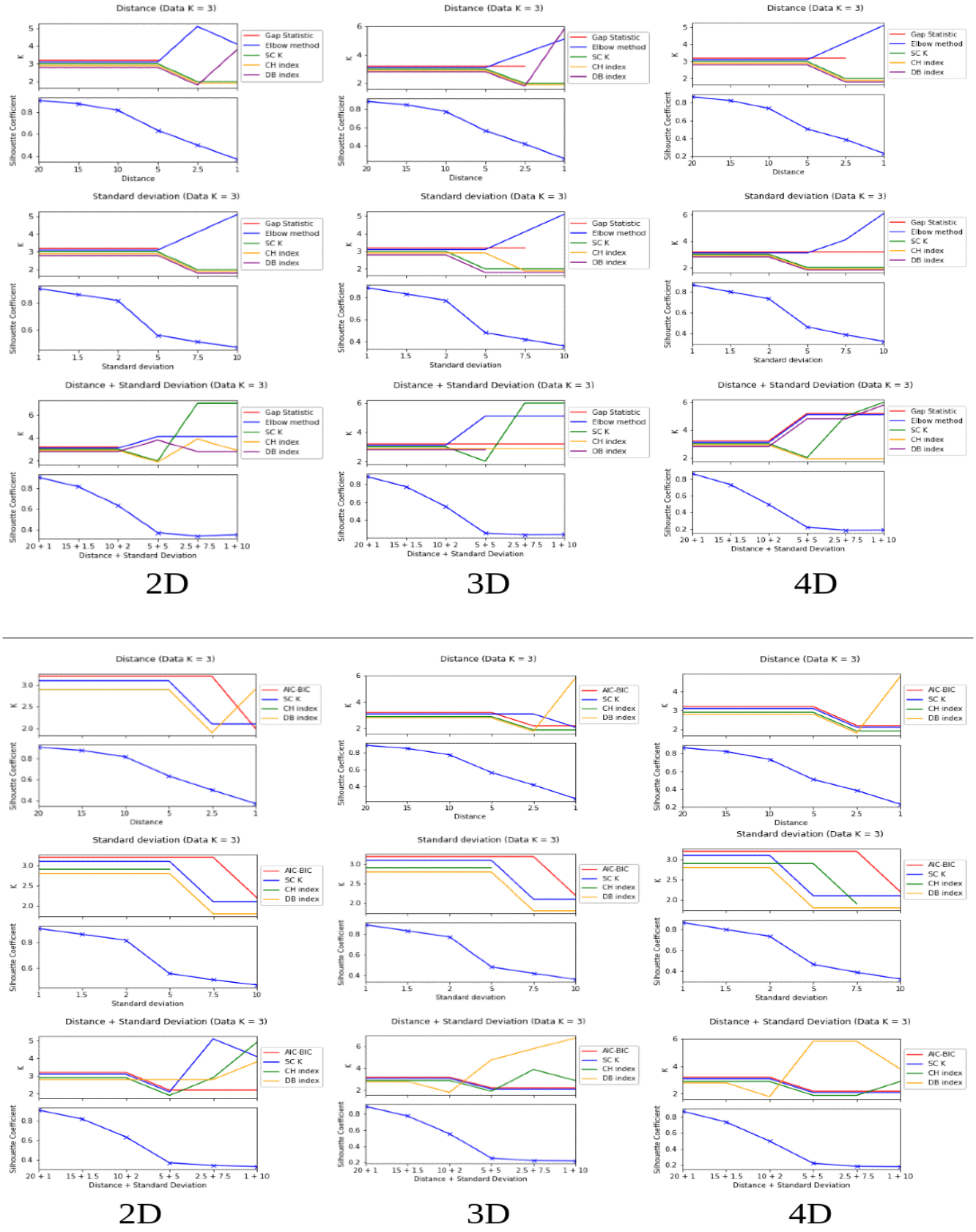


Figure 3.1: Test data results obtained by varying the Distance (Top Row), Standard Deviation (Middle Row) and both combined(Bottom Row). The upper fig. are results obtained from algorithms based on K-Means clustering while lower fig. are obtained from GMM. The single line plot (blue) below every graph show the values of Silhouette Coefficient. All the values are integers. Some lines are adjusted (+/- 0.1 or +/- 0.2) manually for better visuals.

# Chapter 4
# Fermi catalog analysis

Data was obtained from the Fermi catalog ([15], [16], [17], [18]). Data from the Fermi satellite was obtained from FERMIGBRST - Fermi GBM Burst Catalog (July 12, 2008 - January 18, 2022; 3214 GRBs). The data archive has several parameters, out of which a few are selected based on the past literature.

Note: All the log values mentioned in this study are with the base e, i.e., natural log.

## 4.1 Parameters

The "Band" function, a joint smoothly broken power-law function, has historically been used to characterize GRB spectra [19]. This function accurately captures the primary characteristics of the GRB spectra.

The following parameters are used in this work:

T90: The time period, measured in seconds, over which 90% of the burst fluence was accumulated. The time at which 5% of the total fluence was detected serves as the start of the T90 interval, and the time at which 95% of the total fluence was detected serves as the end of the T90 interval.

Peak flux: A single spectrum over the time range of the peak flux of the burst.

Fluence: A single spectrum over the entire burst duration selected by the duty scientist.

Flux(n): Peak flux in the timescale of n ms.

Epeak[1] : The peak energy of a Band function fit to a single spectrum over the duration of the burst, in keV.

Alpha[1]: The power law index, alpha, of a Band function fit to a single spectrum over the duration of the burst.

Beta[1]: The power law index, beta, of a Band function fit to a single spectrum over the duration of the burst.

Amplitude[1] : The amplitude of a Band function fit to a single spectrum over the duration of the burst, in photon/cm$^2$/s/keV.

The description of each parameter in detail has been given on the Catalog Website.

[1]Considered both values of parameters obtained from Band function when fitted to the spectra of total burst duration and the peak flux regions.

## 4.2 Method 1: Top to bottom

In this method, multiple parameters were initially considered, which might define the group characteristics, and then removing redundant parameters to get an optimized model.

Initially, T90, Fluence, Flux1024, Flux64, Fluence Band Amplitude, Fluence Band Epeak, Fluence Band Alpha, Fluence Band Beta, T50, Flux256, and Peak Flux Band Epeak were taken into consideration. These parameters were considered in the several past pieces of literature mentioned in Table 1 from [10]. Then a correlation plot (Fig. 4.1) was used to remove the dependent parameters to obtain independence in the data.

After the analysis, T90, Fluence, Fluence Band Amplitude, Fluence Band Epeak, Fluence Band Alpha, Fluence Band Beta, and Peak Flux Band Epeak were considered for further analysis. Fluence, Fluence Band Amplitude, Fluence Band Epeak, and Peak Flux Band Epeak have a wide range; hence, the natural log of these parameters is taken.

Without any further analysis, this data was taken as input to determine the number of clusters and classification. The results obtained are mentioned in Table 4.1, which infers that the number of clusters should be taken as 5. Using this information, the clustering was done and the properties of five classes are depicted in the violin plot (Fig. 4.2) and radar plot (Fig. 4.3).

A violin plot shows the probability density of the data at different values, usually smoothed by a kernel density estimator. Violin plots are plotted for each parameter. The distribution of GRBs for each class is shown for each parameter. The boxplot in the middle shows you the median (the small white dot in the middle), first quartile, third quartile, minimum and maximum. A radar chart is also plotted. It is a graphical approach to presenting multivariate data that takes the shape of a two-dimensional chart with three or more quantitative variables depicted on axes, all pointing in the same direction.

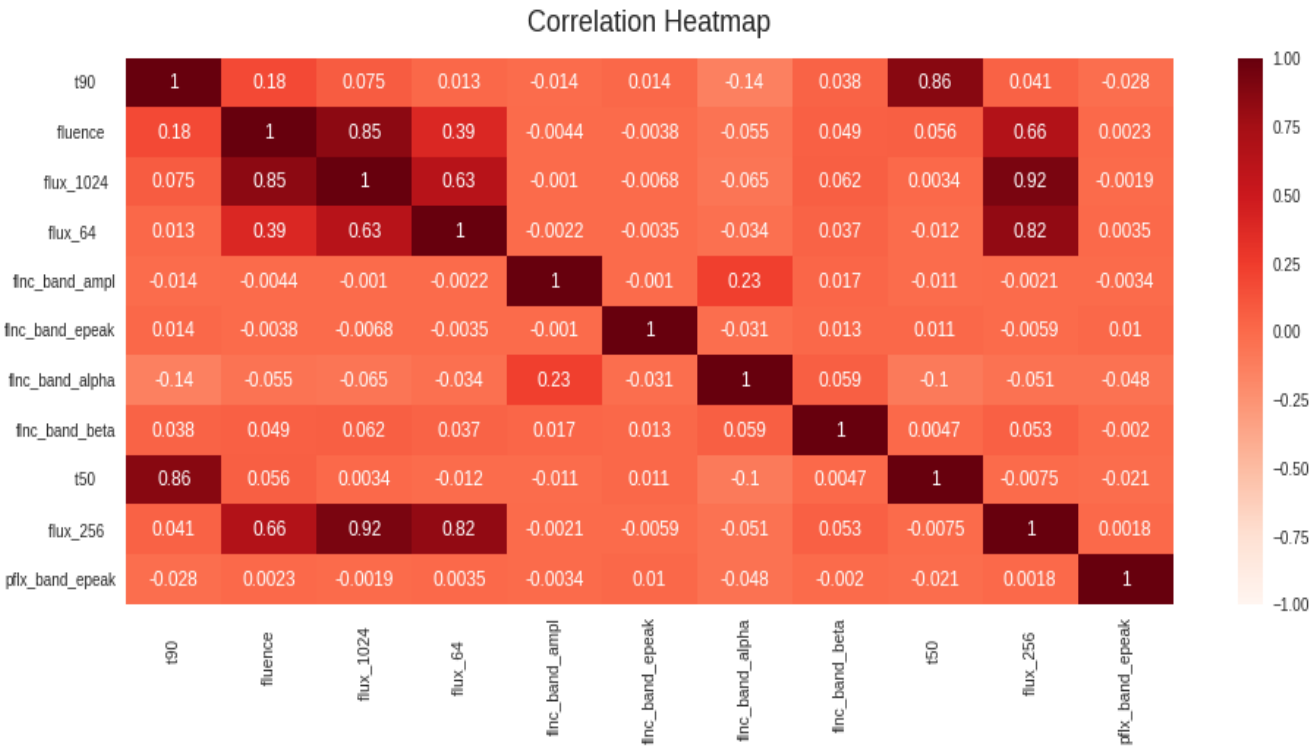


Figure 4.1: Correlation Map for Fermi Data Parameters

| **Algorithm** | **Metric** | **k (Number of Clusters)** |
|---|---|---|
| GMM | AIC-BIC | 5 |
| | Silhouette Score | NA ($< 0.6$) |
| | Calinski-Harabasz Index | 2 (next highest 5) |
| | Davies-Bouldin Index | 5 |
| Kmeans | Gap Statistic | 3 |
| | Elbow Method | 5 |
| | Silhouette Score | NA ($< 0.6$) |
| | Calinski-Harabasz Index | 2 |
| | Davies-Bouldin Index | 5 |

Table 4.1: Results obtained for number of clusters

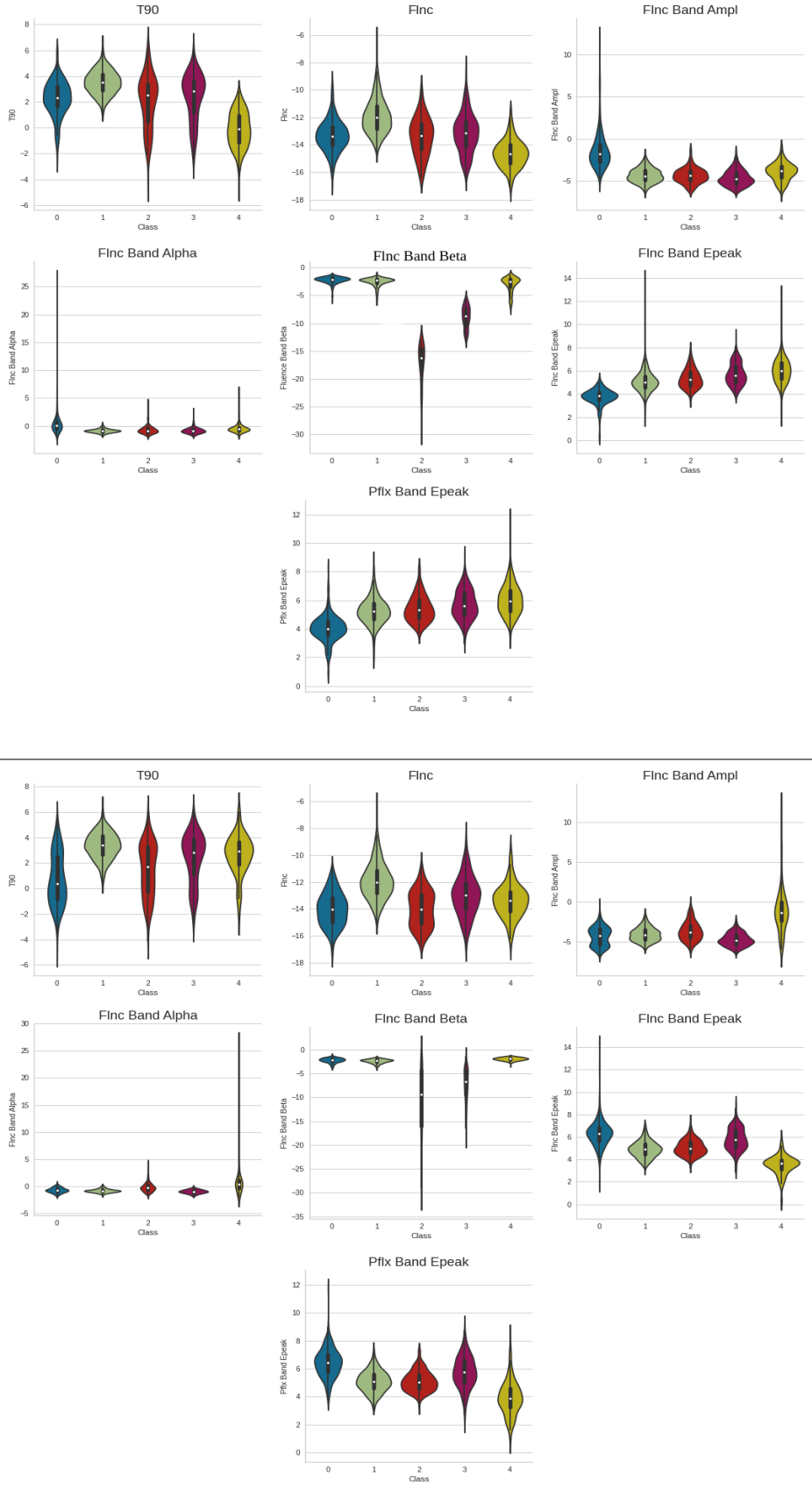


Figure 4.2: Violin plots showing the distribution various parameters for the classifications. The boxplot in the middle shows you the median (the small white dot in the middle), first quartile, third quartile, minimum and maximum. Properties of the classes of GRBs using Method 1. Top: Classification using K-Means Clustering. Bottom: Classification using GMM Clustering.

## 4.3 Method 2: Bottom to Top

In this method, instead of starting with multiple parameters, only a single parameter (T90) is taken into account, and then the parameters are added accordingly. Measures of model performance that consider model complexity are provided by the Akaike information criterion (AIC) and the Bayesian information criterion (BIC). In AIC and BIC, a term that measures how well the model fits the data is combined with a term that penalizes the model proportionately to the number of parameters. This way, the model with the lowest value is considered the least complex and provides a better fit for the data.

Table 4.2 shows the lowest AIC-BIC value of each model with parameters (1st column) and the number of clusters according to the AIC-BIC method. Log values of parameters - T90, Epeak, and Fluence are considered due to the high range. Outliers are removed using the Interquartile Range Rule. Four models are selected, three with the lowest AIC-BIC value with two parameters and one with three parameters (Models with Fluence parameters are considered over Peak flux as they show the same trend in AIC-BIC values). The algorithms listed in the previous section are used to determine the number of clusters, and the results are presented in Table 4.3.

The Davies-Bouldin Index is the most reliable GMM-based metric in 2D (data with two parameters). For K-Means, all the metrics have equivalent performance; hence the majority can be considered.

For the case of (T90, Alpha), the best metric for GMM is DB index and that gives k=2 and in K-means method all methods are equally good metrics, among them DB index also gives k=2 and therefore, number of clusters were chosen as 2. However, it should be noted that other metrics like AIC-BIC in GMM suggests k=3 and similarly methods like elbow method and Gap statistics under K-means also suggest k=3. Therefore, there is a possibility of a third cluster present in the data but it is currently ambiguous. This needs to be further investigated in the future.

Whereas, for 3D data, AIC-BIC for GMM-based and Gap Statistic for K-Means has better performance. (Fig. 3.1)

| AIC-BIC value | k<br>Band Parameter<br>(AIC/BIC) | k<br>Peak Flux Parameters<br>(AIC/BIC) |
|---|---|---|
| **T90, Fluence** | 3 (14160/14290) | 3 (14160/14290) |
| **T90, Alpha** | 3 (9210/9340) | 3 (10165/10294) |
| **T90, Epeak** | 2 (13329/13425) | 2 (13429/13525) |
| **T90, Alpha, Fluence** | 3 (15058/15278) | 3 (15991/16210) |
| **T90, Alpha, Epeak** | 3 (14043/14262) | 3 (14917/15136) |
| **T90, Epeak, Fluence** | 3 (19282/19503) | 3 (19345/19566) |
| **T90, Alpha, Epeak, Fluence** | 3 (19596/19927) | 3 (20427/20757) |

Table 4.2: Model Complexity measurement using AIC-BIC. (Note: Individual values have no interpretation, the values are relative.)

| | | **T90, Alpha** | **T90, Epeak** | **T90, Fluence** | **T90, Epeak, Alpha** |
|---|---|---|---|---|---|
| **GMM** | **AIC-BIC** | 3 | 2 | 3 | 3 |
| | **SC** | 2 | 2 | 2 | 2 |
| | **CH Index** | 5 | 2 | 2 | 2 |
| | **DB Index** | 2 | 2 | 2 | 2 |
| **KMeans** | **Gap Statistics** | 3 | 2 | 2 | 3 |
| | **Elbow Method** | 3 | 3 | 2 | 3 |
| | **SC** | 2 | 2 | 2 | 2 |
| | **CH Index** | 4 | 2 | 3 | 2 |
| | **DB Index** | 2 | 2 | 2 | 2 |

Table 4.3: Number of cluster given by metric for selected models

Hence,

Model 1 [log(T90), Fluence Band Alpha]: k = 2

Model 2 [log(T90), log(Fluence Band Epeak)]: k = 2

Model 3 [log(T90), log(Fluence)]: k = 2

Model 4 [log(T90), Fluence Band Alpha, log(Fluence Band Epeak)]: k = 3

have been considered for clustering.


Figure 4.3: Mean of properties of the classes of GRBs using Method 1. Left: Classification using K-Means Clustering. Right: Classification using GMM Clustering.

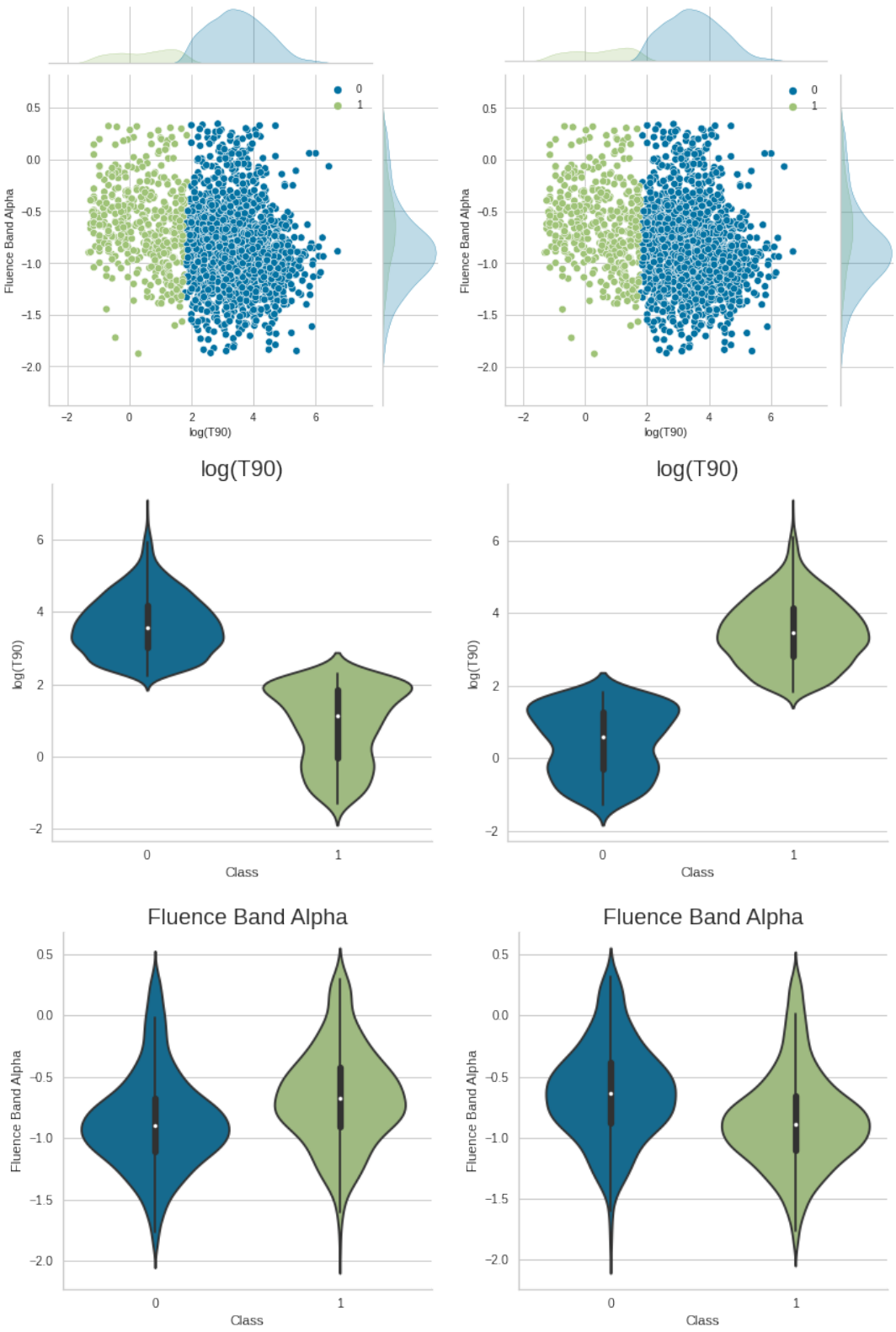


Figure 4.4: Model 1: Clustering and properties of the classes (violin plots) of GRBs using K-Means Clustering (left) and GMM (right) with the parameters, log(T90) and Fluence Band Alpha.

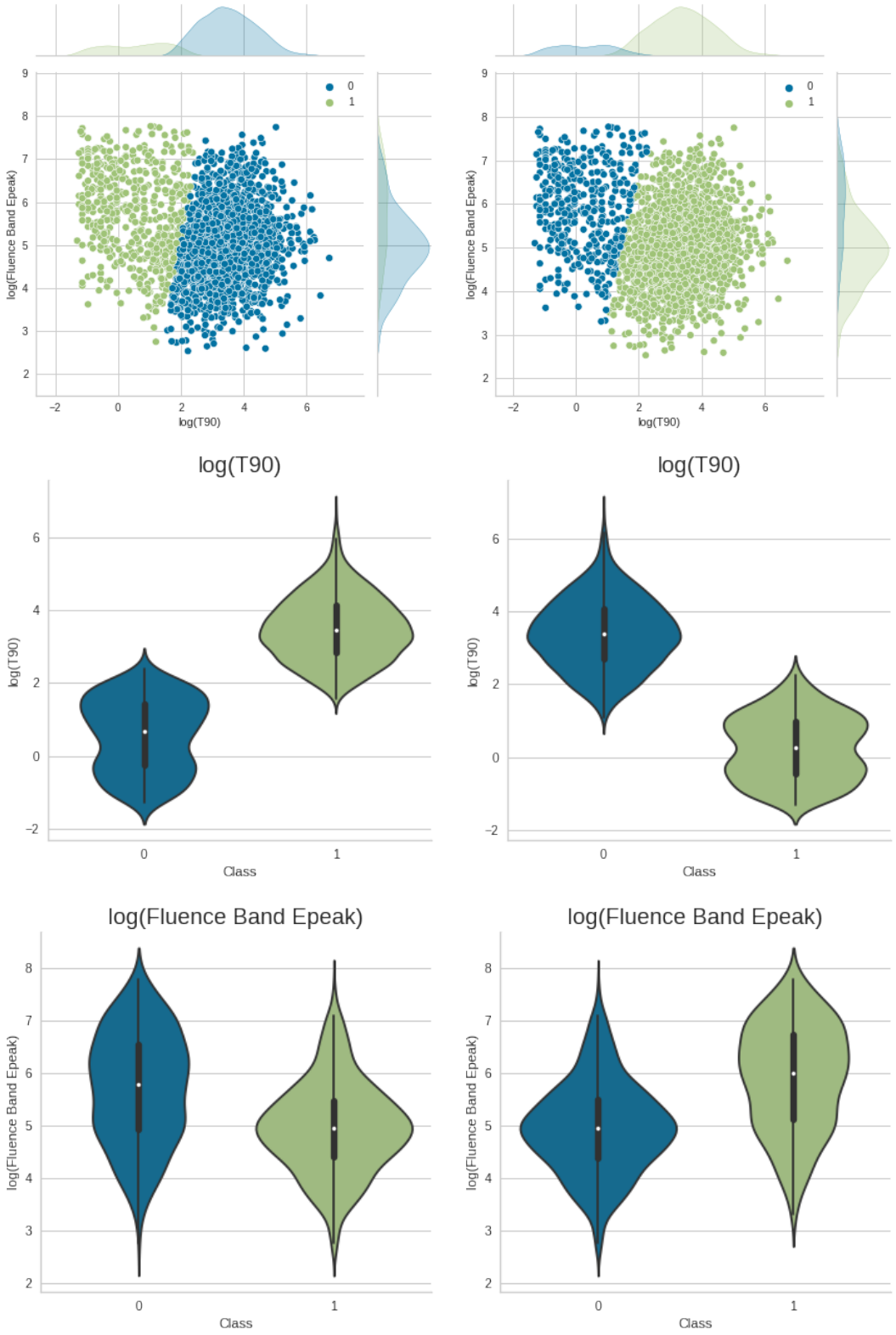


Figure 4.5: Model 2: Clustering and properties of the classes (violin plots) of GRBs using K-Means Clustering (left) and GMM (right) with the parameters, log(T90) and log(Fluence Band Epeak).

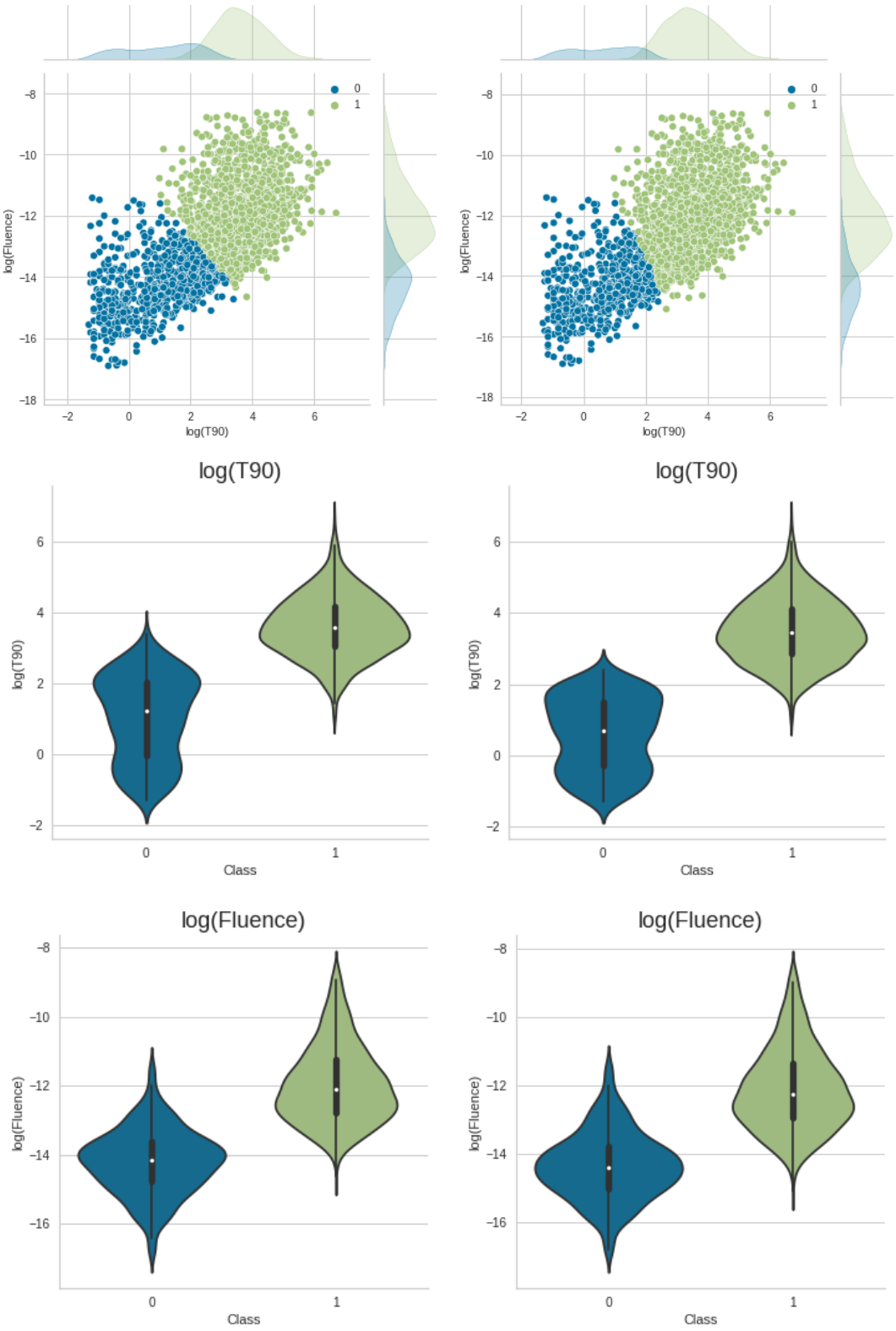


Figure 4.6: Model 3: Clustering and properties of the classes (violin plots) of GRBs using K-Means Clustering (left) and GMM (right) with the parameters, log(T90) and log(Fluence).

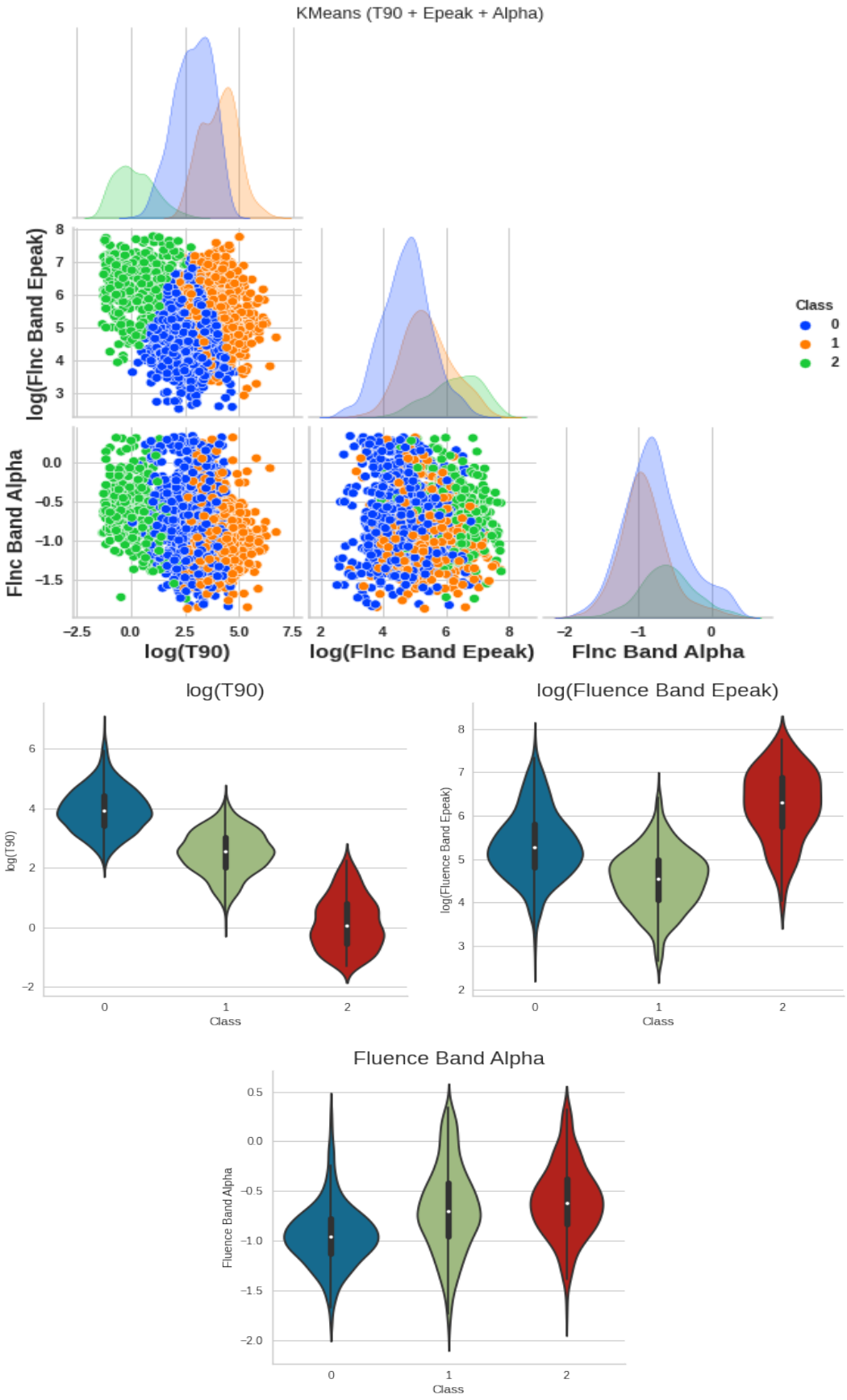


Figure 4.7: Model 4: Clustering and properties of the classes (violin plots) of GRBs using K-Means Clustering.

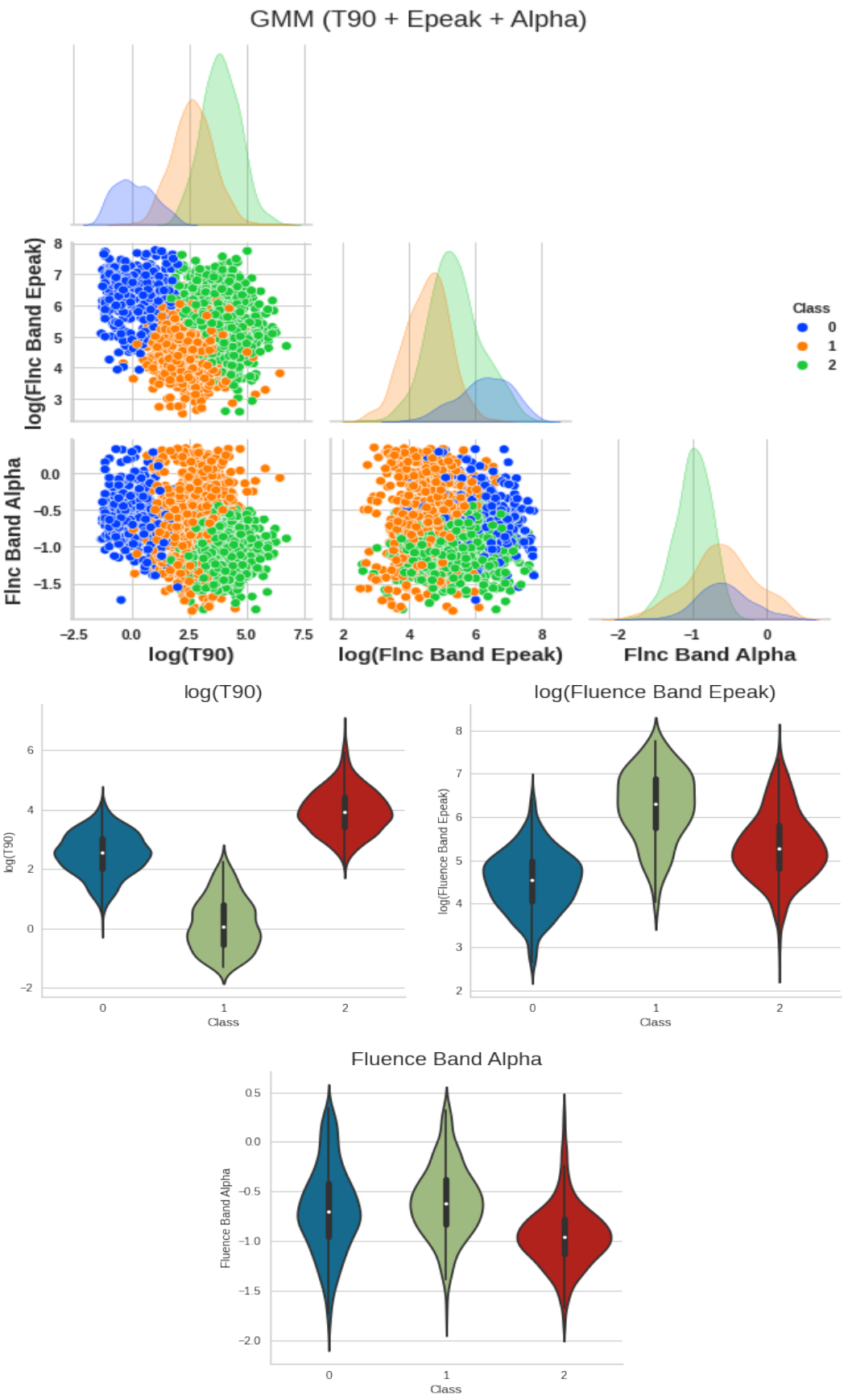


Figure 4.8: Model 4: Clustering and properties of the classes (violin plots) of GRBs using GMM Clustering.

# Chapter 5
# Results and Conclusions

In this study, initially, various metric algorithms based on Gaussian Mixture Modeling and K-Means clustering are tested for sample data which was generated using multivariate Gaussian in 2D, 3D and 4D. This gave the idea for selecting the number of clusters if the values obtained by metrics differ.

Method 1 (Top to Bottom): The data, after cleaning and removing dependencies, consisted of 7 observables (parameters). Using the metric algorithms, the number of clusters decided were five. Then the clustering was done using K-Means clustering and Gaussian Mixture Modeling.

Top to bottom method was considered due to complex behaviour observed in light curve of each GRB. Complex models can lead to a higher number of clusters. There can be two reasons; first, the nature of GRBs is complex, and hence it requires more parameters for categorization, which would eventually lead us to the answer of their progenitors. Second, it may be due to the overfitting of data. A detailed study of Top to Bottom approach with various parameters (not limited to parameters used in this study) can lead to interesting revelations.

Method 2 (Bottom to Top): Various parameters were added along with T90. The best set of parameters were chosen using AIC-BIC. The set of parameters with four lowest score were chosen for further analysis. (Here, AIC-BIC was used to select the best set of parameters.). The number of clusters for selected models were decided using the metric algorithms (Here, AIC-BIC, along with other metrics, was used to select the number of clusters).
Model 1 [log(T90), Fluence Band Alpha]: k = 2
Model 2 [log(T90), log(Fluence Band Epeak)]: k = 2
Model 3 [log(T90), log(Fluence)]: k = 2
Model 4 [log(T90), Fluence Band Alpha, log(Fluence Band Epeak)]: k = 3
The differences between models given by K-Means clustering and GMM might narrow down with more data.

Model 1 has the lowest AIC-BIC value among all models. It classifies GRBs into 2 classes which was decided using the metric algorithms. The best metric for GMM is DB index and that gives k=2 and in K-means method all algorithms are equally good metrics, among them DB index

also gives k=2 and therefore, number of clusters were chosen as 2. However, it should be noted that other metrics like AIC-BIC in GMM suggests k=3 and similarly methods like elbow method and Gap statistics under K-means also suggest k=3. Therefore, there is a possibility of a third cluster present in the data but it is currently ambiguous. This needs to be further investigated in the future.

Model 2 classifies GRBs into two classes—long-soft bursts and short-hard bursts. Here long/short - soft/hard indicates GRBs with long/short duration (T90) and low/high value of Epeak. This Model is in corroboration with the [20], which also classifies GRBs into long-soft GRBs and short-hard GRBs using statistical analysis. The result of getting almost 3/4 of the total GRBs (the total count is lower than actual data due to several missing values of parameters and removal of outliers) as long-soft GRBs is also consistent with [20]. In [20], the classification was made using T90 and Hardness Ratio (Epeak/Fluence).

Model 3 (and in Appendix C) shows a linear correlation between Fluence and T90, which was expected as Fluence is integrated flux over the entire duration. Hence, due to this correlation, this Model should not be considered as well.

Model 4 indicates three classes of GRBs. This Model is statistically less significant than other models and can be a result of overfitting, but further studies of this Model can be done, and classification using T90, Alpha, and Epeak can be an interesting topic for further studies. This model also provides an evidence for a 3rd Class stated by multiple studies in past.

Hence, the most suitable Model for the classification of GRBs, according to this study, is Model 1 (T90, Alpha). The properties of GRBs in this Model are described in Table 5.1. From the classification, the GRBs can be further studied in detail to determine the similarities within the class and differences between classes. This would aid in determining the source of these explosions.

| | **KMeans** | | | | **GMM** | | | |
|---|---|---|---|---|---|---|---|---|
| | Class 0 (1481) | | Class 1 (584) | | Class 0 (1637) | | Class 1 (428) | |
| | T90 | Alpha | T90 | Alpha | T90 | Alpha | T90 | Alpha |
| **Mean** | 55.711 | -0.863 | 3.797 | -0.659 | 51.157 | -0.852 | 2.294 | -0.625 |
| **Median** | 35.649 | -0.898 | 3.072 | -0.675 | 31.744 | -0.885 | 1.792 | -0.635 |
| **STD** | 64.100 | 0.393 | 2.956 | 0.395 | 62.564 | 0.394 | 1.759 | 0.392 |

Table 5.1: Statistics for classification with count in parenthesis.

# Appendix A
# Comparing Fermi and Swift Data

Fermi observes light in the energy range between 8-10 keV to greater than 300 GeV, while swift observes in 15–150 keV band. The Swift catalog data was obtained from Swift website (17th Dec, 2004 - 18th Jan, 2022). There are several GRBs that are observed by both satellites. Here, T90 and Fluence from both Swift and Fermi of the same GRBs are compared. It is observed that the measurement is different for both satellites. A best fit line (red) is drawn while comparing it to line y = x (black).

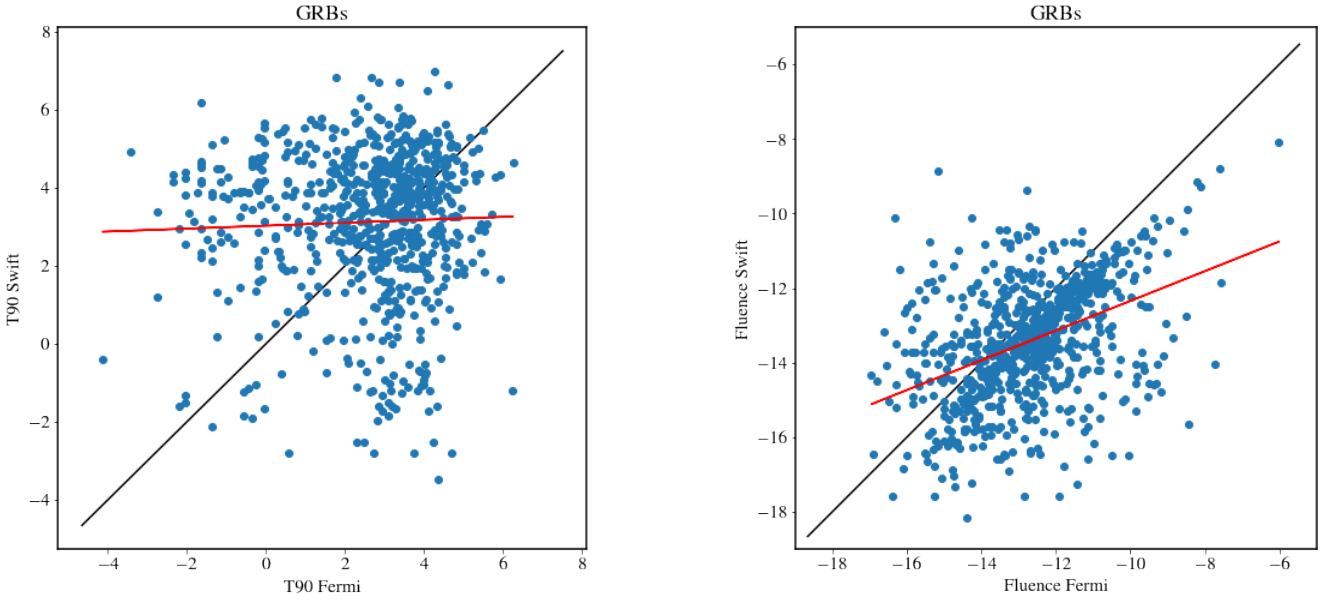


Figure A.1: Comparing T90(left) and Fluence(right) of the same GRBs observed by Fermi and Swift.

It is observed that neither the values of T90 nor Fluence are similar for the same GRB when observed from different satellites. This suggests that, using just T90 or fluence might not lead to right classification as these parameters cannot solely define a GRBs according to observable parameters.

# Appendix B
# T90: Astrosat

T90 was analyzed for Astrosat satellite. Data was obtained from Astrosat website (6th Oct, 2015 - 11th Jan, 2022) (T90 is the only parameter available for the Astrosat GRB data.) The bimodal classification from [1] is obtained.

| Algorithm | Metric | k (Number of Clusters) |
|---|---|---|
| GMM | AIC-BIC | 2 |
| | Silhouette Score | 2 (0.655) |
| | Calinski-Harabasz Index | NA (Increasing Linearly) |
| | Davies-Bouldin Index | 2 |
| KMeans | Gap Statistic | NA |
| | Elbow Method | 3 |
| | Silhouette Score | 2 (0.64) |
| | Calinski-Harabasz Index | NA (Increasing Linearly) |
| | Davies-Bouldin Index | 2 |

Table B.1: Results obtained for number of clusters

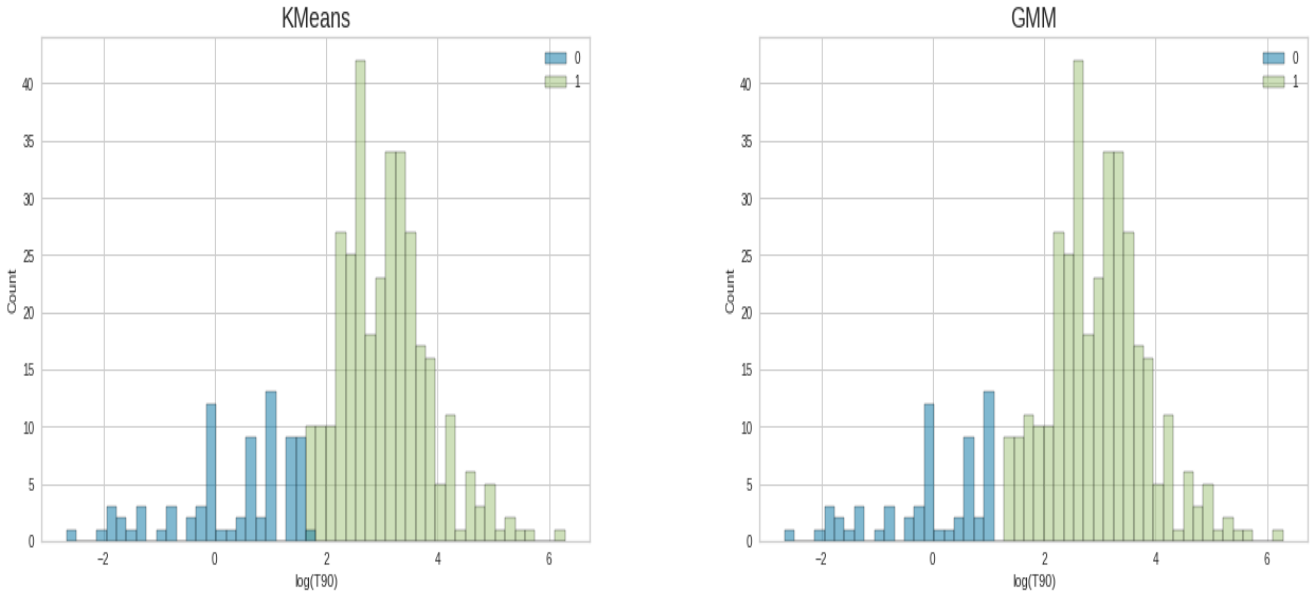


Figure B.1: Clustering of Astrosat data (T90) using KMeans(left) and GMM(right).

| T90 | KMeans | | GMM | |
|---|---|---|---|---|
| | Class 0 | Class 1 | Class 0 | Class 1 |
| **Count** | 60 | 349 | 79 | 330 |
| **Mean** | 1.434 | 30.431 | 2.168 | 31.925 |
| **Median** | 1 | 20 | 2 | 21.5 |
| **STD** | 1.025 | 43.138 | 1.608 | 43.9 |

Table B.2: Statistics of Classification

# Appendix C
# T90 + Fluence : Swift

Data was analyzed for Swift satellite using the parameters T90 and Fluence. (Note: Swift data only includes T90 and Fluence of GRBs).

| Algorithm | Metric | k (Number of Clusters) |
|---|---|---|
| GMM | AIC-BIC | 2 |
| | Silhouette Score | NA (< 0.6) |
| | Calinski-Harabasz Index | 3 |
| | Davies-Bouldin Index | 2 |
| KMeans | Gap Statistic | 2 |
| | Elbow Method | 3 |
| | Silhouette Score | NA (< 0.6) |
| | Calinski-Harabasz Index | 3 |
| | Davies-Bouldin Index | 2 |

Table C.1: Results obtained for number of clusters

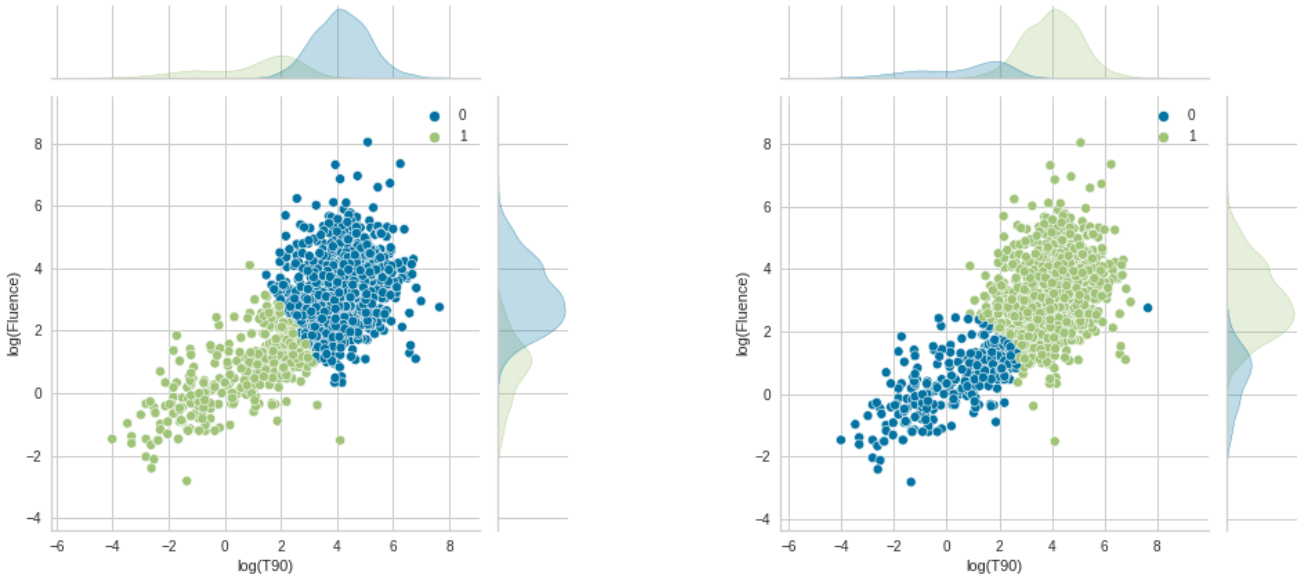


Figure C.1: Clustering of Swift data (T90 and Fluence) using KMeans(left) and GMM(right).

| | KMeans | | | | GMM | | | |
|---|---|---|---|---|---|---|---|---|
| | Class 0 (1004) | | Class 1 (398) | | Class 0 (310) | | Class 1 (1092) | |
| | T90 | Fluence | T90 | Fluence | T90 | Fluence | T90 | Fluence |
| **Mean** | 101.209 | 51.566 | 6.190 | 3.672 | 10.876 | 2.530 | 92.221 | 48.030 |
| **Median** | 63.8 | 21 | 4.8 | 2.53 | 3.25 | 2 | 56 | 19 |
| **STD** | 133.038 | 140.960 | 6.362 | 4.569 | 119.100 | 2.230 | 135.699 | 114.730 |

Table C.2: Statistics for classification with count in parenthesis.

# Bibliography


[1] C. Kouveliotou, C. A. Meegan, G. J. Fishman, N. P. Bhat, M. S. Briggs, T. M. Koshut, W. S. Paciesas, and G. N. Pendleton, "Identification of two classes of gamma-ray bursts," *The Astrophysical Journal*, vol. 413, p. L101, Aug. 1993.

[2] A. I. MacFadyen and S. E. Woosley, "Collapsars: Gamma-ray bursts and explosions in "failed supernovae"," *The Astrophysical Journal*, vol. 524, pp. 262–289, oct 1999.

[3] D. A. Perley, Y. Niino, N. R. Tanvir, S. D. Vergani, and J. P. U. Fynbo, "Long-duration gamma-ray burst host galaxies in emission and absorption," *Space Science Reviews*, vol. 202, pp. 111–142, mar 2016.

[4] Z. Cano, S.-Q. Wang, Z.-G. Dai, and X.-F. Wu, "The observer's guide to the gamma-ray burst supernova connection," *Advances in Astronomy*, vol. 2017, pp. 1–41, 2017.

[5] D. Eichler, M. Livio, T. Piran, and D. N. Schramm, "Nucleosynthesis, neutrino bursts and -rays from coalescing neutron stars," *Nature*, vol. 340, pp. 126–128, jul 1989.

[6] R. Narayan, B. Paczynski, and T. Piran, "Gamma-ray bursts as the death throes of massive binary stars," *The Astrophysical Journal*, vol. 395, p. L83, aug 1992.

[7] T. Ahumada and L. P. Singer, "Discovery and confirmation of the shortest gamma-ray burst from a collapsar," *Nature Astronomy*, vol. 5, pp. 917–927, July 2021.

[8] B. Zhang, B.-B. Zhang, E.-W. Liang, N. Gehrels, D. N. Burrows, and P. Mészáros, "Making a short gamma-ray burst from a long one: Implications for the nature of GRB 060614," *The Astrophysical Journal*, vol. 655, pp. L25–L28, jan 2007.

[9] B. Zhang, B.-B. Zhang, F. J. Virgili, E.-W. Liang, D. A. Kann, X.-F. Wu, D. Proga, H.-J. Lv, K. Toma, P. Mészáros, D. N. Burrows, P. W. A. Roming, and N. Gehrels, "DISCERNING THE PHYSICAL ORIGINS OF COSMOLOGICAL GAMMA-RAY BURSTS BASED ON MULTIPLE OBSERVATIONAL CRITERIA: THE CASES OFiz/i= 6.7 GRB 080913,iz/i= 8.2 GRB 090423, AND SOME SHORT/HARD GRBs," *The Astrophysical Journal*, vol. 703, pp. 1696–1724, sep 2009.

[10] L. Salmon, L. Hanlon, and A. Martin-Carrillo, "Two dimensional clustering of swift/BAT and fermi/GBM gamma-ray bursts," *Galaxies*, vol. 10, p. 77, jun 2022.

[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[12] T. Calinski and J. Harabasz, "A dendrite method for cluster analysis," *Communications in Statistics - Theory and Methods*, vol. 3, no. 1, pp. 1–27, 1974.

[13] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, pp. 224–227, apr 1979.

[14] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 2, pp. 411–423, 2001.

[15] D. Gruber and A. Goldstein, "The fermi gbm gamma-ray burst spectral catalog: Four years of data," , vol. 211, p. 12, Mar. 2014.

[16] P. Narayana Bhat and C. A. Meegan, "The third fermi gbm gamma-ray burst catalog: The first six years," , vol. 223, p. 28, Apr. 2016.

[17] A. von Kienlin and C. A. Meegan, "The second fermi gbm gamma-ray burst catalog: The first four years," , vol. 211, p. 13, Mar. 2014.

[18] A. von Kienlin and C. A. Meegan, "The fourth fermi-gbm gamma-ray burst catalog: A decade of data," , vol. 893, p. 46, Apr. 2020.

[19] D. Band, J. Matteson, L. Ford, B. Schaefer, D. Palmer, B. Teegarden, T. Cline, M. Briggs, W. Paciesas, G. Pendleton, G. Fishman, C. Kouveliotou, C. Meegan, R. Wilson, and P. Lestrade, "BATSE Observations of Gamma-Ray Burst Spectra. I. Spectral Diversity," , vol. 413, p. 281, Aug. 1993.

[20] F.-W. Zhang, L. Shao, J.-Z. Yan, and D.-M. Wei, "Revisiting the Long/Soft-Short/Hard Classification of Gamma-Ray Bursts in the Fermi Era," , vol. 750, p. 88, May 2012.