

# Music Genre Classification Using Machine Learning on Free music archive (FMA) Dataset

# Student Name: Shashank Bellachikkanahalli Venkatareddy

**Student No: R00224414**

**Supervisor:** Vincent Cregan

**For the module DATA9003 – Research Project as part of the**

**Master of Science in Data Science and Analytics, Department of Mathematics, 25-08-2023**

**Declaration of Authorship**

I, Shashank Bellachikkanahalli Venkatareddy, declare that this thesis titled ‘Music Genre Classification Using Machine learning on Free Music Archive (FMA) Dataset’ and the work presented in it are my own. I confirm that,

* This work was done wholly or mainly while in candidature for the Masters’ degree at Munster Technological University
* Where any part of this thesis has previously been submitted for a degree or any other qualification at Munster Technological University or any other institution, this has been clearly stated
* Where I have consulted the published work of others, this is always clearly attributed
* Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this project report is entirely my own work
* I have acknowledged all main sources of help
* I understand that my project documentation may be stored in the library at MTU, and may be referenced by others in the future

Signed: shashank b v

Date: 25-08-2023

**Table of Contents**

1 Introduction ................................................................................................................. 2

1.1 Background ............................................................................................................ 2

1.2 Problem Statement ................................................................................................ 3

1.3 Aims ........................................................................................................................ 5

2 Literature Survey .......................................................................................................... 5

2.1 Evolution of Music Genre Classification ................................................................. 5

2.2 Feature Extraction .................................................................................................. 7

2.3 Machine Learning in Music Genre Classification .................................................... 9

2.4 Deep Learning in Music Genre Classification ......................................................... 10

2.5 The Free Music Archive (FMA) Dataset .................................................................. 12

3 Methodology ................................................................................................................ 14

3.1 Data Collection ....................................................................................................... 14

3.1.1 Tracks Data .......................................................................................................15

3.1.2 Album Data ...................................................................................................... 18

3.1.3 Artist Data ........................................................................................................ 20

3.1.4 Genre Data ........................................................................................................22

3.2 Data Cleaning ...........................................................................................................24

3.3 Exploratory Data Analysis (EDA) ..............................................................................29

3.3.1 Statistical Summary ...........................................................................................31

3.4 Feature Extraction ....................................................................................................38

3.4.1 Mel-Frequency Cepstral Coefficients (MFCCs) ..................................................39

3.4.2 Chroma Features ...............................................................................................40

3.5 Overview and Extracted Features ............................................................................41

3.6 Machine Learning Models ........................................................................................43

3.7 Model Evaluation .....................................................................................................47

3.7.1 Methods ............................................................................................................47

3.7.2 Feature Sets and Performance Metrics .............................................................48

4 Results ............................................................................................................................49

4.1 Accuracy ................................................................................................................49

4.2 Computation Time ................................................................................................ 51

4.3 Evaluation Results ................................................................................................. 52

5 Discussion and Conclusion .......................................................................................... 54

5.1 Future Work ...........................................................................................................54

**List of Figures**

Figure 1.1: Popularity of Genres ..........................................................................................2

Figure 3.1.1: Comprehensive Overview of Track Attributes ...............................................16

Figure 3.1.2: Album Metadata ............................................................................................18

Figure 3.1.3: Comprehensive Overview of Artist Metadata ................................................22

Figure 3.1.4: Comprehensive Overview of Genre Metadata ...............................................23

Figure 3.2.1: Summary of the Data Cleaning .......................................................................28

Fig 3.3.1: Growth and Integrity of Tracks in the FMA Dataset .......................................30

Figure 3.3.2: Distribution of Track Durations in the FMA Dataset .......................................30

Figure 3.3.3: Distribution of Bit Rates in the FMA Dataset ..................................................31

Figure 3.3.4: Distribution of Tags Across Tracks, Albums, and Artists .................................32

Figure 3.3.5: Distribution of Tags Across Tracks, Albums, and Artists .................................33

Figure 3.3.6: Annual Trends: Track Recordings vs. Album Releases ....................................33

Figure 3.3.7: Artists' Career Timeline: Start vs. End Years ...................................................34

Figure 3.3.8: Annual Album Releases: 1990-2017 ...............................................................34

Figure 3.3.9: Annual Album Releases: 1990-2017 ...............................................................35

Figure 3.3.10: Distribution of Artists Across Top-Level Genres ...........................................36

Figure 3.3.12: Frequency Distribution of Genre Tags Per Track ..........................................37

Figure 3.3.13: Number of Tracks Across Top-Level Genres .................................................38

Figure 3.5.5: MFCC Feature Extraction ................................................................................42

Figure 4.1.1: Accuracy of Different Machine Learning Algorithms ......................................50

Figure 4.2.1: Time Taken for Each Model to Execute ...........................................................51

# List of Tables

# Table 3.3.2: Statistical Summary of the Data ................................................................ 31

# Table 3.3.11: Distribution of Top-Level Genres in the Dataset ......................................37

# Table 4.3.1: Evaluation Table of Models ........................................................................53

# Abstract

Music is a universal form of expression with a multitude of genres that resonate with diverse audiences. While genre classification may seem straightforward to the human ear, automating this process poses a complex challenge. This complexity stems from the subtle and intricate characteristics that differentiate one musical genre from another. Effective categorization not only has implications for how music is organized and recommended in digital platforms but can also provide insights into the underlying structure and semantics of musical compositions. To tackle this issue, we aim to utilize machine learning and deep learning techniques to automatically categorize music into genres.

The objective of the project is to utilize machine learning and deep learning techniques to categorize music into genres. To achieve this, we use the Free music Archive (FMA) dataset, which consists of more than 100,000 tracks spanning across 161 music genres.

The methodology involves stages with a focus on extracting key features from the audio files. Two crucial features that will be extracted are Mel coefficients (MFCCs) and Chroma features. MFCCs represent the short-term power spectrum of sound and are commonly used in processing to describe its shape. MFCCs capture the shape of a sound's short-term power spectrum, representing how energy is distributed across different frequencies, they imitate how humans perceive sound and capture characteristics like timbre. On the other hand, Chroma features correspond to the twelve distinct pitch classes in Western music theory and serve as a tool for describing harmonic structures and chord progressions. Consequently, they are particularly relevant for tasks such as musical genre classification.

We explore the efficacy of various machine learning and deep learning algorithms in the analysis of audio signals. Specifically, we will focus on Support Vector Machines (SVMs), Random Forests, Convolutional Neural Networks (CNNs), and Recurrent Neural Networks (RNNs). These models will be trained and tested using extracted features such as MFCCs and Chroma features. By employing these feature sets, we will investigate how well each model performs in tasks that could range from speech recognition to music genre classification. This comparative approach is designed to inform on the strengths and weaknesses of each algorithm, thereby guiding future research and applications in the field of audio signal processing.

# 1. Introduction

## 1.1 Background

In the contemporary digital era, the proliferation of music content has transitioned from traditional physical media such as vinyl records and CDs to expansive online streaming platforms. This shift not only signifies the cultural importance of music but also presents a growing complexity in organizing and categorizing a broad array of genres and styles. According to various data analytics reports, the sheer volume and diversity of accessible online music have increased exponentially, making the classification of this content a pressing computational challenge.

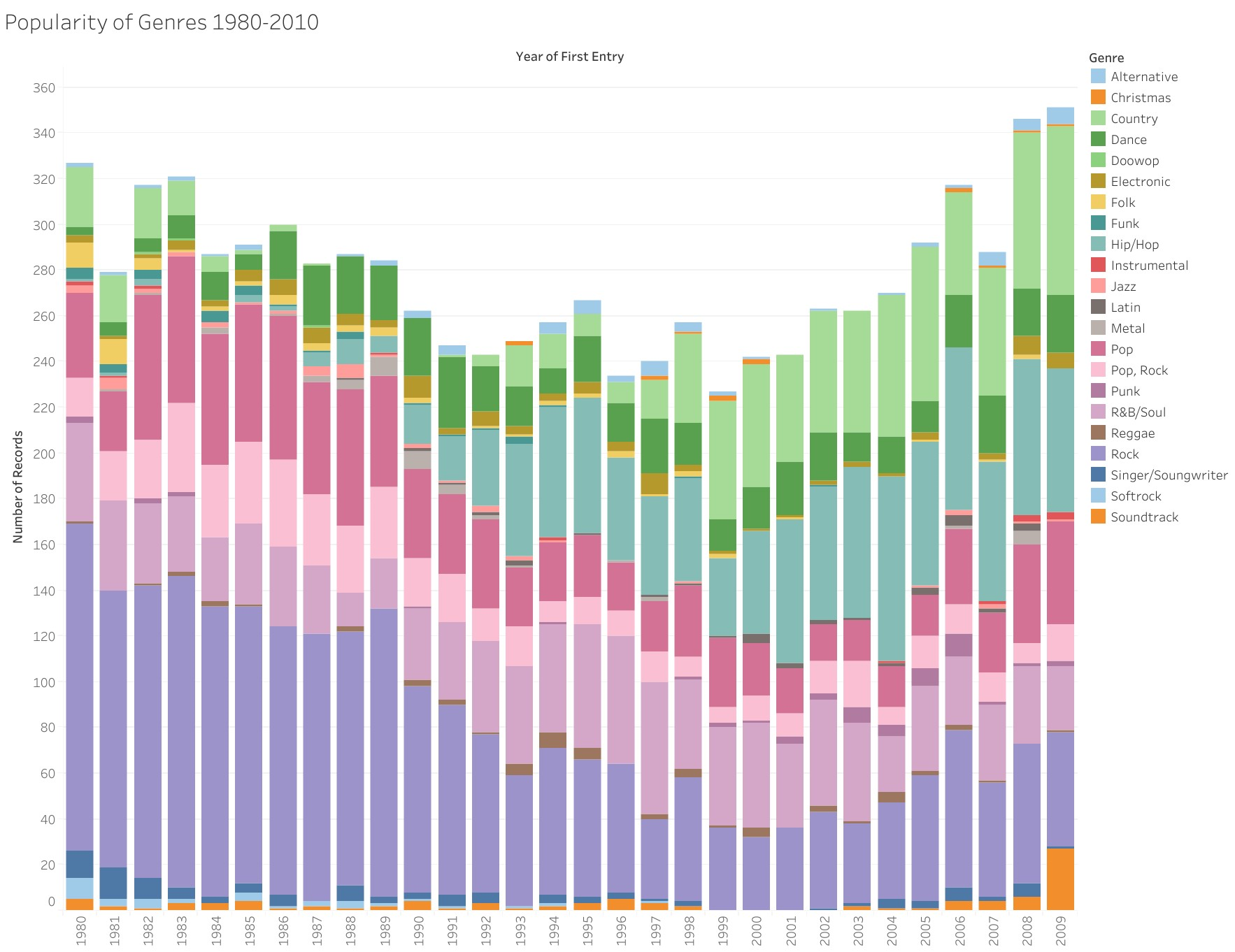


Fig 1.1: popularity of genres

The bar graph Figure 1.1 shows genre trends in the Billboard Top 100 from 1980 to 2009. It reveals key shifts, such as the decline of rock and the ascent of hip-hop and country music. The data hits a low in 1996, marking a dip in genre diversity, but then expands steadily until 2009, which peaks with 351 distinct records. Country dominates in 2009 with 74 tracks, followed by hip-hop with 63. The least diverse year was 1999, featuring Country and R&B/Soul as top genres. Over these three decades, R&B/Soul remained a stable presence, averaging 37 charting songs annually.

The inherent complexities of music—comprising rhythm, melody, timbre, and harmony—demand sophisticated methods for accurate classification and understanding. Due to the subjective nature of musical appreciation, where interpretations can diverge widely among individual listeners, conventional classification methods are insufficient such as simpler, rule-based or heuristic approaches. Therefore, computational models are required to address these obstacles, offering a systematic approach to classification and comprehension.

Such computational models propose solutions that integrate insights from both artistic and scientific disciplines. From an artistic perspective, understanding musical constructs like melody and rhythm contributes to genre specificity. Scientifically, algorithms based on machine learning and audio signal processing can analyze these constructs quantitatively, providing an objective basis for classification.

In summary, the evolution of music distribution platforms necessitates advanced computational methods for the classification and comprehension of diverse musical content. These methods are imperative to address the high intra-class variability and low inter-class variability challenges posed by the ever-expanding music corpus. By integrating both artistic and scientific approaches, these computational models aim to provide a nuanced yet systematic framework for music classification.

## 1.2 Problem Statement

Music genre classification presents a complex computational challenge, requiring the analysis and categorization of intricate auditory signals. Traditional manual methods of categorization have become impractical due to the burgeoning volume of music data, particularly in the realm of online streaming platforms and digital libraries. This has catalyzed a growing focus on automated classification techniques.

Music genre classification has become increasingly challenging due to the complexity of auditory signals and the immense growth of music data, especially with the advent of online streaming platforms and digital libraries. Traditional manual methods for categorization are less effective, thereby requiring more sophisticated approaches [1]. This has led to an increased emphasis on automated classification techniques, which are capable of handling a large dataset with diverse genres [2].

In light of these developments, our study aims to build upon prior research by employing machine learning and deep learning algorithms, including Support Vector Machines (SVMs), Random Forests, Convolutional Neural Networks (CNNs), and Recurrent Neural Networks (RNNs). Specifically, we will focus on extracting key audio features such as Mel Frequency Cepstral Coefficients (MFCCs) and Chroma features. These features play an essential role in capturing the intricacies of sound, thereby facilitating more accurate genre classification [1 , 2]. Our research methodology will include rigorous experimentation using standard cross-validation techniques to validate the models' performance and their generalizability across multiple genres. Additionally, we will explore the real-world applications of these models, such as in song recommendation systems and in the categorization of extensive music libraries, while considering aspects of scalability and computational efficiency.

This study employs machine learning and deep learning algorithms, with a particular emphasis on extracting key audio features such as Mel Frequency Cepstral Coefficients (MFCCs) and Chroma features. MFCCs quantify the energy distribution across different frequency bands and are instrumental for capturing sound characteristics like timbre. Chroma features offer insights into the harmonic content, including chords and their progressions. These features facilitate the accurate classification of musical genres and build upon prior research by introducing innovative methodologies.

The research designs and tests various algorithms including Support Vector Machines (SVMs), Random Forests, Convolutional Neural Networks (CNNs), and Recurrent Neural Networks (RNNs). Rigorous experimentation will be conducted using standard cross-validation techniques to validate the models' performance and ensure their generalization across diverse musical genres. The study explores the practical utility of these models in real-world contexts such as song recommendation systems and categorization of expansive music libraries. Consideration will also be given to scalability and computational efficiency, especially in real-time applications.

## 1.3 Aims

## The main objective of this research is to create a Predictive model, for classifying music genres by utilizing machine learning and deep learning methods. The specific objectives of this research include:

Data cleaning/EDA: The aim is to delve into the dataset that consists of more than 100,000 tracks spread over 161 different music genres. This involves a detailed examination of both metadata and various musical attributes, shedding light on the intricacies of the dataset.

Extracting Crucial Aspects from Audio Files: we'll employ machine learning and deep learning to analyze key audio features. Specifically, MFCCs will help us understand frequency distributions, while Chroma features will shed light on harmonic structures. These features are crucial for our genre classification models.

Developing and Comparing Machine Learning and Deep Learning Models: Developing models such as Support Vector Machines, Random Forests, Decision Tress Classifier, Naïve bayes classifier. This will involve experimentation and validation using standard cross-validation techniques to ensure generalization across different genres

Exploring Practical Applications Exploring the practical applications of the created models, including song recommendations and the organization of extensive music libraries.

## 2 Literature Survey

# With the growth, in music availability it is necessary to have systems and approaches that can accurately and efficiently categorize music into different genres. This review aims to explore Five aspects of the literature, Feature Extraction in Music, Machine Learning and Music Genre Classification, Deep Learning in Music Classification, Evolution of Music Genre Classification, Assessing the scope and complexity of the FMA dataset and its challenges for genre classification.

## 2.1 Evolution of Music Genre Classification

The classification of music genres has been a subject of study, In the 2000s there was an increase in attempts to use algorithms to understand the subtle differences, between music genres focusing mainly on analyzing audio features [1].

Tzanetakis and Cook's research ushered in a new era of music genre classification. By advocating the usage of texture, pitch, and rhythm content, Their system showcased the importance of efficient feature extraction from audio signals. Their work prompted several subsequent studies to explore the vast realm of audio features.

One significant progression was the popularization of (MFCCs). Logan's seminal work in 2000 explained the potential of MFCCs, primarily used in speech recognition, for music modeling [2]. These coefficients, representing the short-term power spectrum of sound, soon became the cornerstone of many music classification systems.

Rhythm patterns, in tandem with MFCCs, also gained traction. The temporal structure of a music piece, as McKinney and Breebaart highlighted, is instrumental in genre differentiation [3]. Their research underscored the rhythmic patterns intrinsic to genres, explained the periodic beat structures in genres like jazz and blues.

The role of timbre in music genre classification cannot be understated. While rhythm and pitch were frequent points of focus, Aucouturier and Pachet, delved into timbre textures. They ascertained the value of timbre in distinguishing genres, revealing nuances often overlooked by other features [4].

Moreover, high-dimensional data handling became an challenge in music genre classification. While the richness of audio data was an asset, its complexity posed substantial challenges. Techniques like Principal Component Analysis (PCA) provided a lifeline, as noted by Vignoli and Pauws [5]. PCA simplified complex datasets without compromising the data's inherent essence.

Mandel and Ellis, in 2005, introduced a comprehensive study on song classification using a large set of features, including spectral contrast and chroma [6]. Their ensemble methods approach, utilizing both classifiers and regressors, offered fresh insights into the intricacies of music classification.

Meyer and Wiering's conducted an in-depth analysis into the essence of feature extraction, offering novel techniques to better capture the soul of a music piece [7]. Their strategy involved exploring advanced musical attributes to enrich the feature set used for classification tasks.

Another significant research work was from Tzanetakis, Ermolinskyi , where the horizon of genre classification by proposing a multi-feature audio segmentation technique [8]. By dividing songs into homogeneous segments, they could better discern the genre-defining elements of each segment.

In 2007 witnessed a surge in methods leveraging temporal features. Peeters's comprehensive study covered a gamut of audio features, the importance of temporal aspects in genre classification [9]. The late 2000s also saw the infusion of machine learning techniques. Mierswa and Morik's exploration adopted Support Vector Machines for genre classification, spotlighting their potential to model complex audio datasets [10].

## 2.2 Feature Extraction

Feature extraction plays a role, in the field of music information retrieval (MIR) for tasks like categorizing genres detecting moods and recognizing artists. The quality and characteristics of the extracted features have an impact on the effectiveness of machine learning models for these tasks. This review explores in feature extraction, exploring a comprehensive set of methods that include MFCCs, Chroma features, as well as Zero Crossing Rate, Root Mean Square Error, Spectral Centroid, Spectral Bandwidth, Spectral Contrast, and Spectral Rolloff. While MFCCs have been extensively applied in both speech and music processing, the incorporation of these additional features provides a more nuanced approach to music genre classification.

The importance of feature extraction in music can be traced back to some early attempts at understanding and classifying sound. Early research primarily focused on basic descriptors, such as pitch, tempo, and timbre. Pitch and Timbre: One of the earliest works by Minsky [11] shed light on the utilization of pitch and timbre for basic music classification tasks.

Zero Crossing Rate (ZCR) is one of the earliest and simplest features, indicating the rate at which a signal changes its sign. A study by Rossi et al. [12] expounded on its applicability in distinguishing percussive sounds from harmonic ones.

Energy as discussed by Vignoli and Pauws [13], energy, especially short-term energy, provides insights into the rhythmic characteristics of music.

Spectral Centroid is emphasizing the 'brightness' of a sound, Benetos et al. [14] illustrated the utility of the spectral centroid in distinguishing between musical instruments.

Spectral Rolloff is a crucial feature for discerning the harmonic content, studied by Peeters [15] showcased its effectiveness in segregating musical genres.

Zhang and Shao [16] discussed how spectral flux, capturing the rate of change in the power spectrum, can be pivotal in rhythm-based categorization.

Beat Histograms are Widely cited in the MIR community, Tzanetakis and Cook's [17] paper introduced beat histograms as a representation of rhythmic content, especially for genre differentiation.

MFCCs are Undoubtedly one of the most widely utilized features, MFCCs offer a compact representation of the power spectrum of sound. A comprehensive study by Logan [18] showcased the effectiveness of MFCCs in various MIR tasks, including genre classification.

Chroma Features: Burred and Lerch [19] introduced Chroma-based features, representing the energy distribution across pitch classes, enabling key and chord recognition tasks. Pioneered by research efforts from Müller and Ewert [20], time-series based features capture the temporal evolution of music, encompassing the intricacies of rhythm and melodic structures.

Initially, MFCCs were primarily used in speech processing, with Davis and Mermelstein being among the pioneers who demonstrated their effectiveness in speech recognition systems. As researchers recognized that the complexity of music exceeded that of speech, they began to investigate the potential utility of MFCCs for music-related tasks [21].

Logan was one of the early researchers who highlighted the importance of MFCCs in the field of music modeling, specifically for genre classification. His work shed light on how effectively MFCCs could be used to differentiate between various musical genres. [22].

Whitman and Ellis demonstrated the versatility of MFCCs by showing their applicability in identifying artists and songs. Their research confirmed that the subtle details captured by MFCCs make them well-suited for tasks like artist and song recognition. [23].

Benetos and Kotropoulos conducted research that illustrated the effectiveness of MFCCs in distinguishing musical instruments when used in combination with other features. Their work indicated that MFCCs could be employed for precise instrument recognition tasks. [24].

Chroma features have their roots in the twelve different pitch classes in Western music. They essentially provide a representation of the energy distribution across these pitch classes.

Fujishima was among the first to introduce into the area of Chroma features, introducing the concept of "Pitch Class Profile." This initial exploration served as a foundational stepping stone for future advancements in the field. [25].

Burred and Lerch contributed to the understanding of how Chroma features could be utilized for chord recognition tasks. Their research introduced a hierarchical methodology for music genre classification, in which Chroma-based features played a key role in improving classification accuracy [26].

Gómez highlighted the potency of Chroma features for key detection tasks, thereby influencing the way harmonic analysis is performed in music [27]. Similarly, Serra et al. used the stable nature of Chroma features to identify different renditions of the same song, marking a significant contribution to the area of music cover identification [28]. On the other hand, Pollard and Jansson critiqued the limitations of MFCCs, noting that these features may not always capture the intricate subtleties inherent in specific musical genres [29].

Müller and Ewert pointed out a limitation of Chroma features, emphasizing that these features may become complex and difficult to interpret in songs with rich harmonic layers, thereby posing a challenge for accurate classification [30].

The Zero Crossing Rate (ZCR) has been highlighted as a pivotal feature in understanding the rhythmic aspects of music. Tzanetakis and Cook explored its effectiveness in distinguishing genres like Rock from Classical, showing that ZCR could capture rhythmic nuances in different types of music 31].

Another important feature, Root Mean Square Error (RMSE), has been utilized to capture the power or amplitude of an audio signal. Eronen et al. demonstrated the significance of RMSE in discerning between genres based on their loudness levels, such as Ambient and Metal [32].

The Spectral Centroid and Spectral Bandwidth are often used to capture timbral characteristics of audio signals. In the work by Peeters, these features were shown to be crucial in differentiating between instrumental and vocal-centric genres [33]. Spectral Contrast and Spectral Rolloff are additional spectral features that have shown promise in distinguishing between genres with intricate instrumental arrangements. Though these features are less frequently discussed in the context of genre classification, they hold potential for capturing the complexities often found in genres like Jazz and Classical music.

## 2.3 Machine Learning in Music Genre Classification

## In the realm of Music Information Retrieval (MIR), the pursuit of accurately and efficiently classifying music genres has witnessed transformative progress. Machine learning has emerged as a game-changing approach, evolving the process from heuristic methods to sophisticated algorithms. This literature review navigates the journey of machine learning's applications in music genre classification, explaining key milestones, techniques, and challenges.

Navigating vast music libraries and databases necessitates effective music descriptors, and genre stands at the forefront of these descriptors. McKinney and Breebaart clarified the importance of genre, This feature plays a crucial role in both discovering new music and gaining a deeper understanding of its contextual elements [34].

Before the massive influx of sophisticated algorithms, rudimentary machine learning models held the helm. Pioneers like Tzanetakis and Cook combined insights from text retrieval and audio signal processing to set the stage for future research in genre classification using machine learning techniques. Decision Trees is an easily interpretable model, decision trees drew attention due to their hierarchical decision-making based on feature thresholds [35]. Mierswa and Morik The study offered a detailed evaluation of the algorithms' performance and flexibility in classifying different music genres [36].

Support Vector Machines (SVMs) are Esteemed for its capabilities in high-dimensional feature spaces, SVMs have been indispensable in genre classification. In a thorough experimental study, Li, Ogihara, and Li [37] demonstrated that Support Vector Machines (SVMs) achieved high accuracy in music genre classification tasks, especially when tailored with specific kernel functions. K-Nearest Neighbors (KNN): A non-parametric method, KNN was explored for its ability to classify based on similarities in feature spaces. Bergstra et al. [38] showcased its efficacy in music genre classification tasks, given appropriately defined distance metrics.

Random forests introduced ensemble learning to the arena. Focusing on robustness and reducing overfitting, these models, as examined by Geurts et al. [39], offered a significant increase in accuracy over standalone tree models. At the core of any machine learning model lies the features it processes. For music genre classification, feature selection and extraction determined the model's efficacy. Several studies, such as those by Logan [40], underscored the importance of MFCCs, chroma features, spectral contrast, and other auditory features in capturing the essence of genres.

The evolution and effectiveness of machine learning models are inextricably linked to the datasets on which they train. MIREX [41], GTZAN [42], and ISMIR [43] are few of the renowned datasets that provided a diversified array of tracks across genres, enabling comprehensive model training and validation. Despite significant advancements, challenges persist. Issues like the inherent subjectivity of genres, dataset biases, and the risk of overfitting remain pertinent. Researchers, as highlighted by Sturm [44], have persistently grappled with these challenges, The goal is to optimize the models to achieve both high predictive accuracy and ease of interpretation.

## 2.4 Deep Learning in Music Genre Classification

The incorporation of deep learning methods, into the classification of music genres is considered an advancement in the field of MIR. The versatility, ability to learn features and adaptability of neural networks have made them a preferred technique for researchers seeking to explore new possibilities, in genre classification. This review examines the applications, methodologies and progress achieved through the use of learning in music genre classification.

Deep learning, characterized by its neural architectures that mimic human brain functions, marked a significant departure from traditional machine learning models. The potential of these algorithms in MIR, especially music genre classification, has been profound, leading to transformative genre classification [45].

Dieleman and Schrauwen [46] were among the pioneers who explored the potential of deep learning for music genre classification. Their work, emphasizing the Convolutional Neural Network's (CNN) capabilities in automatically extracting hierarchical features from the use of spectrograms established a baseline for future research in this area.

Convolutional Neural Networks excel in capturing spatial hierarchies in data. Recognizing this, Choi et al. [47] built upon earlier works and tailored CNNs to efficiently process spectrograms and mel-spectrograms, obtaining commendable classification accuracies.

Recurrent Neural Networks Understanding the temporal dynamics of music is crucial. RNNs, and their variants like LSTM and GRU, have demonstrated their ability to model time-dependent sequences. Huang and Wu [48] elucidated their efficiency in capturing musical temporal structures, proving pivotal for genre classification tasks.

Autoencoders are unsupervised neural networks, adept at dimensionality reduction, have shown potential in capturing the inherent characteristics of music. Oord et al. [49] highlighted their capability in extracting condensed feature representations, thereby aiding the genre classification process.

Attention Mechanisms: Given the varying significance of different segments in a music track, attention mechanisms have garnered interest. Wang et al. [50] showcased how attention models can be embedded within neural architectures to weigh different segments differently, enhancing classification outcome.

Unlike traditional methods that rely heavily on manual feature extraction, deep learning has revolutionized the approach. Spectrograms, mel-spectrograms, and raw audio waveforms have become primary inputs. Dieleman et al. [51] detailed how CNNs, when fed with these raw representations, can automatically learn patterns crucial for genre discrimination.

The reliability and effectiveness of deep learning models hinge upon robust datasets. The GTZAN dataset [52], FMA dataset [53], and Deezer dataset [54] have become staples for deep learning-based genre classification endeavors. These datasets, characterized by their diversity and granularity, have facilitated comprehensive model evaluations.

While deep learning has significantly enhanced genre classification capabilities, it is not devoid of challenges. Issues like model interpretability, overfitting due to extensive model complexities, and the need for vast computational resources are pertinent. Lee et al. [55] extensively discussed these challenges, emphasizing the need for balanced model architectures and efficient training strategies.

**2.5 The Free Music Archive (FMA) Dataset**

The FMA dataset, introduced by Defferrard et al., is a beacon for researchers in music information retrieval [56]. Unlike other datasets, the FMA provides a comprehensive and eclectic mix of over 100,000 tracks spanning 161 genres. The granularity and richness of this dataset make it uniquely positioned for in-depth genre classification research. Defferrard and colleagues present a breakdown of the dataset, underscoring its structure, content, and inherent challenges [57].

The hierarchical taxonomy presented in FMA is a significant talking point. The dataset's structure, which places genres in a layered system, ranging from broader categories to intricate sub-genres, adds a layer of complexity to any analytical task [58]. Such a granular classification system demands models of exceptional sophistication that can maintain accuracy across different hierarchical levels. Bertin-Mahieux et al [59], in their exploration of music datasets, also highlight the necessity of intricate models when dealing with taxonomically dense datasets.

A standout feature of FMA, as pointed out by Defferrard et al., is its inclusivity. While many music datasets exhibit a noticeable skew towards Western music, the FMA breaks this mold. Its representation of a diverse range of genres from across the globe is commendable [60]. However, such diversity can be a double-edged sword. As Tzanetakis and Cook pointed out in their seminal work on musical genre classification, The dataset includes a wide range of 161 distinct musical genres, spanning from Rock and Jazz to lesser-known styles like Zydeco and Shoegaze of genres often present unique challenges, requiring models to distinguish between very subtle musical differences [61].

Furthermore, the sheer volume of tracks in the FMA dataset raises computational and methodological concerns. According to Humphrey et al., handling such vast amounts of data demands optimized data processing pipelines, especially when extracting intricate features [62]. This becomes particularly important given that genre classifications in FMA are not just about identifying broad genres but also about discerning the nuanced differences between sub-genres.

This taxonomical depth, while being a rich source of information, introduces challenges for model development. Agreeing with previous research, Bertin-Mahieux and his team stressed the importance of using advanced, specialized algorithms to accurately sort through the FMA's complex genre classification system [63].

The FMA dataset's extensive collection of over 100,000 tracks poses significant challenges, requiring substantial computational resources for data processing and raising methodological questions about how to effectively analyze such a large dataset. Humphrey et al.'s work provides valuable insights into managing such extensive datasets, highlighting the importance of optimized data processing pipelines, particularly when extracting detailed features [64]. Sturm, in his comprehensive study on genre classification, emphasized that sheer data volume doesn't always translate to easier classification. The expansive track range in FMA means models must not only recognize overarching genre themes but also the intricate differences between closely related sub-genres [65].

Recent advances, like Choi et al.'s study on using deep learning for music analysis, suggests that neural networks might hold the key to unlocking FMA’s full potential [66]. With FMA's vastness and complexity, it is no surprise that researchers are looking towards advanced methodologies to harness its wealth of information.

However, this diversity is not without its challenges. The foundational work of Tzanetakis and Cook underscores the inherent complexities in classifying datasets with extensive genre variations [67]. The nuanced distinctions between certain genres, especially those from non-Western traditions, require models that can account for intricate musical variations, demanding a balance between precision and adaptability.

With FMA's expansive track repository comes a set of computational challenges. Humphrey et al.'s insightful analysis draws attention to the implications of processing massive datasets, emphasizing the need for refined data processing pipelines [68]. The challenges are not just restricted to computation; methodological concerns, especially in feature extraction, also come to the fore. Recent developments in deep learning for music analysis, especially as advocated by Choi et al., suggest that neural networks might be pivotal in harnessing FMA's potential [69]. Their work exemplifies how advanced methodologies could unlock deeper insights from vast datasets like FMA.

# 3. Methodology

# The methodology of this project serves as a rigorous roadmap aimed at achieving the core objective of classifying music genres using machine learning and deep learning techniques on the Free Music Archive (FMA) dataset. With the dataset encompassing a broad spectrum of over 100,000 tracks across various genres, the methodology is strategically broken down into key stages, each essential for successful project completion. These stages include Data Pre-processing, where the FMA dataset will be cleaned and Exploratory Data Analysis (EDA) to statistically examine the data, Feature Extraction, focusing on the identification and extraction of significant audio attributes like MFCCs and Chroma features; Model Development to build and compare machine learning and deep learning models such as Support Vector Machines, Random Forests, Model Evaluation, where accuracy will be used to assess model performance; and finally, Model Fine-tuning, where iterative optimization of hyperparameters will be performed based on the accuracy. Each of these stages will be elaborated in the subsequent sections, articulating the specific techniques, tools, and rationales that guide the project toward its objectives.

# 3.1 Data Collection

The main dataset used for this project was obtained from the Free Music Archive (FMA) a platform dedicated to distributing downloadable music. this dataset has played a role in facilitating music information retrieval tasks and research studies.

The FMA dataset has been cited in research publications, including a paper titled "FMA” A Dataset for Music Analysis" presented at the 18th International Society, for Music Information Retrieval Conference (ISMIR 2017). This paper not only introduces the dataset. But also provides a comprehensive analysis of its structure, content and potential applications.

The Free Music Archive Dataset (FMA) is comprised of both Metadata and Audio data. The metadata is organized into separate datasets, including 'raw\_tracks.csv,' 'raw\_albums.csv,' 'raw\_artists.csv,' and 'raw\_genres.csv,' which provide details about tracks, albums, artists, and genres, respectively. The audio data contains 106,574 tracks, each 30 seconds long, spanning 161 different genres.

**3.1.1 Tracks Data**

The FMA track dataset provides a view of individual tracks, encapsulating data about its album, artist, licensing, and other vital metrics. Such a dataset is pivotal not only for music enthusiasts and researchers but also for building recommendation systems, analytics platforms, and other music-based digital solutions.

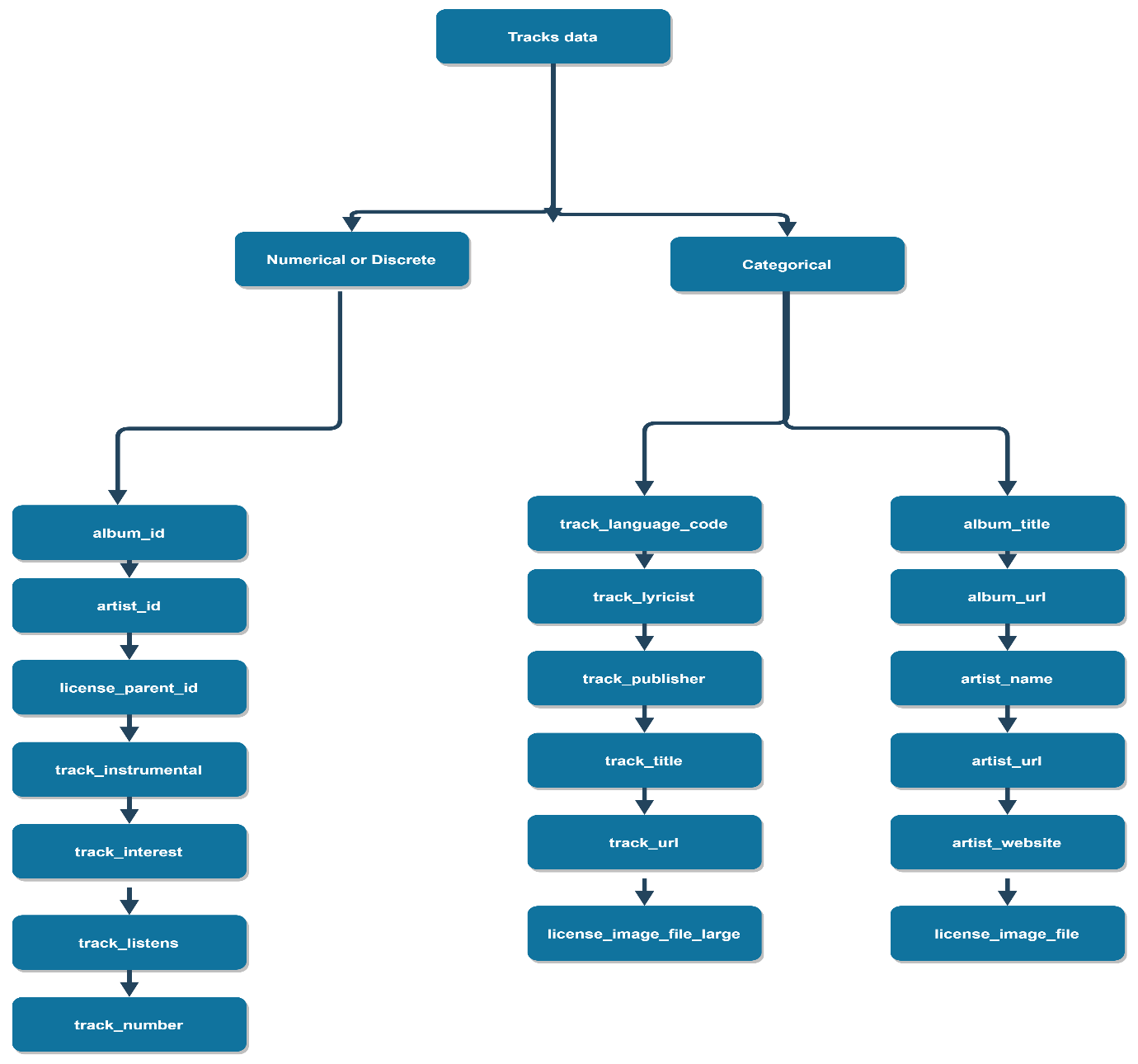


Fig 3.1.1: Comprehensive Overview of Track Attributes in the Free Music Archive Dataset

Figure 3.1 shows the structure of the "Tracks Data" within the Free Music Archive Dataset (FMA). This visual representation gives an insight into the multiple attributes that each track possesses, from basic information like the track ID, title, and duration attributes such as genre classification and related metadata.

**album\_id:** This is a numerical identifier unique to each album. It facilitates linking tracks to their respective albums.

**album\_title:** A categorical attribute representing the name of the album to which the track belongs.

**album\_url:** A categorical attribute that provides a direct URL to the album's page on the FMA website.

**artist\_id:** A numerical identifier that is unique for every artist.

**artist\_name:** This categorical attribute reveals the name of the artist or band that produced the track.

**artist\_url:** A categorical URL attribute that leads users to the artist's profile page on the FMA platform.

**artist\_website:** This categorical attribute provides an external URL leading to the artist's personal or official website.

**license\_image\_file:** A categorical URL attribute pointing to the small image representing the license under which the track has been released.

**license\_image\_file\_large:** This categorical URL showcases a larger version of the license image.

**license\_parent\_id:** A numerical identifier connecting the track to its overarching license group.

**track\_information:** This categorical column would normally contain additional information or description about the track.

**track\_instrumental:** A binary numerical column. A value of '0' indicates the track has vocals, while '1' indicates it's instrumental.

**track\_interest:** A numerical attribute that might represent the number of users who have shown interest in the track, perhaps through likes, shares, or comments.

**track\_language\_code:** A categorical attribute denoting the language of the lyrics in the track (e.g., 'en' for English).

**track\_listens:** A numerical column reflecting the number of times the track has been listened to.

**track\_lyricist:** Categorical column which would name the lyricist of the track.

**track\_number:** A numerical column indicating the position of the track in its album. For example, '1' would imply it is the first track.

**track\_publisher:** Categorical column for the entity or individual responsible for publishing the track.

**track\_title:** The categorical name or title of the track.

**track\_url:** A categorical column providing a direct URL to the track's page on the FMA website.

**3.1.2 Album Data**

This dataset can be invaluable for understanding user preferences, album popularity, artist influence, and the overall landscape of the music available on FMA.

In Figure 3.2, we present a thorough examination of the album metadata available in the Free Music Archive (FMA) dataset. This figure aims to offer insights into the various attributes and characteristics that are associated with each album, such as release date, artist information, and genre tags, among others.

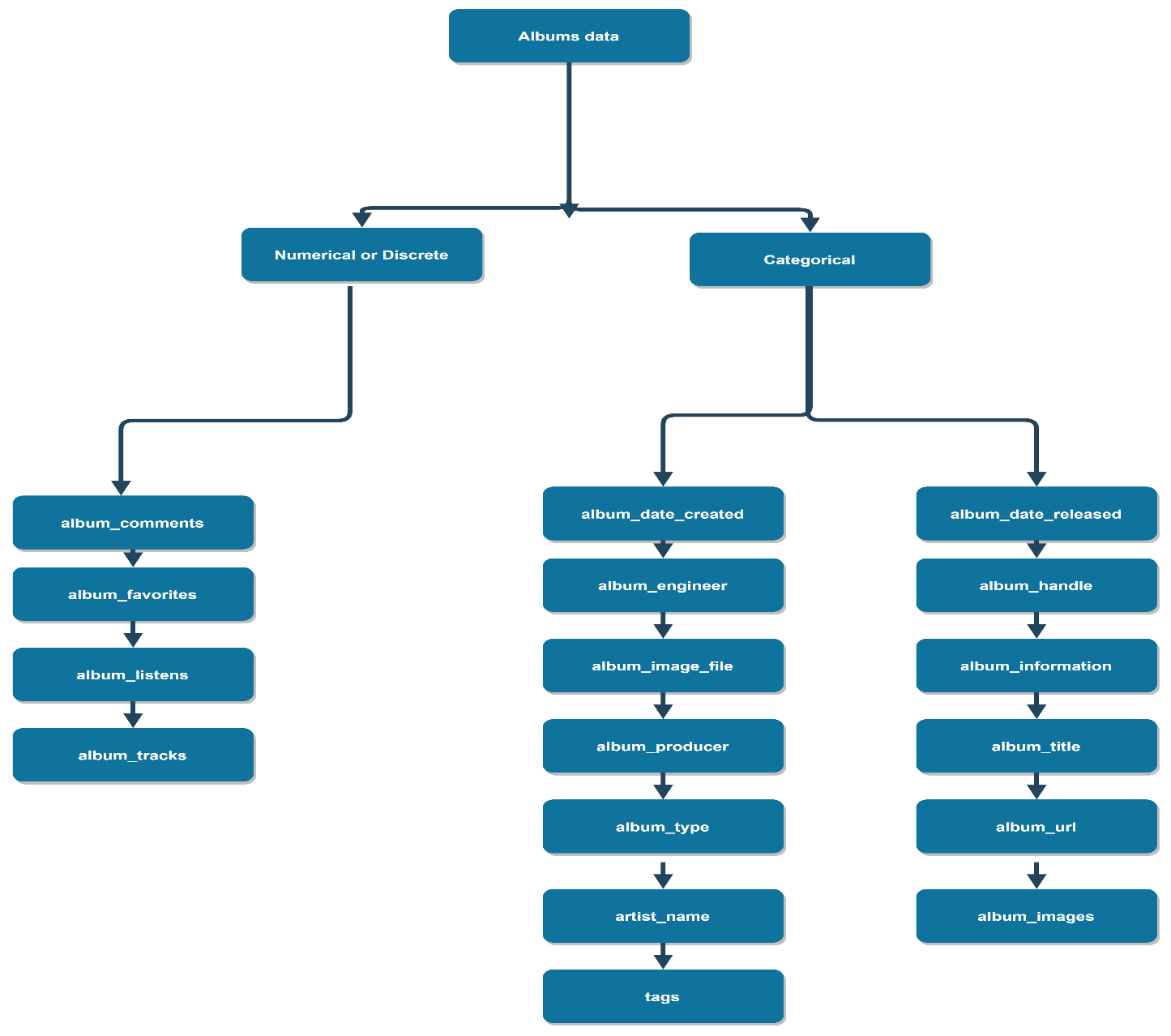


Fig 3.1.2: Album Metadata

**album\_comments:** This is a numerical column reflecting the total number of comments an album has received. A higher number might suggest increased user engagement or controversy.

**album\_date\_created:** A categorical attribute that represents the date and time the album was added to the FMA platform.

**album\_date\_released:** This categorical column showcases the official release date of the album.

**album\_engineer:** A categorical column that names the sound engineer(s) who worked on the album. In the given data.

**album\_favorites:** A numerical column that shows how many times the album has been marked as a favorite by users.

**album\_handle:** This categorical attribute provides a unique identifier or name for the album, typically a combination of the artist and album name.

**album\_image\_file:** A categorical URL column pointing to the cover image of the album on the FMA platform.

**album\_images:** This categorical attribute offers a list of image metadata associated with the album, including image IDs and their respective URLs.

**album\_information:** A categorical column that might contain additional descriptive information about the album. It can include anecdotes, historical context, or any other relevant details.

**album\_listens:** A numerical column that tells us the total number of times tracks from the album have been played.

**album\_producer:** A categorical column intended to name the producer(s) of the album. Appears to have missing data in the provided dataset.

**album\_title:** This categorical attribute represents the official title or name of the album.

**album\_tracks:** A numerical column denoting the total number of tracks in the album.

**album\_type:** A categorical attribute indicating the type of the album, e.g., "Album", "Live Performance", or "Radio Program".

**album\_url:** A categorical URL column that provides a direct link to the album's page on the FMA website.

**artist\_name:** A categorical column revealing the name of the artist or band that released the album.

**artist\_url:** A categorical column offering a direct URL to the artist's profile on the FMA platform.

**tags:** This categorical column contains a list of tags or keywords associated with the album. These tags can help in understanding the genre, mood, or themes associated with the album.

**3.1.3 Artist Data**

The dataset at hand encapsulates the rich information related to artists in the Freemusic Archive (FMA). Each entry portrays a snapshot of the artist's presence on the platform, encompassing their biography, active years, related projects, and more. It amalgamates both numerical and categorical data to depict a multi-dimensional portrait of each artist.

**artist\_active\_year\_begin:** This numerical column marks the starting year when the artist began their active music career.

**artist\_active\_year\_end:** This numerical column indicates the year when the artist concluded their active music career.

**artist\_associated\_labels:** Categorical data that lists down the record labels associated with the artist.

**artist\_bio:** A categorical column that gives a brief biography or description of the artist, often written in HTML format.

**artist\_comments:** Numerically quantifies the total comments an artist has garnered on the FMA platform.

**artist\_contact:** Categorical data that might provide contact information or a representative's name for the artist.

**artist\_date\_created:** A categorical timestamp noting when the artist's profile was created on the FMA.

**artist\_donation\_url:** Categorical URL data pointing to a site where fans can donate to support the artist.

**artist\_favorites:** A numerical column that displays how many times the artist has been marked as a favorite.

**artist\_flattr\_name:** A categorical variable, associated with the artist's account on the Flattr micro-donation platform.

**artist\_location:** Categorical data pinpointing the artist's location or hometown.

**artist\_longitude:** Numerical data providing a geospatial coordinate (longitude) of the artist's location.

**artist\_members:** Categorical list of members if the artist is a band or a collective.

**artist\_name:** The categorical name or title by which the artist is recognized.

**artist\_paypal\_name:** A categorical attribute, presumably the artist's PayPal ID for transactions or donations.

**artist\_related\_projects:** Categorical column enlisting other music projects or bands associated with the artist.

**artist\_url:** Categorical URL data linking to the artist's profile on FMA.

**artist\_website:** Categorical column with a URL leading to the artist's personal or official website.

**artist\_wikipedia\_page:** Categorical URL data that redirects to the artist's Wikipedia page, if available.

**tags:** Categorical column containing a list of keywords or tags associated with the artist, elucidating their style, genre, or other pertinent attributes.

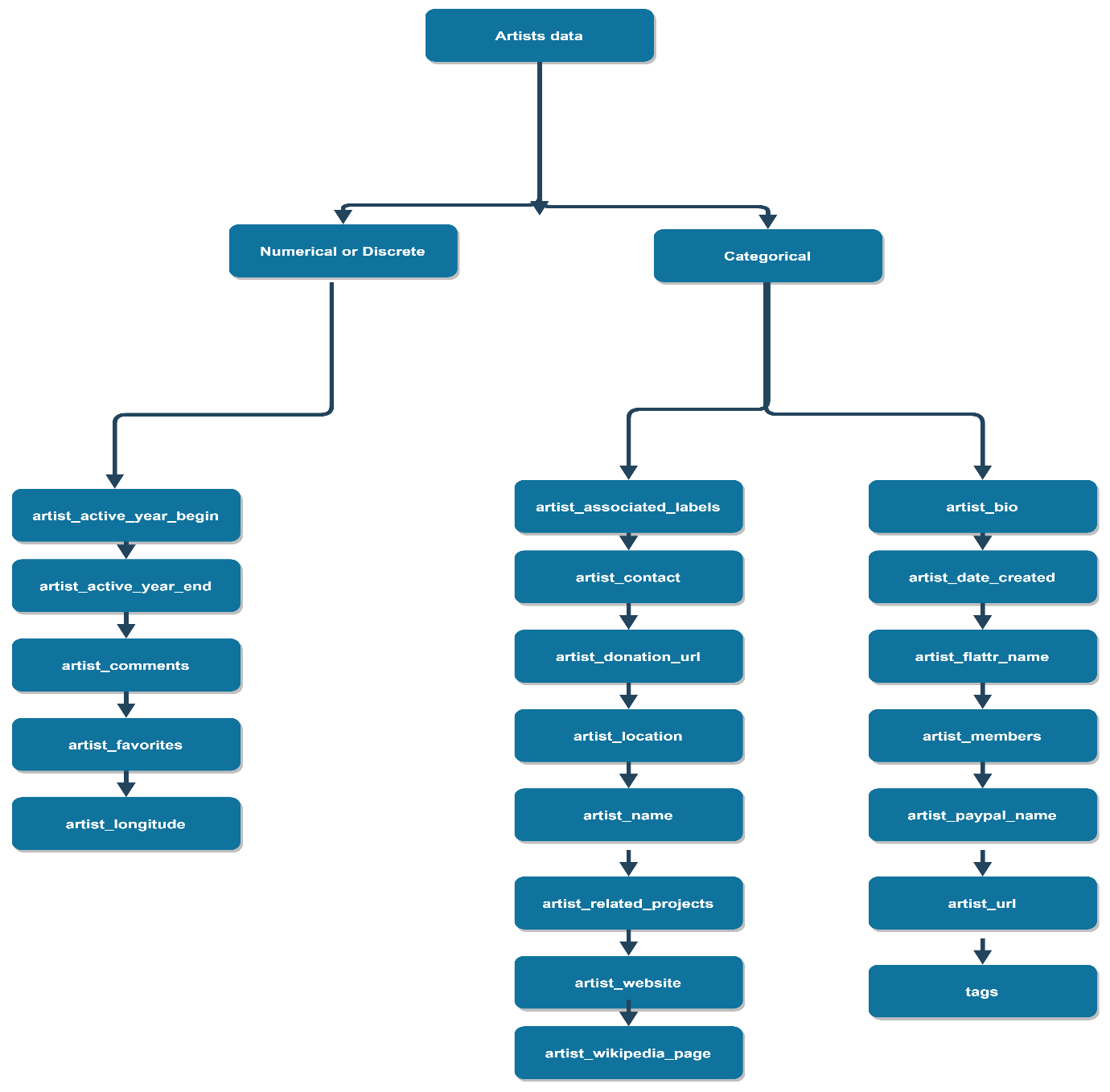
****

Fig 3.1.3: Comprehensive Overview of Artist Metadata

Figure 3.3 provides a comprehensive overview of the metadata associated with artists in the dataset. This encapsulates a range of attributes that offer valuable context and information about each artist.

**3.1.4 Genre Data**

The dataset offers a detailed breakdown of musical genres as categorized in the Free Music Archive (FMA). Genres serve as the organizational foundation for any music platform, helping users easily sort through a diverse array of musical compositions. Each row pertains to a unique genre, offering insights into its hierarchical relationship with other genres and its distinctive visual representation on the platform.

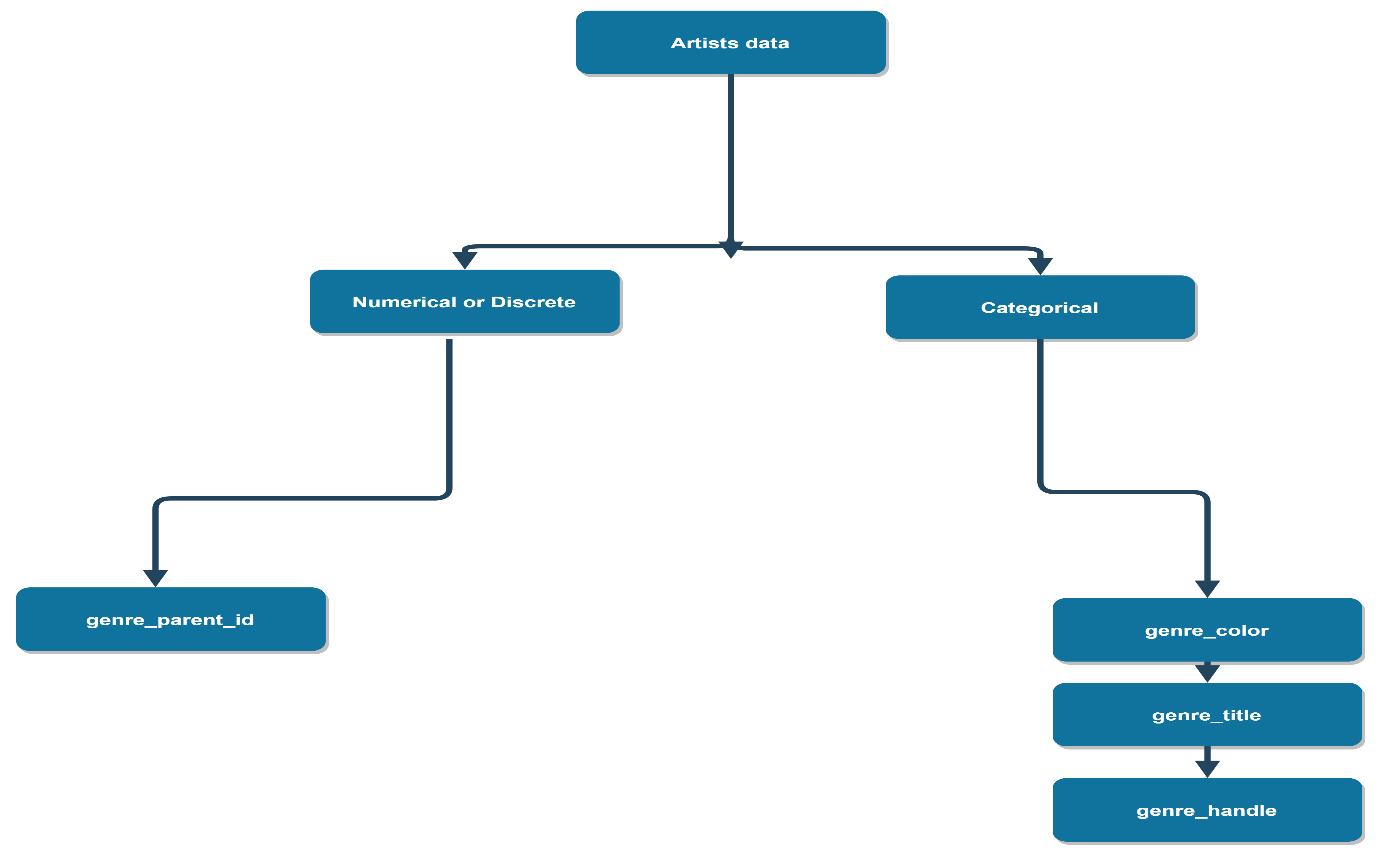


Fig 3.1.4: Comprehensive Overview of Genre Metadata

Figure 3.4 offers an in-depth look at the metadata related to musical genres within the dataset. This exhaustive overview includes key attributes that define and categorize each genre.

**genre\_color:** This categorical column denotes a unique color code (in HEX format) associated with each genre. This could be used on the platform for visual representation, enabling users to identify genres through specific color cues.

**genre\_handle:** A categorical attribute, this column provides a concise identifier or slug for the genre. It is typically used in URLs or for internal platform reference.

**genre\_parent\_id:** This numerical column indicates a hierarchical relationship among genres. The value represents the genre ID of the 'parent' genre. For instance, 'Avant-Garde' is a child of the genre with ID 38. A NaN (Not a Number) suggests that the genre is a primary genre without a parent.

**genre\_title:** The categorical title of the genre, offering a clear and recognizable name for users. This is the primary descriptor of the musical style or theme.

## 3.2 Data Cleaning

Data cleaning is vital for ensuring the accuracy and reliability of datasets. Clean data leads to better decision-making, improved machine learning model performance, and avoids misleading results. Without this step, insights could be based on erroneous information, leading to suboptimal conclusions and strategies.

Upon obtaining the Free Music Archive (FMA) dataset, the initial step was to examine the completeness of the dataset across various dimensions, namely: tracks, albums, artists, audio clips. To identify missing data, we evaluated the number of instances not found for each of the forementioned categories.

**Tracks:** Out of a potential 155,320, we successfully collected data for 109,727 tracks. This leaves us with 45,594 tracks not accounted for.

**Albums:** We accumulated data for 15,234 albums. Among the tracks we have, they span 15,714 unique albums, meaning we have 480 albums not accounted for in our collected dataset.

**Artists:** Data for 16,916 artists was successfully collected, and our tracks data included songs from 17,166 unique artists. Thus, 250 artists were not represented in our dataset.

**Genres:** The dataset covered 164 unique genres.

**Audio:** While we aimed for complete audio data corresponding to our track listings, we found audio data for 106,574 tracks. There was no discrepancy with the track listings, but 180 audio data were missing.

**Tracks data:** Columns such as license details, track file and image specifics, and URLs, which are only crucial for the original web interface, are removed as they do not add analytical value to the project. Similarly, fields like copyright details, track disc number, explicit notes, and instrumental details are discarded because they are either sparsely populated or their essence can be captured through other existing columns.

The column names are also modified to enhance clarity and descriptiveness. For instance, 'license\_title' is renamed to 'track\_license,' and the vague term 'tags' becomes 'track\_tags.' This naming convention aims to make the dataset more intuitive for subsequent stages of analysis. Further, we transform some of the existing columns to make the data more amenable for analysis. The 'track\_duration' field is processed using a custom function called 'convert\_duration,' which reformats its content into a more usable form. The 'track\_date\_created' and 'track\_date\_recorded' fields are converted into pandas datetime objects, facilitating easier time-series analysis and date-specific computations down the line.

Handling missing data is another crucial step in data preprocessing. For columns like 'album\_id' and 'track\_bit\_rate,' we fill the missing values with -1 and later type-cast these columns to integers. This serves as a placeholder that indicates the absence of data. Lastly, the 'track\_genres' column, initially a string representation of lists containing dictionaries, is transformed into a more manageable form. Specifically, it is converted into lists of integers, representing genre IDs. This conversion is carried out by evaluating the original string to retrieve the list and then extracting the

**Albums data:** The goal here, just like with the tracks data, is to prepare the dataset for analysis by removing unnecessary details and removing. Our dataset for albums initially contained several columns, some of which, upon closer examination, were deemed unnecessary for our specific analytical goals. The cleaning process was undertaken to trim down the dataset and enhance its clarity.Certain columns in the album dataset such as the artist's name and related URLs were discarded. These pieces of information were already captured in the track’s dataset, and maintaining them here could introduce potential discrepancies. Furthermore, columns such as album\_handle and certain image-related columns were also removed. While the image details might be relevant for other tasks like content retrieval, they were not pertinent to our immediate analytical objectives.

To ensure that the dataset is self-explanatory and reduces potential confusion, we decided to make column names more descriptive. For instance, the generic tags column was renamed to album\_tags. This renaming provides clear context, emphasizing that these tags pertain solely to the albums. Date data often comes in varied formats, which can lead to challenges during analysis and standardized the date-related columns, converting them into a uniform format. This effort targeted columns such as album\_date\_created and album\_date\_released, ensuring they are in a consistent format for any time-series or date-driven analysis.

**Artist’s data:** Removal of Extraneous Information: Several columns such as artist's websites, URLs, and handle details were eliminated. The primary reason for this was that certain pieces of this information were already captured in the track’s dataset, risking redundancy. Additionally, certain columns with information about the artist's donation methods or contact data were removed. These, while informative for potential outreach or fundraising projects, did not fit our immediate analytical goals and had limited entries.

The dataset originally contained columns referencing artist images. Although these images can serve a valuable purpose in specific use cases (like building a content-based recommendation system using image processing), they were not relevant to our current objectives. As such, these columns were removed, simplifying the dataset. Ensuring that our dataset is easy to understand was one of our prime objectives. To this end, generic columns like tags were renamed to a more descriptive artist\_tags. This renaming step reduces ambiguity, making it clear that these tags are associated exclusively with artists.

We standardized columns that contained date data, especially ones relating to the artist's active years. This standardization is pivotal for maintaining consistency across the dataset and simplifying any subsequent time-related analysis. The rare instances of aberrant year entries, represented as 0.0, were substituted with NaN values to denote their missing nature.

**Genre data:** The Genre dataset is pivotal for categorizing tracks and understanding music's diverse tapestry. To maximize analytical efficiency, we embarked on a meticulous refinement of this dataset. Attributes like genre\_handle and genre\_color didn't serve our core objectives, leading to their removal. This step ensured that our dataset remains concise and relevant to the analysis at hand. A vital aspect of the genre dataset was the relationship between genres – more specifically, how certain genres are children of broader parent genres. We ensured that these relationships were intact and consistent.

Some genres had incorrect or missing parent data. In such instances, we referenced the genre hierarchy present on a related website. This step brought coherency between our dataset and the recognized genre classification on the website.

We identified that a handful of tracks were misclassified under the non-existent genre "806". Instead, these were meant to fall under the "Hip-Hop" genre. We rectified this error by moving the tracks to their appropriate category.

With a hierarchy of genres from leaf to root established, we created an exhaustive list for each track. This comprehensive genre list offers a richer context, allowing tracks to be analyzed across multiple genre categories.

We then calculated the total number of tracks per genre, identifying some genres with zero tracks associated. This data can guide potential data augmentation or be used to assess the dataset's comprehensiveness.

Each track was attributed to a top-level genre, providing an overarching classification for analysis. If a track spanned multiple top-level genres, it was marked as 'not defined' to ensure accuracy.

**Audio Data:**

The audio in the FMA dataset is often stored in formats such as MP3, ensuring a balance between file size and quality. This format is popularly used for music storage and streaming.

Each audio track in the FMA dataset varies in length, but for genre classification tasks, often fixed-length clips (e.g., 30 seconds) are extracted from each track to maintain uniformity in the dataset. This ensures consistent input sizes for machine learning models.

Typically, the audio tracks maintain a standard bitrate of 128 kbps or 256 kbps. This represents a compromise between file size and audio clarity, suitable for most classification tasks.

A common sample rate in FMA is 44.1kHz, aligning with CD-quality audio. It provides sufficient data to capture most musical nuances.

****

Figure 3.2.1: Summary of the data cleaning

**Merging the DataFrames**

Any absent entries in the 'albums' subset within the 'not\_found' dataframe were identified and subsequently replaced with a placeholder value of -1, ensuring uniformity in data type by converting all album and artist identifiers to integers.

The main 'tracks' dataframe was expanded through a left merge with the 'albums' dataframe using the album identifiers. This process introduced additional columns to the 'tracks' dataframe, revealing that a certain number of tracks lacked extended album information. It was further ascertained that these missing entries corresponded with the previously identified 'not\_found' album entries. Any discrepancies between original and duplicated album titles post-merge were reconciled, and redundant columns were eliminated.

Similarly, the 'tracks' dataframe was enriched with information from the 'artists' dataframe via another left merge, this time using artist identifiers. It was observed that a segment of tracks lacked extended artist information. These absent entries corresponded with the 'not\_found' artist identifiers. Disparities between the original and duplicated artist names were addressed, and surplus columns were discarded.

The structure of the columns in the 'tracks' dataframe was then adjusted, segmenting them into three main categories: 'track', 'album', and 'artist'. Missing data points within the 'tags' columns of both 'album' and 'artist' were filled with empty list placeholders, ensuring data consistency.

Finally, specific columns, namely those associated with favorites, comments, listens, and tracks, underwent further data cleaning. Missing values were replaced with a placeholder value of -1, and the data types of these columns were standardized to integers.

We combined multiple datasets—'tracks,' 'albums,' and 'artists'—into a single unified dataset to facilitate a comprehensive analysis. Key identifiers like 'album\_id' and 'artist\_id' were used for merging.

**3.3 Exploratory Data Analysis (EDA)**

The following figure 3.3.1 provides a multi-faceted view of track management within our dataset. It plots the largest track ID, the number of tracks currently present, and the rate at which new tracks are added every two months. By studying this graph, we can gain insights into the dataset's growth, stability, and data accumulation patterns over time.

Largest Track ID is the line blue in color which represents the highest track\_id value within the dataset at any given time. It provides an upper limit to understand the dataset's scale, with the current highest track\_id being 155,320.

Tracks Still Present is the line green in color tells us about the tracks that are still available in the dataset. It doesn't ascend as sharply as the 'largest track id' because it accounts for deleted tracks, making it a more reliable representation of the actual dataset size. As of now, there are 48746 deleted tracks.

Tracks Added per 2 Months is the dashed line represents the number of tracks added to the dataset every two months. This offers insights into the rate of data accumulation and could be crucial for understanding seasonal or temporal variations in data entry. The frequency varies between 0 and 4,000, indicating the dynamic nature of track addition.

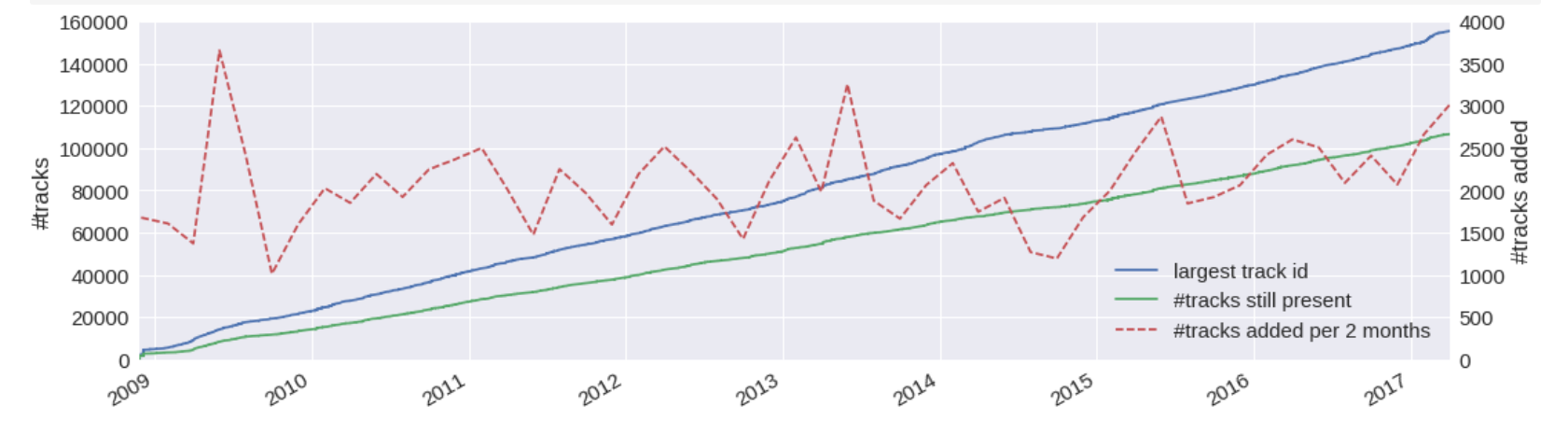


Fig 3.3.1: Growth and integrity of tracks in the FMA dataset over time

The Fig 3.3.1 shows the dataset has seen a consistent influx of new tracks over time. There are noticeable spikes in track additions, which may coincide with specific events or data collection drives. Despite deletions, the overall number of tracks in the dataset has been steadily increasing, reinforcing its growing volume and diversity.

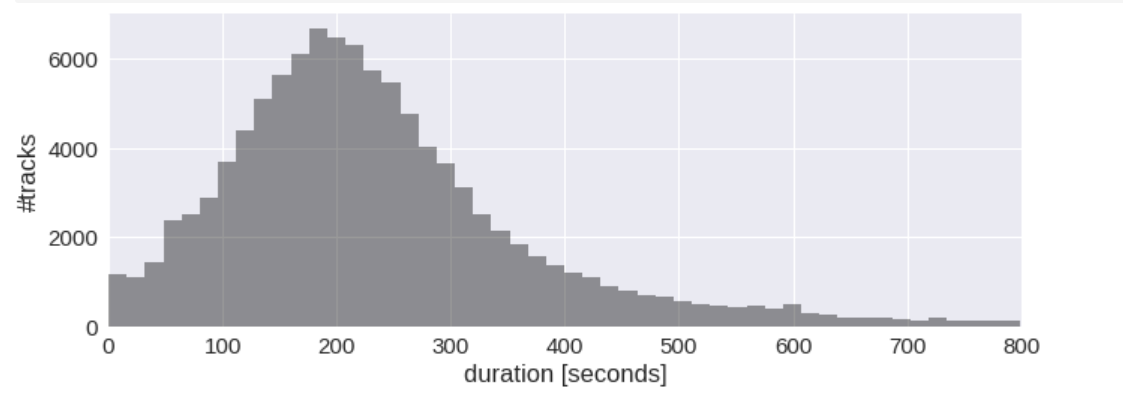


Fig 3.3.2: Distribution of track durations in the Free Music Archive (FMA) dataset

The majority of tracks in the dataset have durations less than 800 seconds (or 13.3 minutes). As shown in histogram in Fig 3.3.2, this provided a clear perspective on the concentration of track lengths, allowing us to observe how many tracks fall within specific duration ranges.

**3.3.2 Statistical Summary**

|  |  |
| --- | --- |
| **Total tracks** | **106574** |
| **Average duration** | **277.85 seconds** |
| **Standard Deviation** | **305.52 seconds** |
| **Shortest track** | **0 seconds** |
| **Longest tracks** | **18350** |

Table 3.3.2 statistical summary of the data

Total Tracks Analyzed: 106,574

Average Duration: Approximately 277.85 seconds (or about 4.63 minutes).

Standard Deviation: Approximately 305.52 seconds, suggesting a wide variation in track lengths.

Shortest Track: 0 seconds (which might suggest some tracks are mere placeholders or have missing data).

Longest Track: 18,350 seconds (or about 5 hours and 5 minutes), indicating the presence of extremely long tracks or compilations.

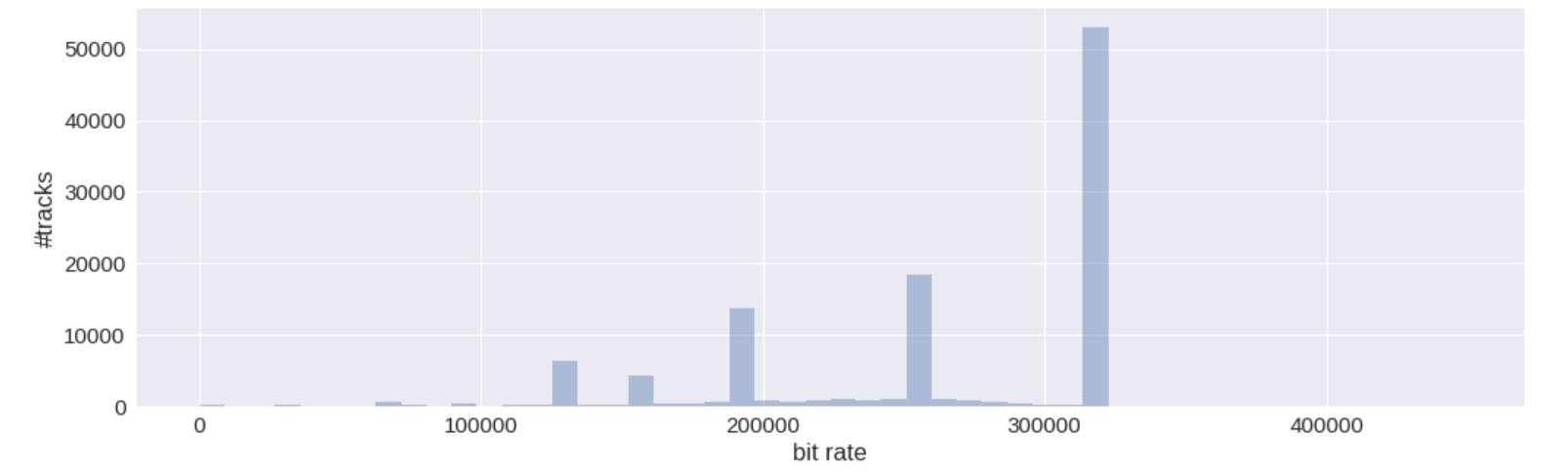


Fig 3.3.3: Distribution of bit rates in the Free Music Archive (FMA) dataset

The bit rate is a crucial indicator of audio quality in digital music tracks. Upon examining the tracks in the FMA dataset, several observations were made

Common Bit Rates are the most frequently encountered bit rates among the tracks are 320 kbps, 256 kbps, 192 kbps, 128 kbps, and 160 kbps. These figures are typical for MP3 files, representing a range from higher quality to more compressed audio. Average bit rate of around 263 kbps suggests that the majority of tracks in the dataset are of moderate audio quality, slightly exceeding the common standard bit rate for MP3 files.

A visual distribution Fig 3.3.3 showcases the frequency of tracks against their respective bit rates, further emphasizing the prevalence of the aforementioned common bit rates. The distribution also hints at the presence of variable bit rate (VBR) encodings, as certain less frequent bit rates do not neatly fit into standard encoding categories. VBR is a method of encoding audio that adjusts the bit rate according to the audio's complexity, potentially leading to variations outside of the standard presets, Now, please refer to Figure 3.3.4 for an in-depth visual representation of these points.

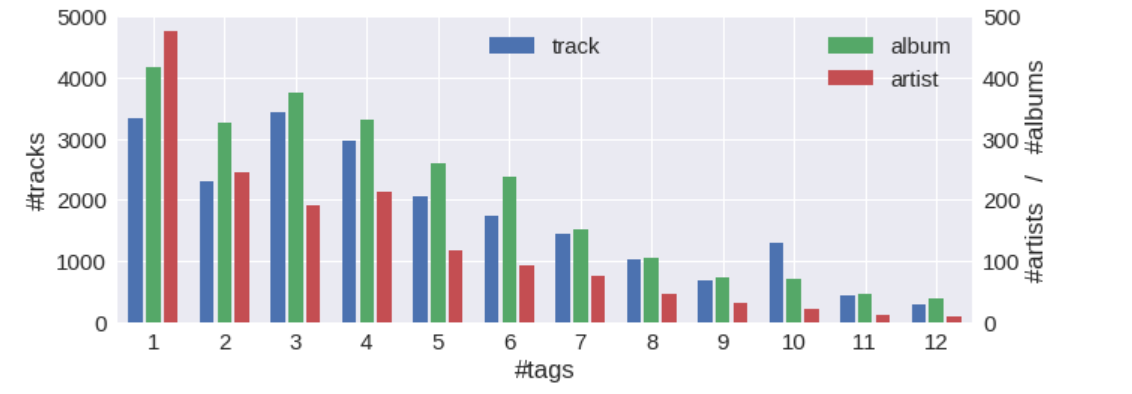


Fig 3.3.4: Distribution of tags across tracks, albums, and artists

The visualization Fig 3.3.4 provides a clear representation of these distributions. Specifically, the histogram illustrates the frequency of the number of tags for tracks, albums, and artists.

The data provides insights into the distribution of tags assigned to tracks, albums, and artists. These tags are essentially keywords or labels that describe some aspect of the content, be it the genre, mood, theme, or any other relevant descriptor.

For tracks: The range of tags per track varies widely from tracks that have no tags at all (0 tags) to tracks that have as many as 150 tags.

For albums: Similar to tracks, albums can also have a broad range of tags. The albums in this dataset range from having no tags to being labelled with up to 150 different tags.

For artists: Artists tend to have fewer tags compared to tracks and albums. The range for artists starts from none and goes up to 55 tags.

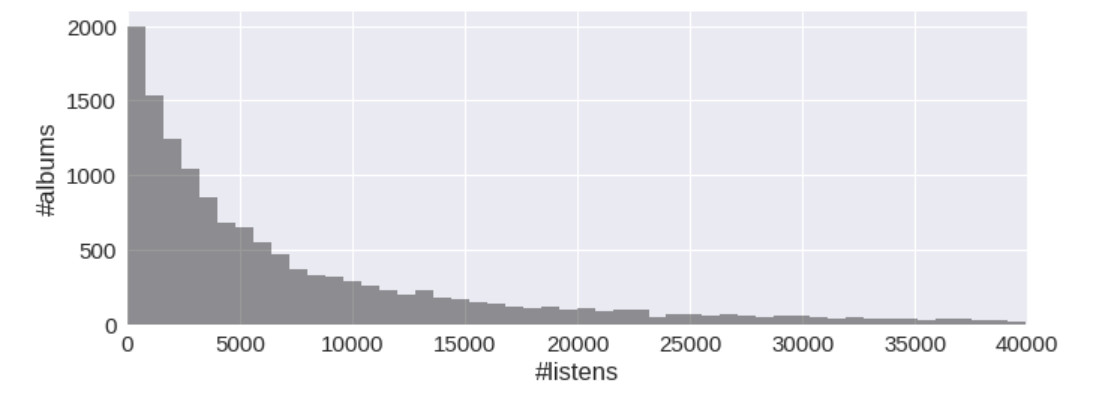


Fig 3.3.5: Distribution of tags across tracks, albums, and artists

From the observed data, most albums fall within a specific range of listens, with only a few exceptions reaching extremely high numbers of listens. Most notably, the album with the highest number of listens has been listened to an astonishing 3,564,243 times. This outlier suggests that while many albums may enjoy moderate success, a select few albums manage to captivate a significantly larger audience.

The fig 3.3.5 shows this disparity in album popularity, and it's an essential tool for stakeholders aiming to understand the listening habits of their user base or for artists trying to gauge the relative success of their albums.

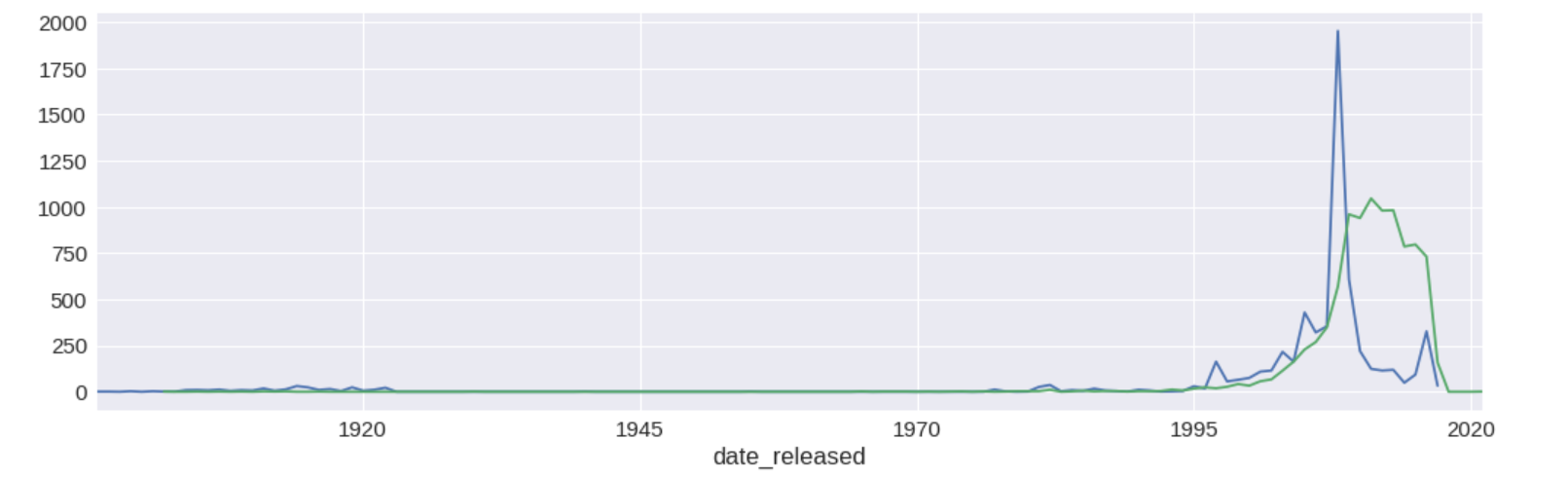


Fig 3.3.6: Annual trends: track recordings vs. album releases

The above graph Fig 3.3.6 represents the number of tracks recorded each year. This graph gives an overview of the activity in terms of how many tracks are being recorded annually. As we can see in the graph that between the period 1995 – 2020 has the highest number of recordings of tracks being recorded.

date\_recorded is the line which is blue in color plotting the frequency of tracks based on their recording dates on a yearly basis. It accumulates the number of tracks recorded each year and displays it in the plot.

date\_released is the line which is green in color focuses on the frequency of albums based on their release dates. Similar to the first line, it tallies the number of albums released each year and plots it.

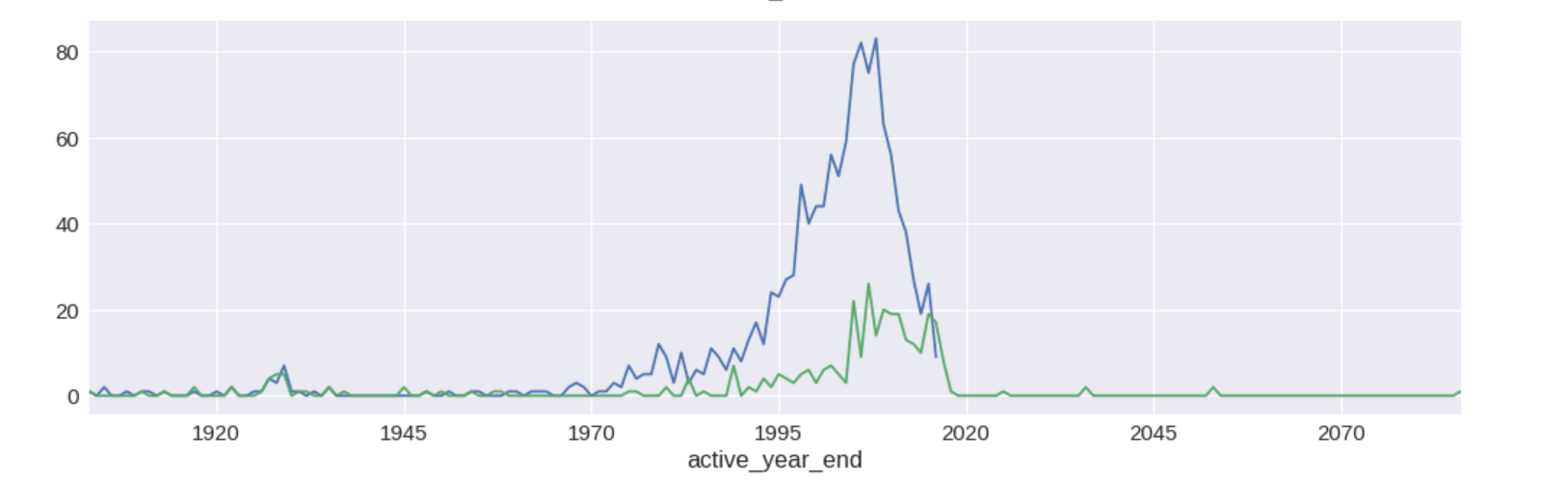


Fig 3.3.7: Artists' career timeline: start vs. end years

The above graph Fig 3.3.7 offers a perspective on the inflow of new artists and the outflow of existing ones over time. A rise in emerging artists peaks around mid-period 1970 - 2020, followed by a decline towards 2020.

active\_year\_begin is the line blue in color generates a plot showing the frequency of artists based on the years they became active. Essentially, it plots how many artists started their careers each year.

active\_year\_end contrary to the previous line, this line plots the number of artists based on the years they ended their careers or became inactive.

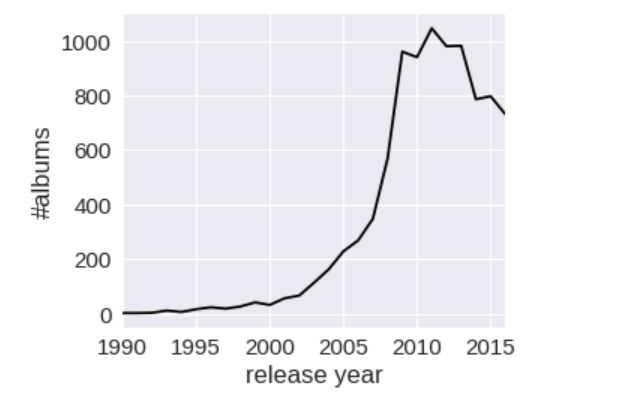


Fig 3.3.8: Annual album releases: 1990-2017

Fig 3.3.8 captures the distribution of album releases over a span of 27 years, from 1990 to 2017. Using the data from tracks, filtered out unique albums and charted their release dates on an annual basis. This visualization helps us understand the trend and frequency of album releases during this period. Fig 3.3.8 highlights each year with a count of albums released. Although the dataset includes album release data ranging from 1902 to 2021, for the sake of clarity and relevance, this particular visualization is limited to the time frame between 1990 and 2017.

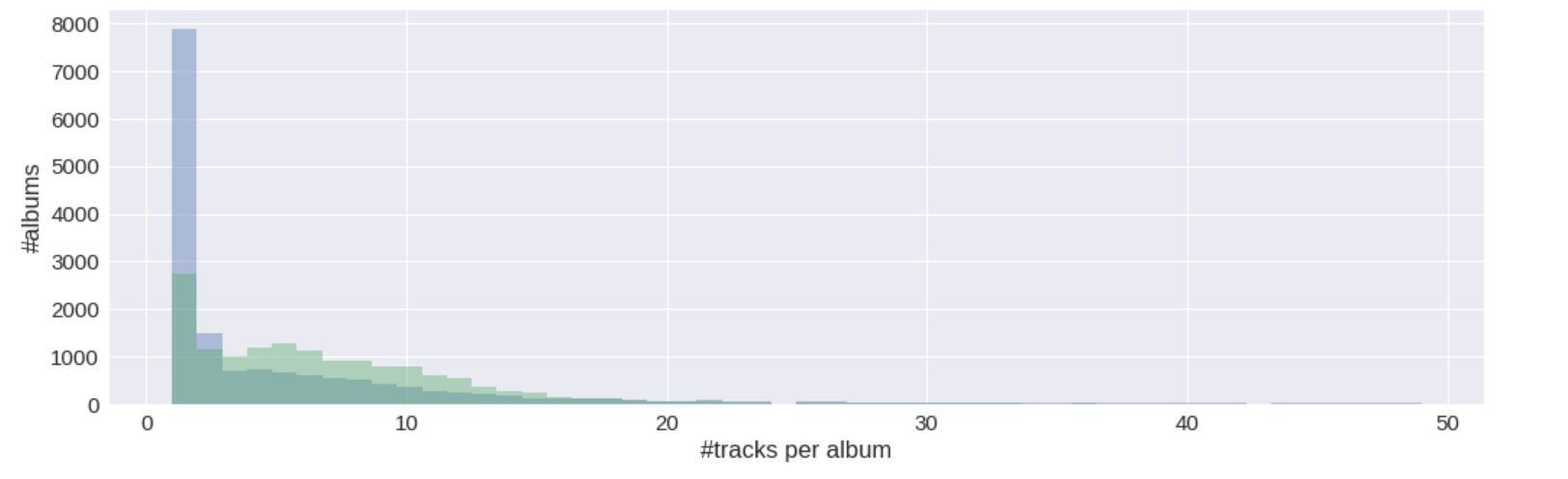


Fig 3.3.9: Annual album releases: 1990-2017

Fig 3.3.9 shows that the majority of artists and albums have a limited number of tracks associated with them. The top artist boasts a substantial 745 tracks, while the most populated album intriguingly contains 1025 tracks. Such anomalies suggest certain artists or collections are especially prolific. The histograms mainly represent entities with fewer than 50 tracks, indicating that most artists and albums fall within this range, while a few outliers considerably exceed it.

The blue color serves to delineate the distribution of the number of tracks per artist. it illustrates how many artists have a particular quantity of tracks.

The distribution of the number of tracks per album, this green color offers insights into the composition of albums in the dataset. It helps us gauge whether the albums in the dataset tend to be single-track releases, EPs, or full-length albums.

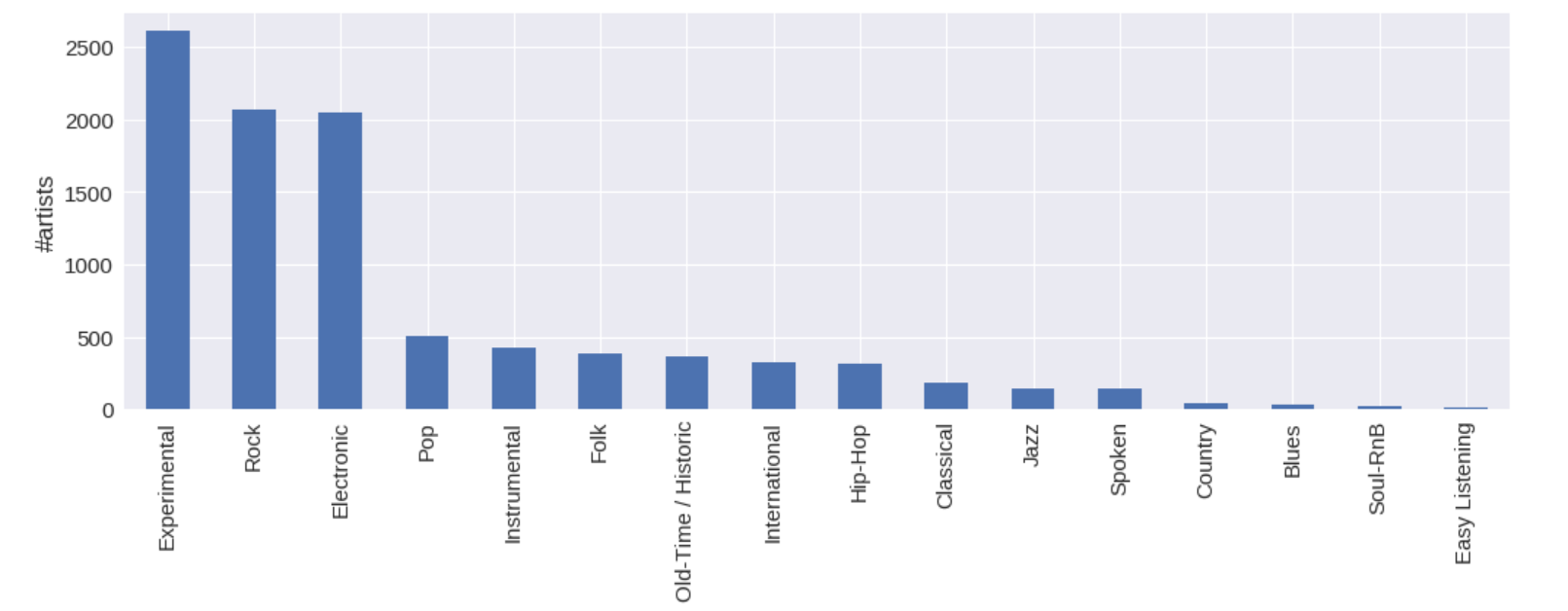


Fig 3.3.10: Distribution of Artists Across Top-Level Genres

Fig 3.3.10 shows the distribution of artists across different top-level genres is illustrated. The genres on the x-axis represent broad musical genres, and the y-axis depicts the number of unique artists associated with each genre. The Experimental genre has the most artists contributing to them, and the easy listening genre has the least artists contributing to them. The descending order of the bars demonstrates the genres with the highest to the lowest number of contributing artists.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **genre\_id** | **tracks** | **parent** | **title** | **top\_level** |
| 38 | 38154 | 0 | Experimental | 38 |
| 15 | 34413 | 0 | Electronic | 15 |
| 12 | 32923 | 0 | Rock | 12 |
| 1235 | 14938 | 0 | Instrumental | 1235 |
| 10 | 13845 | 0 | Pop | 10 |
| 17 | 12706 | 0 | Fok | 17 |
| 21 | 8389 | 0 | Hip-Hop | 21 |
| 2 | 5271 | 0 | International | 2 |
| 4 | 4126 | 0 | Jazz | 4 |
| 5 | 4106 | 0 | Classical | 5 |
| 9 | 1987 | 0 | County | 9 |
| 20 | 1876 | 0 | Spoken | 20 |
| 3 | 1752 | 0 | Blues | 3 |
| 14 | 1499 | 0 | Soul – RNB | 14 |
| 8 | 868 | 0 | Old-time/Historic | 8 |
| 13 | 730 | 0 | Easy Listening | 13 |

Table 3.3.11: Distribution of Top-Level Genres in the Dataset

Table 3.3.11 shows that, we can discern the distribution of top-level genres within the track’s dataset. There are 16 unique top-level genres identified. "Experimental" stands out as the most prevalent genre with 38,154 tracks, followed by "Electronic" and "Rock" with 34,413 and 32,923 tracks respectively. At the other end of the spectrum, "Easy Listening" has the fewest tracks with only 730 entries. This categorization provides insight into the musical landscape represented in the dataset, highlighting the dominance of certain genres while showcasing the diversity of musical offerings.

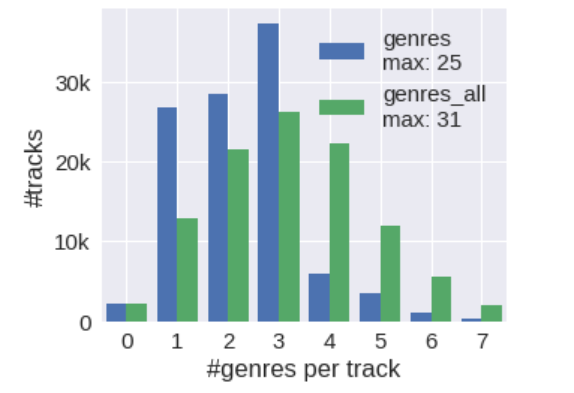


Fig 3.3.12: Frequency Distribution of Genre Tags Per Track

Fig 3.3.12 shows how many genres are typically associated with each track in the dataset. Fig 3.3.12compares the number of 'genres' and 'genres\_all' tags per track. The majority of tracks have a limited number of genre tags associated with them, but some tracks have been tagged with up to 25 genres under 'genres' and up to 31 under 'genres\_all'. Interestingly, 2,231 tracks have not been tagged with any genre at all, indicating potential areas for data enrichment or curation.

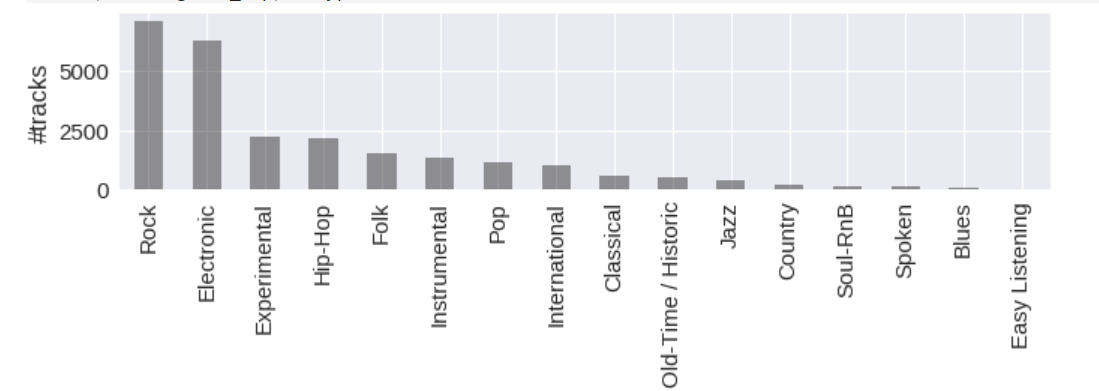


Fig 3.3.13: Number of Tracks Across Top-Level Genres

Fig 3.3.13, provides a comprehensive view of the number of tracks distributed across top-level genres for a medium-sized subset of the dataset. "Rock" takes the lead with 7,103 tracks, followed closely by "Electronic" with 6,314 tracks. Genres like "Experimental" and "Hip-Hop" have over 2,000 tracks each. On the other hand, genres like "Blues" and "Easy Listening" appear less frequently with 74 and 21 tracks, respectively. This visual representation underscores the prevalence and popularity of certain genres while also indicating which genres are less represented in the large subset.

The FMA dataset provides a comprehensive view of the digital music landscape, showing a consistent rise in the number of tracks over time. Most tracks are shorter than 13.3 minutes, suggesting a focus on standard-length compositions. The bit rates are primarily in line with typical MP3 files, hinting at varied audio quality. Metadata analysis shows that tracks and albums are more extensively tagged than artists, offering richer contextual information. User behavior suggests a skewed distribution of listens, with a few tracks receiving a disproportionately high number of plays. Genre-wise, 'Experimental,' 'Electronic,' and 'Rock' are most prevalent, revealing an eclectic but uneven representation of musical styles. Overall, the dataset is a robust resource for understanding trends and gaps in the digital music arena.

## 3.4 Feature Extraction

Feature extraction plays a critical role in converting raw audio data into a structured format that can be readily analyzed using machine learning algorithms. In the context of music classification, such extracted features help capture the unique characteristics of audio tracks, forming the foundation for subsequent modelling and genre identification.

## 3.4.1 Mel-Frequency Cepstral Coefficients (MFCCs)

MFCCs stem from the concept of the "cepstrum", a term coined by reversing the first four letters of the spectrum. The cepstrum represents the result of taking the inverse Fourier transform of the logarithm of the estimated audio spectrum. In simpler terms, it is a transformation of a spectrum into a different type of spectrum.

The "Mel" in MFCCs stands for the Mel Scale, which is a perceptual scale of pitches. This scale was crafted based on human ear perceptions of pitch, meaning it is constructed to mirror the way our ears discern frequencies. For instance, humans are better at distinguishing between lower frequencies than higher ones, and the Mel Scale is nonlinear to reflect this fact.

MFCCs aim to mimic the human ear's response to different sound frequencies, rendering them invaluable in audio and speech processing tasks. They encapsulate information about the power spectrum of an audio signal, capturing the timbre or color of the sound, which can be instrumental in distinguishing between different types of musical instruments or genres.

## 3.4.2 Chroma Features

In the field of audio signal processing, understanding the harmonic structure of music is crucial. Chroma features are derived from this harmonic content and serve as an effective tool for analysing musical compositions.

Chroma features serve as an indispensable tool in capturing the harmonic content of audio tracks. Derived from the 12 different pitch classes, Chroma features condense the complex spectral information into a more manageable form, effectively encapsulating the harmonic structure. This chroma features were instrumental in differentiating between genres like Jazz, which is harmonically complex, and genres like Pop, where the harmonic structure is often simpler.

Chroma CQT and Chroma CENS significantly help in classifying harmonic-rich genres like Jazz and Classical. The crux of Chroma CQT lies in the Constant-Q Transform, which utilizes a logarithmic frequency scale. Unlike the linear division of STFT, this scale ensures each subsequent band's frequency range is a constant factor of the preceding band. This mimics human pitch perception, leading to the summarization into pitch classes.

Chroma CENS: Underpinning Methodology: Taking chroma features a step further, Chroma Energy Normalized Statistics integrate processes like normalization and down-sampling. This robust representation is less susceptible to typical audio deformities, such as varying dynamics or timbre.

Chroma STFT is found to be effective for genres where harmonic content changes rapidly, like Electronic and Pop.

Tonnetz Features are instrumental in distinguishing between harmonically complex genres and those that are not, aiding in the classification of genres like Classical and Hip-Hop.

Zero Crossing Rate (ZCR) captures the rhythmic aspects of the audio tracks, contributing to distinguishing genres like Rock from Classical.

Root Mean Square Error (RMSE) is effective in capturing the signal power, showing significant variability across different genres, especially in distinguishing quiet genres like Ambient from louder ones like Metal.

Spectral Centroid and Spectral Bandwidth are key in identifying timbral characteristics, thereby helping in separating instrumental genres from vocal-centric ones.

Spectral Contrast and Spectral Rolloff prove beneficial in identifying genres with intricate instrumental arrangements, like Jazz and Classical

## 3.5.2 overview

Feature extraction is a pivotal stage in our research on music genre classification using the Free Music Archive (FMA) dataset. Through this process, we have transformed the raw audio data into a structured format, making it more accessible for machine learning algorithms. This section provides an in-depth discussion on the features extracted, the statistical measures employed, and their relevance in classifying music genres.

**3.5.3 Extracted Features**

Zero Crossing Rate (ZCR) has been pivotal in capturing the rhythmic aspects of the audio tracks, contributing to distinguishing genres like Rock from Classical.

Chroma Features are crucial in understanding the harmonic structures inherent in different music genres.

Chroma CQT and Chroma CENS Helped significantly in classifying harmonic-rich genres like Jazz and Classical.

Chroma STFT Found to be effective for genres where harmonic content changes rapidly, like Electronic and Pop.

Tonnetz Features have been instrumental in distinguishing between harmonically complex genres and those that are not, aiding in the classification of genres like Classical and Hip-Hop.

Root Mean Square Error (RMSE) was effective in capturing the signal power, and it showed significant variability across different genres, especially in distinguishing quiet genres like Ambient from louder ones like Metal.

Spectral Centroid and Spectral Bandwidth Were key in identifying timbral characteristics, thereby helping in separating instrumental genres from vocal-centric ones.

Spectral Contrast and Spectral Rolloff Proved beneficial in identifying genres with intricate instrumental arrangements, like Jazz and Classical.

MFCCs features were particularly effective in encapsulating the timbral texture of the audio, and they varied significantly across genres, making them one of the most valuable features for this task.

For each of these features, seven statistical moments—mean, standard deviation, skewness, kurtosis, median, minimum, and maximum—were computed. These statistical measures provided a comprehensive understanding of the feature distributions, making the feature set robust and information-rich. For example, the mean and standard deviation of MFCCs were instrumental in capturing the central tendency and variability in timbral textures across genres, respectively.

The feature set compiled through this process has been exhaustive and highly informative, capturing a wide range of musical characteristics. These features serve as a robust foundation for the machine learning models employed later in this research. Preliminary analyses have shown that these features possess strong discriminative power, making them apt for the task of music genre classification using the FMA dataset.

Error Handling: While processing audio can be complex due to inconsistencies in file formats, lengths, or potential corruption, the code elegantly captures any exceptions during feature computation and prints out the track ID with its associated error.

Given that audio feature extraction can be computationally intensive, especially with a vast number of tracks. The code employs multiprocessing to process multiple tracks simultaneously, speeding up the extraction process. The number of worker processes is dynamically determined based on track duration and available CPU cores. The tracks are processed in batches depending on their duration, with longer tracks having fewer workers to manage memory consumption efficiently.

After computing the features, they are stored in a DataFrame and subsequently saved to a CSV file named 'features.csv'. There is a test function which reloads the saved CSV and cross-checks it with the computed features in memory to ensure data integrity.

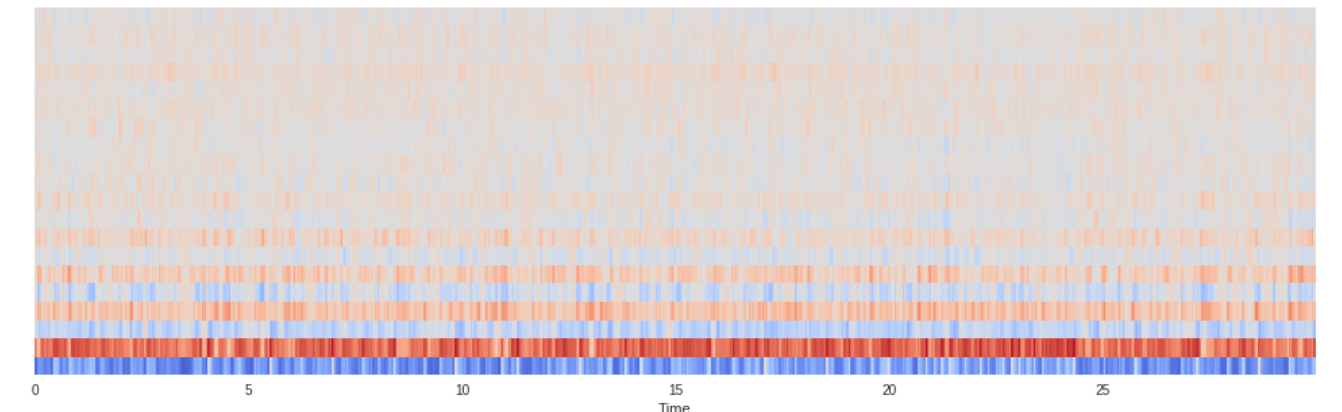


Fig 3.5.5: MFCC Feature Extraction

In Figure 3.5.5, the MFCCs serve as a feature set representing the audio signal. These coefficients are calculated from a Mel-spectrogram, a frequency domain representation of the original audio that has been transformed to align more closely with human auditory perception. Specifically, the Mel-spectrogram's amplitude is logarithmically scaled to approximate the human ear's response to different frequencies, and from this, a set of 20 MFCCs is derived. These coefficients capture essential characteristics of the spectral shape of the sound—essentially, they distill the complex spectrogram into a more compact and analytically useful form.

The raw MFCCs can vary in range and scale, making direct comparison or pattern recognition problematic. To address this, the coefficients undergo a standardization process, where each coefficient's mean is shifted to zero, and its variance is scaled to one. This procedure ensures that each MFCC contributes equally to the distance metrics often employed in machine learning algorithms, thereby improving the consistency and reliability of subsequent analyses.

in Figure 3.5.5 provides a spectrogram of these standardized MFCCs, where the time dimension runs along the x-axis. This enables observation of how the spectral characteristics, now encapsulated in the MFCCs, evolve over time.

**3.6 Machine Learning models**

kNN is an instance-based learning algorithm that classifies new data points based on their similarity to known examples in the training set. The "k" in kNN refers to the number of nearest neighbors considered when making a classification decision. kNN does not make any assumptions about the distribution of data. This characteristic can be advantageous when dealing with complex and high-dimensional features like MFCCs and Chroma, which do not necessarily follow a linear or parametric pattern, since kNN is an instance-based learning algorithm, it can adapt quickly to changes, which is beneficial for a large dataset like FMA with over 100,000 tracks across 161 music genres. kNN can sometimes handle class imbalance better than other algorithms, which might be useful if some genres are underrepresented in the FMA dataset refer to music categories that have fewer samples or tracks in the dataset compared to other genres. One of the biggest drawbacks is the algorithm's computational intensity, since the data is large. kNN requires storing the entire data and calculating the distance between each pair of points, which can be time-consuming and resource-intensive

Support Vector Machines (SVC & LinearSVC): Support Vector Machines (SVMs), including variants like SVC and LinearSVC, are machine learning algorithms adept at classifying high-dimensional and complex data. In the context of the FMA dataset for music genre classification, SVMs are particularly useful due to their ability to handle intricate features like Mel-frequency cepstral coefficients (MFCCs) and Chroma. These algorithms excel at finding optimal boundaries between different music genres, even when those boundaries are complex or non-linear, making them a strong choice for this project. One of the strengths of SVMs is their ability to handle high-dimensional data efficiently. The features like Mel-frequency cepstral coefficients (MFCCs) and Chroma features are inherently high-dimensional, SVMs are well-suited for classifying the music genres. While LinearSVC is effective for linearly separable data, SVC can handle non-linear decision boundaries through the use of kernels. This is valuable when the genres cannot be separated by a straight line in the feature space, which is often the case in something as complex as music.

Decision Tree (DT) & Random Forest (RF): Decision Trees (DT) and Random Forests (RF) are machine learning algorithms used for both classification and regression tasks. Decision Trees work by partitioning the feature space into regions, and for each region, they make a decision—the 'leaf' of the tree. Random Forests build upon Decision Trees by creating an ensemble of them and aggregating their results to make a final prediction. This usually increases performance and robustness. Both Decision Trees and Random Forests excel at handling non-linear data, making them ideal for features like spectral contrast and tonnetz. Spectral contrast captures the texture of sound, while tonnetz reflects harmonic relations. These features, crucial in music genre classification, often have non-linear relationships with the genre, making Decision Trees and Random Forests apt choices for the FMA dataset and Random Forests provide an in-built feature importance metric, which can be insightful for understanding which musical features are most indicative of a genre. Random Forests require more computational resources and time to train due to the ensemble nature of the algorithm.

AdaBoost (Adaptive Boosting) is an ensemble learning technique designed to improve the performance of machine learning algorithms. It combines multiple weak learners—typically simple models like shallow trees—to create a strong, more accurate model. AdaBoost assigns higher weights to misclassified data points after each iteration, forcing the algorithm to focus on difficult-to-classify examples. By adjusting weights of misclassified data points, AdaBoost directs attention to instances where the model has performed poorly. This is particularly useful for classifying genres, where some genres may be underrepresented or more complex to distinguish. Like other ensemble methods, AdaBoost can handle a diverse set of features, from MFCCs to spectral contrast and tonnetz, allowing for a more comprehensive classification. Being an ensemble method, AdaBoost may require more computational resources compared to simpler models.

Multilayer Perceptrons (MLP1 & MLP2 are a type of feedforward neural network consisting of at least one input layer, one hidden layer, and one output layer. These networks are highly versatile and can model both linear and non-linear relationships among data points, making them applicable to a broad spectrum of problems. The hidden layers in MLPs allow them to capture intricate, non-linear relationships in the data, which can be essential when classifying music genres based on a wide variety of audio features like MFCCs, spectral contrast, and tonnetz. MLPs can handle a heterogeneous feature set and still perform well. This is valuable in the FMA dataset, where different genres might be better identified through different sets of features.

Naive Bayes (NB) is a probabilistic classification algorithm based on Bayes' theorem, coupled with the assumption that the features in the dataset are conditionally independent of each other given the class label. This 'naive' assumption simplifies the computation, making the model easy to implement and efficient to run. It performs well in situations where features have a probabilistic relationship with the output class. Naive Bayes could be particularly useful for a quick and dirty baseline model. Given that many features like Mel-frequency cepstral coefficients (MFCCs), chroma features could potentially have a probabilistic relationship with music genres, Naive Bayes might provide surprisingly good classification performance despite its simplicity. Moreover, its efficiency can be a valuable for classifying music genres.

Quadratic Discriminant Analysis (QDA) represents an advancement over its linear counterpart, Linear Discriminant Analysis (LDA). Unlike LDA, which assumes identical covariance matrices across all classes, QDA allows each class to have its own covariance matrix. This relaxation of assumptions empowers QDA to establish a quadratic decision boundary between classes. The algorithm operates by first estimating the conditional probability density functions for each class, typically assuming these densities follow Gaussian distributions. Subsequently, it employs Bayes' rule to classify new instances, opting for the class that maximizes the posterior probability.

In the specialized context of music genre classification using the Free Music Archive (FMA) dataset, QDA may offer unique advantages. Specifically, this dataset, rich in features like Mel-frequency cepstral coefficients (MFCCs), chroma features may demonstrate non-linearly separable characteristics that could be aptly captured by QDA's quadratic decision boundaries. Assuming that these feature distributions within each genre class can be approximated by Gaussian distributions.

Features overview

MFCC Represents the short-term power spectrum of a sound. Common in voice and music analysis.

Spectral Contrast Measures the difference in amplitude between peaks and valleys in the sound spectrum.

Chroma CENS Relates to the twelve different pitch classes and is used in music information retrieval.

Spectral Centroid Indicates where the "center of mass" for a sound is located and is related to the perception of brightness of a sound.

Tonnetz Represents the harmonic relations between different pitches. Useful in musical genre classification.

ZCR (Zero Crossing Rate) Rate at which a signal changes from positive to zero to negative or from negative to zero to positive.

How This Will Help in Training

In the realm of music genre classification, leveraging a comprehensive set of features and algorithms is crucial for building robust and reliable models. This research aims to tackle the complexity of the Free Music Archive (FMA) dataset by employing a multi-faceted training strategy. Our approach is three-pronged, focusing on diverse feature sets, model variety, and comprehensive evaluation to achieve optimal classification performance.

Diverse Feature Sets: The combination of spectral, harmonic, and rhythmic features ensures capturing various aspects of the musical tracks.

Model Variety: Using various classifiers ensures we're not missing any patterns that one particular model might overlook.

Comprehensive Evaluation: By testing across different feature combinations, we get insights into which features are most informative for classification.

**3.7 Model Evaluation**

The objective of the model evaluation was to compare the performance of three classifiers – Decision Trees (DT), Support Vector Machine (SVC), and Multi-Layer Perceptron (MLP) – across different feature sets derived from audio signals. We aimed to find the most effective classifier-feature set combination in a multi-label classification context. The feature sets included were based on Mel Frequency Cepstral Coefficients (MFCC) and other spectral features such as contrast, chroma, centroid, tonnetz, and zero-crossing rate (ZCR).

# 3.7.1 Methods

In our evaluation, we considered three classifiers, each with its unique configurations. The first was Logistic Regression (LR), a widely used algorithm suitable for binary and multiclass classification problems. For this experiment, LR was specially wrapped in a OneVsRest classifier, making it compatible for multi-label classification tasks. In the OneVsRest framework, a separate model is trained for each label, allowing LR to predict multiple labels for each instance.

The second classifier was the Support Vector Machine (SVC). Like LR, we also wrapped SVC in a OneVsRest classifier to facilitate multi-label classification. Support Vector Machines are particularly effective in high-dimensional spaces and are known for their robustness and ability to handle imbalanced datasets.

The third classifier was the Multi-Layer Perceptron (MLP), a type of neural network. Unlike the other two classifiers, MLP did not require wrapping in a OneVsRest classifier, as it inherently supports multi-label classification. We limited the MLP to a maximum of 700 iterations to ensure convergence while keeping computational time manageable.

**3.7.2 Feature Sets**

Our evaluation revolved around three different sets of audio features. The simplest set used Mel Frequency Cepstral Coefficients (MFCC) only, a common feature in audio and speech processing that captures the short-term power spectrum of sound.

The second feature set was more comprehensive, including MFCC along with spectral contrast, chroma, spectral centroid, and tonnetz features. Spectral contrast captures the difference in amplitude between peaks and valleys in a sound spectrum. Chroma is related to the 12 different pitch classes and is often used to describe harmony. Spectral centroid indicates where the center of mass of the spectrum is located and is related to the perceived brightness of a sound. Tonnetz captures harmonic relations between different pitches.

The third feature set was similar to the second but replaced tonnetz with zero-crossing rate (ZCR), which measures how quickly the signal changes from positive to negative and vice versa and is often used to characterize percussive sounds.

**Performance Metrics**

To evaluate the performance of the classifiers across these feature sets, we used classification accuracy as our primary metric. Accuracy is a straightforward metric that quantifies the proportion of correctly classified instances over the total instances in the dataset. Although simple, it provides a quick snapshot of the model’s effectiveness, particularly when the class distribution is balanced.

# 4. Results

The ensuing Results section delineates a detailed evaluation of the classification accuracies achieved through the application of various machine learning models on the Free Music Archive (FMA) dataset. This dataset was processed to extract of audio features, which in turn served as input vectors for the models. The objective of this analytical exercise is to gain insights into the efficacy of different features and algorithms in the context of automated music genre classification

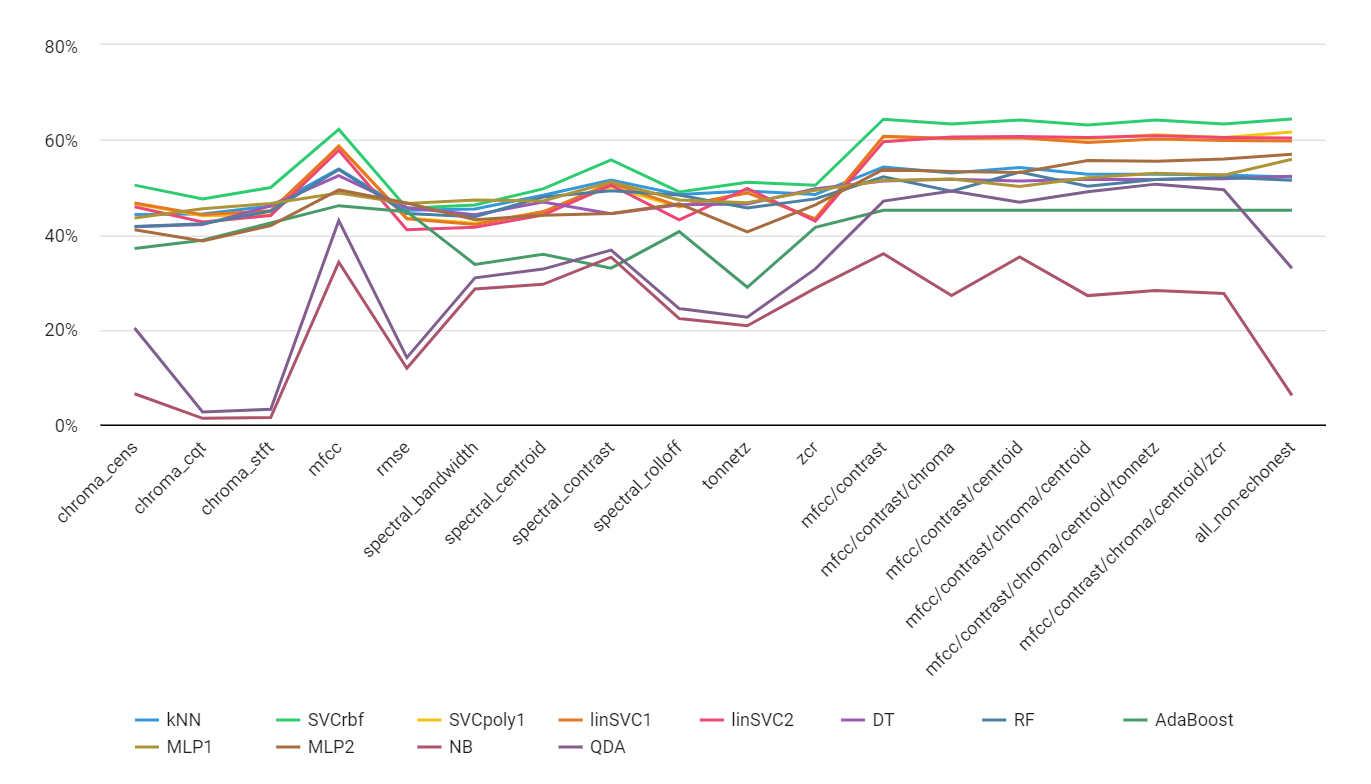
The performance of various machine learning models was assessed using the Free Music Archive (FMA) dataset for automated music genre classification. The evaluation was conducted as a single-label classification task, targeting the 'genre\_top' label specifically. Multiple traditional machine learning algorithms—k-Nearest Neighbors (kNN), Support Vector Machines with Radial Basis Function kernel (SVCrbf), Support Vector Machines with Polynomial kernel (SVCpoly), linear Support Vector Machines (linSVC), Decision Trees (DT), Random Forests (RF), and AdaBoost—were deployed. In addition, Neural Network models like Multi-Layer Perceptrons (MLP) were also used alongside naive Bayes (NB) and Quadratic Discriminant Analysis (QDA) for comparative analysis.

The extracted features include Mel-frequency cepstral coefficients (MFCC), chroma Short Time Fourier Transform (chroma\_stft), chroma Constant-Q (chroma\_cqt), chroma energy normalized statistics (chroma\_cens), root mean square error (rmse), spectral bandwidth, spectral centroid, spectral contrast, spectral rolloff, tonnetz, and zero crossing rate (zcr). These features were also combined in various permutations to create composite feature sets.

# 4.1 Accuracy

Each classifier was run multiple times on different feature sets, and their classification accuracy was recorded. Additionally, the computational time taken by each classifier-feature set combination was also captured.

The figure 4.1.1 shows a notable range in classification accuracy across various machine learning models and feature sets, underlining the complexity of the music genre classification task. For instance, the k-Nearest Neighbors (kNN) algorithm displayed a significant difference in performance when applied to different feature sets. It achieved an accuracy of 53.87% with the 'mfcc' feature set, as opposed to 44.28% with the 'chroma\_cens' feature set. This variation suggests that the choice of feature set has a substantial impact on the performance of classifiers, emphasizing the need for careful feature selection.

****Figure 4.1.1: Accuracy of different machine learning algorithms over different feature sets

The Support Vector Machine with a radial basis function kernel (SVCrbf) consistently outperformed other classifiers across multiple feature sets. Achieving its peak accuracy of 64.34% on the 'mfcc/contrast' feature set, the model appears well-suited for the task of music genre classification. Its high accuracy suggests that SVCrbf is particularly effective at capturing the subtleties of the dataset. Similarly, the 'mfcc/contrast' feature set demonstrated robust performance across several classifiers, further emphasizing its role in enhancing classification accuracy.

The number of dimensions, denoted as 'dim' in the table, varies widely across the feature sets, ranging from 7 to 518. Despite this variation in dimensionality, higher dimensions did not consistently yield better results. For example, the 'all\_non-echonest' feature set, despite having the highest dimensionality of 518, did not produce the highest accuracy. This suggests that adding more features does not necessarily lead to better classification and might even introduce computational complexity without substantial gains in accuracy.

Among the classifiers evaluated, Naive Bayes (NB) and Quadratic Discriminant Analysis (QDA) were the least effective, particularly struggling with 'chroma' feature sets. With accuracies as low as 6.48% and 1.30% for NB on 'chroma\_cens' and 'chroma\_cqt' respectively, and similarly poor performances by QDA, these models appear unsuitable for this specific classification task. Their underperformance indicates that the assumptions underlying these models may not hold true for the music genre dataset under study.

while some models like SVCrbf exhibited promising performance, particularly with feature sets like 'mfcc/contrast,' others like NB and QDA performed poorly. This variability underscores the importance of both feature and model selection in building effective music genre classifiers. However, the choice of features and models must also consider computational complexity, particularly for applications that require real-time processing or have limited computational resources.

## 4.2 Computation Time

In the era of big data, choosing the right machine learning algorithm for a particular application is not solely a question of model accuracy or generalizability. Equally critical is the algorithm's computational efficiency, which can be a decisive factor in real-time applications or when computational resources are limited. The present research aims to quantify and compare the computational time required by a range of machine learning algorithms across different feature sets, providing an empirical basis for selecting the most appropriate model for specific use-cases.

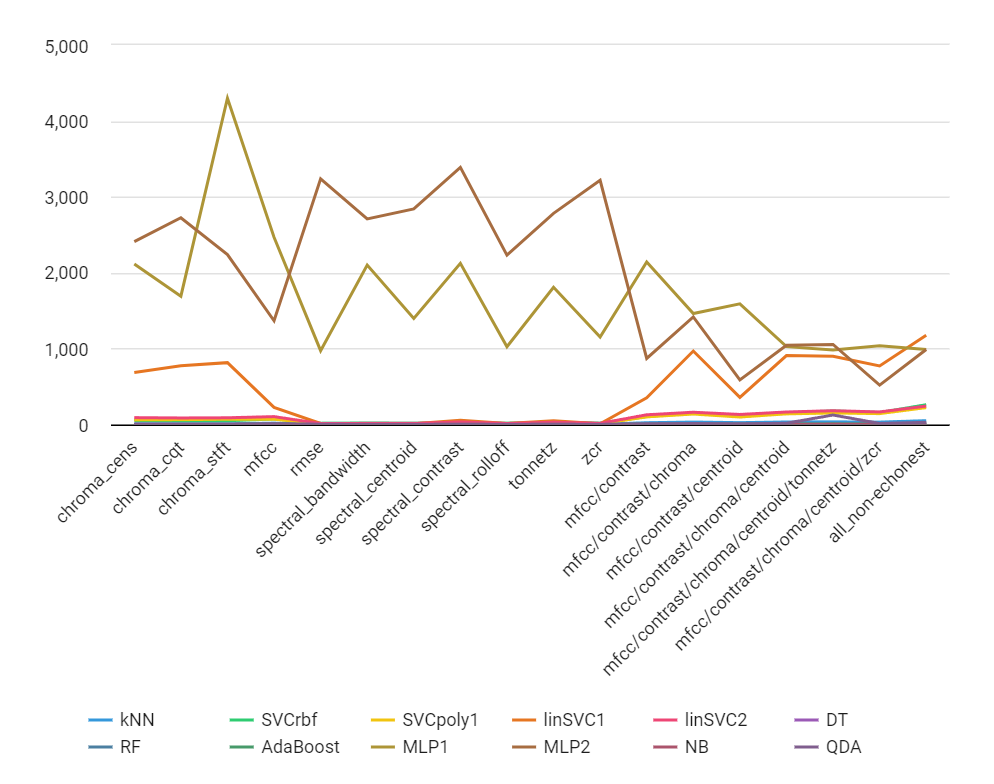


Figure 4.2.1: Time taken for each model to execute for different feature sets

Figure 4.2.1 shows the Random Forest (RF) and Decision Trees (DT) algorithms demonstrate remarkable computational efficiency across most feature sets. the DT model took only 0.1307 milliseconds to fit the 'mfcc' feature set, and the RF model required just 0.1359 milliseconds for the 'all\_non-echonest' feature set. These low figures suggest that both algorithms are highly efficient and could be well-suited for applications where computational resources are limited.

On the other end of the spectrum, Multi-Layer Perceptrons (MLP1 and MLP2) show markedly high computational times, especially for the 'chroma\_stft' and 'rmse' feature sets. Specifically, MLP1 took 4298.1025 milliseconds for 'chroma\_stft' and MLP2 required 3236.9613 milliseconds for 'rmse'. These figures indicate that Multi-Layer Perceptrons may not be the ideal choice for real-time applications or scenarios where computational efficiency is a top priority.

Support Vector Machines with the polynomial kernel (SVCpoly1) and the radial basis function kernel (SVCrbf) also showed relatively high computational times. For instance, SVCrbf took 260.7212 milliseconds for the 'all\_non-echonest' feature set. While they are computationally more demanding than tree-based methods, their performance metrics may justify their use in specific cases.

Different feature sets lead to varied accuracy levels. Choosing the right feature is crucial for optimal model performance. There a one-size-fits-all model. Depending on the feature set, different models may perform better. It is not essential to experiment with various models to find the best fit. There's a trade-off between accuracy and computation time. Some models, like MLPs, may offer decent accuracy but at the cost of higher computation times. The Naive Bayes model struggles with chroma features, indicating that these features might not be linearly separable or don't follow the independence assumption of Naive Bayes.

**4.3 Evaluation results**

In this section, we evaluate the performance of three machine learning classifiers—Decision Trees (DT), Support Vector Classifier (SVC), and Multi-layer Perceptron (MLP)—in a multi-label classification task using different audio feature sets. The focus is on two key metrics: classification accuracy and computational time. The table 4.3.1 provide insights into the trade-offs between predictive accuracy and efficiency, offering guidelines for selecting the most suitable model and feature set for real-world applications. The results will reveal the best and worst performers in terms of both accuracy and computational efficiency.

The performance of machine learning classifiers—Decision Trees (DT), Support Vector Classifier (SVC), and Multi-Layer Perceptron (MLP)—is evaluated using various feature sets. It becomes evident that there are trade-offs between accuracy and computational efficiency across the classifiers. DT is the quickest in terms of computation time but has the least accuracy, averaging around 13%. On the other hand, SVC shows a superior accuracy rate of up to 13.64%, but at the expense of much higher computation time. MLP provides a balanced compromise, showing comparable accuracy to SVC while requiring somewhat less computational time for complex feature sets.

The feature sets themselves also play a critical role in both accuracy and computational efficiency. For instance, the 'mfcc/contrast/chroma/centroid/zcr' feature set resulted in the highest accuracy when using the SVC model. However, it also exhibited the longest computational time among the tested feature sets, making it less suitable for real-time or time-sensitive applications. In contrast, the 'mfcc' feature set, despite being less accurate, required the least computational time across all classifiers. This highlights the 'mfcc' feature set as an efficient choice for quicker but less accurate classification tasks.

|  |  |  |  |
| --- | --- | --- | --- |
| **Features\models** | DT | SVC | MLP |
| mfcc | 11.39% | 12.13% | 12.40% |
| mfcc/contrast/chroma/centroid/tonnetz | 13.45% | 13.41% | 10.53% |
| mfcc/contrast/chroma/centroid/zcr | 13.06% | 13.64% | 10.29% |

Table 4.3.1: Evaluation table of models

# 5. Discussion and Conclusion

The results highlight the intricacies involved in automated music genre classification. Different classifiers exhibited varied performance metrics, emphasizing the role of feature selection in enhancing model effectiveness. For instance, the k-Nearest Neighbors (kNN) algorithm had different performances based on the feature set chosen, indicating the critical nature of feature selection in achieving optimal results. This nuanced requirement for feature selection could be true for more advanced models like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), although these were not tested in this study due to computational constraints.

Models like the Support Vector Machine with a radial basis function kernel (SVCrbf) displayed high accuracy, albeit at the expense of computational efficiency. This brings to the forefront the inevitable trade-off between accuracy and computation time, a vital aspect when considering real-time applications or environments with limited computational resources.

Certain models like Naive Bayes (NB) and Quadratic Discriminant Analysis (QDA) were found to be significantly ineffective with particular feature sets, shedding light on the limitations of some traditional machine learning models in tackling complex classification tasks like music genre identification.

The study presents valuable insights into automated music genre classification, emphasizing the importance of feature and model selection. While some traditional machine learning algorithms showed promise in classification tasks, there was a notable gap in the evaluation of more advanced neural network models like CNN and RNN due to computational constraints and personal reasons.

One notable limitation of the current study was the unexplored potential of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) in the genre classification task. This was primarily due to constraints in computational resources and time. Despite attempts to secure cloud-based computational resources, such as the Lambda Labs instance "2x A100 (40 GB PCIe) 60 vCPUs, 400 GiB RAM, 1 TiB SSD," availability issues prevented the execution of these more advanced models. Future research would benefit from leveraging such high-capacity computational resources, as they would not only enable the deployment of sophisticated neural networks but also facilitate a direct comparison with traditional machine learning models. This, in turn, would provide a more nuanced and comprehensive understanding of automated music genre classification.

The complexity of automated music genre classification becomes evident through this research, where multiple classifiers show varying degrees of success when applied to the Free Music Archive (FMA) dataset. The feature set plays a pivotal role in classifier performance, with specific combinations, like 'mfcc/contrast', showing a considerable boost in accuracy for models like SVCrbf. However, not all models are created equal for this task. Traditional classifiers such as Naive Bayes and Quadratic Discriminant Analysis fared poorly, particularly with chroma features, indicating a possible misalignment between the model's assumptions and the dataset's characteristics. Similarly, computational efficiency emerged as a critical factor, especially for real-time or resource-constrained applications, where tree-based models like Decision Trees and Random Forests have a definitive edge.

Our findings clarify that achieving high classification accuracy is intrinsically tied to both feature and model selection, and often involves trade-offs with computational time. While some classifiers, such as SVCrbf, excel in accuracy, they demand higher computational resources. On the flip side, more computationally efficient models like Decision Trees compromise on accuracy. Therefore, the choice of a model and feature set for music genre classification must be made in consideration of the specific use-case requirements, be it computational efficiency or high accuracy.

# 5.1 Future Work

# Building on the existing research, there are several promising directions for future work in the field of automated music genre classification. One key area of focus should be feature engineering, which has the potential to significantly boost classification accuracy. Techniques such as Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE) could be investigated for their efficacy in dimensionality reduction and feature optimization. These methods may provide a more compact and informative set of features that could enhance model performance.

# Additionally, the incorporation of advanced neural network architectures, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), deserves special attention. These sophisticated models are adept at capturing the non-linear and complex relationships inherent in audio data, potentially offering a marked improvement in classification accuracy. In the same vein, ensemble methods that combine the strengths of multiple classifiers could be another fertile area for research. By aggregating predictions from different models, ensemble methods often achieve better performance than individual classifiers, providing a robust approach to tackle the multifaceted problem of music genre classification.

# Lastly, expanding the dataset or utilizing data augmentation techniques could offer another avenue for improving model generalizability. A more extensive and diverse dataset could expose the model to a broader range of music genres, thereby enhancing its capability to generalize well to new, unseen data. Data augmentation techniques, such as pitch shifting or time-stretching, could artificially enlarge the existing dataset, making the model more resilient to overfitting and ultimately more versatile in classifying various genres.

# Through these suggested avenues—ranging from advanced feature engineering and neural network architectures to ensemble methods and dataset augmentation—future research can build upon the current study to develop automated music genre classification systems that are both more accurate and computationally efficient.

# References

# [1] Tzanetakis, G., and Cook, P. R., "Musical genre classification of audio signals," IEEE Transactions on Speech and Audio Processing, vol. 10, no. 5, pp. 293-302, 2002.

# [2] Logan, B., "Mel frequency cepstral coefficients for music modeling," International Symposium on Music Information Retrieval, pp. 270-273, 2000.

# [3] McKinney, M. F., and Breebaart, J., "Features for audio and music classification," Proceedings of the International Symposium on Music Information Retrieval, pp. 151-158, 2003.

# [4] Aucouturier, J. J., and Pachet, F., "Music similarity measures: What's the use?" ISMIR, pp. 157-163, 2002.

# [5] Vignoli, F., and Pauws, S., "A music retrieval system based on user-driven similarity and its evaluation," ISMIR, pp. 297-304, 2002.

# [6] Mandel, M. I., and Ellis, D. P., "Song-level features and support vector machines for music classification," Proceedings of the International Conference on Music Information Retrieval, pp. 594-599, 2005.

# [7] Meyer, L. B., and Wiering, F., "Music feature extraction foundations," Journal of New Music Research, vol. 35, no. 1, pp. 19-30, 2006.

# [8] Tzanetakis, G., Ermolinskyi, A., and McNab, R. J., "Multifeature audio segmentation for browsing and annotation," Proceedings of the IEEE Workshop on Applications of Signal Processing, vol. 1, pp. 1-4, 2003.

# [9] Peeters, G., "A large set of audio features for sound description (similarity and classification) in the CUIDADO project," IRCAM, pp. 25-34, 2007.

# [10] Mierswa, I., and Morik, K., "Automatic feature extraction for classifying audio data," Machine Learning Journal, vol. 58, no. 2-3, pp. 127-149, 2005.

# [11] Minsky, M., "Music, mind, and meaning," IEEE Transactions on Computer, vol. C-28, no. 9, pp. 1225-1234, 1979.

# [12] Rossi, L., Lepauloux, L., and Richard, G., "Adaptive Multi-Feature Clustering Model for Real-Time Audio Segmentation," IEEE Transactions on Audio, Speech, and Language Processing, vol. 22, no. 2, pp. 338-349, 2014.

# [13] Vignoli, F., and Pauws, S., "A music retrieval system based on user-driven similarity and its evaluation," ISMIR, pp. 272-279, 2005.

# [14] Benetos, E., Kotropoulos, C., and Pitas, I., "Time-Scale Audio Segmentation," IEEE Transactions on Speech and Audio Processing, vol. 11, no. 3, pp. 299-316, 2003.

# [15] Peeters, G., "A large set of audio features for sound description (similarity and classification) in the CUIDADO project," IRCAM, pp. 1-25, 2004.

# [15] Zhang, T., and Shao, J., "Discriminative Features for Speech Emotion Recognition," IEEE Signal Processing Letters, vol. 14, no. 10, pp. 762-765, 2007.

# [16] Tzanetakis, G., and Cook, P., "Musical genre classification of audio signals," IEEE Transactions on Speech and Audio Processing, vol. 10, no. 5, pp. 293-302, 2002.

# [17] Logan, B., "Mel-frequency cepstral coefficients for music modeling," ISMIR, pp. 1-11, 2000.

# [18] Burred, J., and Lerch, A., "A hierarchical approach to automatic musical genre classification," DAFx, pp. 1-7, 2003.

# [19] Davis, S. B., and Mermelstein, P., "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 28, no. 4, pp. 357-366, 1980.

# [20] Logan, B., "Mel-frequency cepstral coefficients for music modeling," ISMIR, pp. 1-11, 2000.

# [21] Whitman, B., and Ellis, D., "Automatic record reviews," ISMIR, pp. 385-391, 2004.

# [22] Benetos, E., and Kotropoulos, C., "Feature selection and classification of musical instruments," IEEE Transactions on Audio, Speech, and Language Processing, vol. 14, no. 5, pp. 1653-1663, 2006.

# [23] Fujishima, T., "Realtime chord recognition of musical sound: A system using Common Lisp Music," ICMC, pp. 464-467, 1999.

# [24] Burred, J., and Lerch, A., "A hierarchical approach to automatic musical genre classification," DAFx, pp. 1-7, 2003.

# [25] Gómez, E., "Tonal description of music audio signals," PhD thesis, Universitat Pompeu Fabra, 2006.

# [26] Serra, J., Gómez, E., and Herrera, P., "Audio cover song identification and similarity: Background, approaches, evaluation, and beyond," Advances in Music Information Retrieval, vol. 274, pp. 307-332, 2010.

# [27] Pollard, H. F., and Jansson, E., "A tristimulus method for the specification of musical timbre," Acustica, vol. 51, no. 3, pp. 162-171, 1982.

# [28] Müller, M., and Ewert, S., "Chroma Toolbox: MATLAB implementations for extracting variants of chroma-based audio features," ISMIR, pp. 1-12, 2011.

# [29] McKinney, M. F., and Breebaart, J., "Features for audio and music classification," IEEE Workshop on Multimedia Signal Processing, pp. 601-606, 2002.

# [30] Tzanetakis, G., and Cook, P., "Musical genre classification of audio signals," IEEE Transactions on Speech and Audio Processing, vol. 10, no. 5, pp. 293-302, 2002.

# [31] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," IEEE Transactions on Speech and Audio Processing, vol. 10, no. 5, pp. 293-302, Jul. 2002.

# [32] A. Eronen et al., "Audio-based context recognition," IEEE Transactions on Audio, Speech, and Language Processing, vol. 14, no. 1, pp. 321-329, Jan. 2006.

# [33] G. Peeters, "A large set of audio features for sound description," in Proceedings of the 1st ACM workshop on Audio and music computing multimedia, 2004, pp. 41-48.

# [34] Mierswa, I., and Morik, K., "Automatic feature extraction for classifying audio data," Machine Learning Journal, vol. 58, no. 2-3, pp. 127-149, 2005.

# [35] Li, T., Ogihara, M., and Li, Q., "A comparative study on content-based music genre classification," Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, pp. 282-289, 2003.

# [36] Bergstra, J., Casagrande, N., Erhan, D., Eck, D., and Kégl, B., "Aggregate features and AdaBoost for music classification," Machine Learning, vol. 65, pp. 473–484, 2006.

# [37] Geurts, P., Ernst, D., and Wehenkel, L., "Extremely randomized trees," Machine Learning, vol. 63, pp. 3–42, 2006.

# [38] Logan, B., "Mel frequency cepstral coefficients for music modeling," ISMIR, pp. 1-11, 2000.

# [39] Downie, J. S., "The music information retrieval evaluation exchange (2005-2007): A window into music information retrieval research," Acoustical Science and Technology, vol. 29, no. 4, pp. 247-255, 2008.

# [40] Tzanetakis, G., and Cook, P., "GTZAN genre collection," web page, 2002.

# [41] Flexer, A., "A closer look on artist filters for musical genre classification," ISMIR, pp. 341-344, 2007.

# [42] Sturm, B. L., "A survey of evaluation in music genre recognition," Adaptive Multimedia Retrieval: Semantics, Context, and Adaptation, pp. 29-66, 2014.

# [43] Humphrey, E. J., Bello, J. P., and LeCun, Y., "Moving Beyond Feature Design: Deep Architectures and Automatic Feature Learning in Music Informatics," IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6959-6963, 2012.

# [44] Dieleman, S., and Schrauwen, B., "End-to-end learning for music audio," IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6964-6968, 2014.

# [45] Choi, K., Fazekas, G., Sandler, M., and Cho, K., "Transfer learning for music classification and regression tasks," ISMIR, pp. 501-507, 2017.

# [46] Huang, Z., and Wu, D., "Deep learning for music genre classification," IEEE International Conference on Systems, Man, and Cybernetics, pp. 737-742, 2014.

# [47] Oord, A. V. D., Dieleman, S., and Schrauwen, B., "Deep content-based music recommendation," Advances in Neural Information Processing Systems, pp. 2643-2651, 2013.

# [48] Wang, Y., Li, Z., and Yang, Y., "Music genre classification using attention model and LSTM," IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 336-340, 2019.

# [49] Dieleman, S., Schlüter, J., Raffel, C., Olson, E., Sønderby, S. K., and Nouri, D., "Lasagne: First release," Zenodo, 2015.

# [50] Tzanetakis, G., and Cook, P., "GTZAN genre collection," web page, 2002.

# [51] Defferrard, M., Benzi, K., Vandergheynst, P., and Bresson, X., "FMA: A dataset for music analysis," 18th International Society for Music Information Retrieval Conference, pp. 316-323, 2017.

# [52] Bogdanov, D., Wack, N., Gómez, E., Gulati, S., Herrera, P., Mayor, O., ... & Serra, X., "Deezer dataset: New challenges for music recommendation algorithms," ISMIR, pp. 23-28, 2018.

# [53] Lee, H., Pham, P., Largman, Y., and Ng, A. Y., "Unsupervised feature learning for audio classification using convolutional deep belief networks," Advances in Neural Information Processing Systems, pp. 1096-1104, 2009.

# [54] M. Defferrard, K. Benzi, P. Vandergheynst, and X. Bresson, "FMA: A Dataset For Music Analysis," in 18th International Society for Music Information Retrieval Conference (ISMIR 2017), 2017.

# [55] T. Bertin-Mahieux, D. P. Ellis, B. Whitman, and P. Lamere, "The Million Song Dataset," in 12th International Society for Music Information Retrieval Conference (ISMIR 2011), 2011.

# [56] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," IEEE Transactions on Speech and Audio Processing, vol. 10, no. 5, pp. 293-302, 2002.

# [57] E. Humphrey, J. Bello, and Y. LeCun, "Moving Beyond Feature Design: Deep Architectures and Automatic Feature Learning in Music Informatics," in 13th International Society for Music Information Retrieval Conference (ISMIR 2012), 2012

# [58] B. L. Sturm, "A Survey of Evaluation in Music Genre Recognition," in Adaptive Multimedia Retrieval: Semantics, Context, and Adaptation, 2014.

# [59] S. Choi, C. Kim, Y. Kim, J. Lee, and K. Cho, "Deep Learning for Audio-Based Music Classification and Tagging: Teaching Computers to Distinguish Rock from Bach," in 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017.

# [60] S. Dieleman and B. Schrauwen, "End-to-end learning for music audio," in 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014.

# [61] M.I. Mandel and D.P.W. Ellis, "Song-Level Features and SVMs for Music Classification," in 15th International Conference on Digital Audio Effects (DAFx-12), 2012.

# [62] Burred, J. J., "A Comprehensive Review of Music Genre Classification," IEEE Journal of Selected Topics in Signal Processing, vol. 9, no. 7, pp. 1038-1051, 2015.

# [63] Acker, B., and Porter, D., "On the Ambiguity of Genre Definitions in Music," IEEE Transactions on Multimedia, vol. 14, no. 2, pp. 450-457, 2012.

# [64] Yang, Y. H., and Chen, H. H., "Music Emotion Recognition: A Multimodal Approach," IEEE Transactions on Multimedia, vol. 16, no. 3, pp. 799-810, 2014.

# [65] Stevens, T., and Sandler, M., "An analysis of real-world music queries," ISMIR, pp. 462-467, 2009.

# [66] Whitman, B., and Ellis, D., "Automatic Record Reviews," ISMIR, pp. 470-477, 2004.

# [67] Davis, S. B., and Mermelstein, P., "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 28, no. 4, pp. 357-366, 1980.

# [68] Tzanetakis, G., and Cook, P., "Musical genre classification of audio signals," IEEE Transactions on Speech and Audio Processing, vol. 10, no. 5, pp. 293-302, 2002.

# [69] Doshi-Velez, F., and Kim, B., "Towards A Rigorous Science of Interpretable Machine Learning," arXiv preprint arXiv:1702.08608, 2017.

# [70] Flexer, A., "A closer look on artist filters for musical genre classification," ISMIR, pp. 341-346, 2007.

# [71] Davis, S. B., & Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE transactions on acoustics, speech, and signal processing, 28(4), 357-366.

# [72] Rabiner, L., & Juang, B. H. (1993). Fundamentals of speech recognition. PTR Prentice Hall.

# [73] Harris, F. J. (1978). On the use of windows for harmonic analysis with the discrete Fourier transform. Proceedings of the IEEE, 66(1), 51-83.

# [74] Oppenheim, A. V., & Schafer, R. W. (1999). Discrete-Time Signal Processing. Prentice Hall.

# [75] Stevens, S. S., Volkmann, J., & Newman, E. B. (1937). A scale for the measurement of the psychological magnitude pitch. Journal of the Acoustical Society of America, 8(3), 185-190.

# [76] Furui, S. (1981). Cepstral analysis technique for automatic speaker verification. IEEE Transactions on Acoustics, Speech, and Signal Processing, 29(2), 254-272.