SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
http://saucis.sakarya.edu.tr/
e-ISSN: 2636-8129
Publisher: Sakarya University
Vol. 7, No. 2, 2024, 289-301
DOI: 10.35377/saucis...1516717

RESEARCH ARTICLE

Evaluating Feature Selection Algorithms for Machine Learning-Based Musical Instrument Identification in Monophonic Recordings

İsmet Emre Yücel[1], Ulaş Yurtsever[2]

[1]Department of Music Technology, State Conservatory, Sakarya University, Sakarya, Türkiye
[2]Department of Computer Engineering, Sakarya University, Sakarya, Türkiye

Corresponding author:
İsmet Emre Yücel, Department of Music Technology, State Conservatory, Sakarya University, Sakarya, Türkiye
eyucel@sakarya.edu.tr

Article History:
Received: 15.07.2024
Accepted: 21.08.2024
Published Online: 27.08.2024

ABSTRACT
Musical instrument identification (MII) research has been studied as a subfield of the Music Information Retrieval (MIR) field. Conventional MII models are developed based on hierarchical models representing musical instrument families. However, for MII models to be used in the field of music production, they should be developed based on the arrangement-based functions of instruments in musical styles rather than these hierarchical models. This study investigates how the performance of machine learning based classification algorithms for Guitar, Bass guitar and Drum classes changes with different feature selection algorithms, considering a popular music production scenario. To determine the effect of feature statistics on model performance, Minimum Redundancy Maximum Relevance (mRMR), Chi-sqaure (Chi2), ReliefF, Analysis of Variance (ANOVA) and Kruskal Wallis feature selection algorithms were used. In the end, the neural network algorithm with wide hyperparameters (WNN) achieved the best classification accuracy (91.4%) when using the first 20 statistics suggested by the mRMR and ReliefF feature selection algorithms.

Keywords: Musical instrument identification, Audio features, Feature selection, Machine learning

1. Introduction

MII studies emerged as a combination of Audio Content Analysis (ACA) and machine learning methodologies. ACA is an essential part of the MIR research field and uses many approaches to attain useful audio signal information. Interpretations of the acquired information allow higher-level descriptions for the audio by machine learning algorithms. Besides its various uses, from speech recognition to medical signal analysis, ACA has a higher potential in music creation and production, using a combination of machine learning techniques. In that manner, the Intelligent Music Production (IMP) research field is a special implementation of ACA, MIR, and Digital Signal Processing (DSP) techniques for developing automatic or assistive solutions for any specific tasks in music production. This study aims to investigate the most effective audio features for musical instrument identification in the context of music production by supervised machine learning algorithms.

MII studies also known as musical instrument recognition studies back in the 90s. Instrument identification from audio is described as the biggest challenge in the MIR field [1], and even today, this challenge stays current due to the potential benefits of MII in streaming services [2]. The identification scenario depends on the source types whether they consist of single (monophonic) or multiple performances (polyphonic) in the same recording. Early works started with the investigation of single-source identification problems and their solutions [3], [4], [5], [6], [7], [8]. On the other hand, MII of polyphonic audio is a more difficult task than single-sourced audio. During the last decade, many studies have focused on identifying instruments from mixed audio sources [9]. Kitahara et al. pointed out three problems (feature variations in sound mixtures, pitch dependency, and musical context) of MII when polyphonic music [10].

Having diverse audio data is a requirement in polyphonic MII tasks. Commercial expectations and technological improvements from multimedia systems projected by the MPEG group and ISO/IEC 15938 became the standard in which content descriptions for audio and other multimedia formats take place [11]. The perceptual features in this standard were also used to create descriptions of instrumental sounds [17]. Peeters [18] listed the audio features categories as temporal shape, temporal feature, energy features, spectral shape features, harmonic features, and perceptual features. Audio features have been tested in different instrument recognition scenarios [19]. There are some data sets available. Humphrey et al. compared some available datasets for instrument identification [20]. Yücel and Özdemir [21] proposed an audio loop-based dataset regarding the music production perspective. To attain larger datasets, data augmentation may also be another alternative, which Kratimenos et al. [22] referred to for polyphonic MII tasks. Blaszke and Kostek [23] provided a comprehensive literature summary of ML methods and their results in MII.

Audio features are computational processes to attain temporal and timbral information from raw audio signals. Feature extraction refers to acquiring information by mathematical calculations in a short time window. The time window is typically between 2-10 msec durations, and computational results in that specific time duration are described as low-level features (LLF) or descriptors. Temporal, spectral, and statistical features are the three main audio feature categories [12]. Temporal features are calculated directly from the time-domain, spectral features require Fourier transform (STFT) of the signal and statistical features are derived from LLF in longer time frames. Temporal features also known as time-domain features are zero crossing rate (ZCR), root mean square (RMS), energy, envelope related calculations (attack, decay). Spectral features are also described as frequency domain features, and some of them are centroid, bandwidth, contrast, flatness, roll-off, and MFCC (Mel-frequency cepstrum coefficients) [13]. Detailed literature about audio features has been investigated in many studies[14], [15]. Mid-level and high-level descriptors are statistical features derived from LLF and extensively used to attain musical information, such as tempo, rhythm, genre, and mood [16].

Since extensive audio features are the basis for traditional ML applications in music, the dimensionality of feature vectors gets bigger. The bigger dimension causes an over-fitting problem for model predictions. This is mentioned as the curse of dimensionality in literature [17]. Audio features are fundamental components of ML-based solutions for musical applications. Performance of combined audio features in musical similarity studies [18], most effective feature combinations for musical classification [19], efficiency dependence of different audio features in music emotion recognition tasks [20], feature combinations for emotion (considering arousal and valence) recognition in music recordings [21] and feature selection in music genre classification tasks [22] are some examples for investigation of features effectiveness in different MIR sub fields. In musical instrument classification, Liu and Wan [23] investigated 58 features of data extracted from various audio recordings, achieving %93 of classification accuracy after 19 features. Gulhane et al. stated that MFCC 1, spectral roll-off and spectral centroid can be enough for instrument identification [24]. A higher accuracy rate was obtained with MFCC for K-NN classifiers and timbral descriptors (features) when the classification algorithm is Binary Tree [25]. Although some studies focus on deep learning methods directly from raw audio [26] [27], performance estimation of temporal and spectral audio features is worth investigating.

2. Methodology

2.1. Audio Dataset

Audio loops and single-hit samples are frequently used in musical materials during music creation in various mainstream genres such as Electronic/Dance, Pop, Rock etc. The loops are short musical sound recordings categorized by metadata such as tempo, musical tone, and source type. Musicians often use audio loops when developing and refining their musical ideas from scratch. It is therefore useful to use audio loops to create the audio datasets needed to develop AI-based models to solve various problems in music production scenarios. Modern DAWs (Digital audio workstations) provide music creators with many audio loops and sample libraries. The dataset for this study was created with repetitive and single beat audio files that come with Logic Pro, a popular DAW.

In this study, machine learning models are developed considering the monaural source classification approach. The recordings were organized into three instrument family groups as Drums, Basses, and Guitars considering the basic production elements in popular music, thus presenting class names for the dataset. All audio files were checked one by one to ensure that they were appropriate to the instrument family wherein each was placed. The Drums, Basses, and Guitars classes have 731, 406, 985 audio files respectively. The duration of these files varies from 1 second to several seconds.

2.2. Audio Preprocessing

Even though audio loops and samples generally consist of well-edited files, preprocessing is required to prepare them before the execution of audio feature extraction code. The preprocessing may contain one or more tasks including shorten longer files, DC removal, filtering (low-pass, high-pass, band-pass), pre-emphasis, sampling rate conversion, mono-to-stereo conversion, normalization. In this work, some audio files having longer sustaining parts are redundant, therefore they had been shortened where the points that their signal levels drop drastically. Another preprocessing implementation was stereo-to-mono conversion.

2.3. Audio Features and Extraction

For LLF extraction, in this study the preferred time window time and step duration are 4 msec and 2 msec, respectively. The calculated audio features are ZCR, RMS, spectral centroid, spectral bandwidth, spectral contrast, spectral flatness, spectral roll-off, and MFCC (20 bins) (Table 1). Due to diverse file durations, the mean, standard deviation (std), median, and variance (var) of the features were calculated for each audio. As a result, a 2122x108 dataset emerged. Since maximum and minimum value ranges differ for different features, data were normalized between 0-1.

Table 1. The Calculated Audio Features and Statistics.

Audio Features	**Statistics**
Zero crossing rate Root mean square Spectral centroid Spectral bandwidth Spectral contrast Spectral flatness Spectral roll-off MFCC (20 bins)	Mean Standard deviation (std) Median Variance (var)

Zero crossing rate (ZCR): This feature calculates how often a signal polarity changes from positive to negative or vice versa. In general, for a signal with simple harmonic content, such as a sine, ZCR increases as the frequency increases. On the other hand, audio signals with higher harmonic content or noise-like audio signals show higher ZCR values.

Root mean square (RMS): This feature allows the overall energy level of the audio signal to be observed by measuring the average intensity of the sound in each time window.

Spectral centroid: Spectral centroid is a measure that represents the center of mass of the frequency distribution of a sound signal. This feature helps to determine which frequencies the spectral components of a sound concentrate around. The spectral centroid provides information about the overall tonal characteristics of the sound. For example, if the spectral centroid of a music piece is close to higher frequencies, it indicates that the piece has brighter content.

Spectral bandwidth: Spectral bandwidth is a crucial feature in audio processing that describes the range of frequencies present in a sound signal. Specifically, it refers to the width of a frequency range within which most of the signal's energy is concentrated. A sound with a narrow bandwidth would have most of its energy concentrated within a small range of frequencies, like a pure tone. On the other hand, a sound with a wide bandwidth would have energy spread across a broader range of frequencies, like white noise.

Spectral Contrast: Spectral contrast computes the energy difference between adjacent frequency bands. The computation is basically a comparison of higher energy regions (peaks) to that of lower energy regions (valleys) within the spectrum. This feature is used in audio signal processing to characterize the difference in energy levels between different frequency bands within an audio signal's spectrum. It measures how pronounced the peaks and valleys are within the frequency spectrum.

Spectral Flatness: This audio feature (known as Wiener entropy) is a metric employed in audio signal analysis to quantify the relative distribution of energy across the frequency spectrum of a signal. It serves as an indicator of the signal's tonal versus noisy characteristics. Spectral flatness is calculated as the ratio between the geometric mean and the arithmetic mean of the power spectrum. A spectral flatness value closer to 1 indicates a relatively even energy distribution across the frequency spectrum, indicative of tonal or harmonic sounds. Conversely, values closer to 0 signify a more peaked distribution, suggesting a dominance of noise or non-harmonic components in the signal.

Spectral roll-off: This feature is defined as the frequency below which a specified percentage of the total spectral energy of a signal is contained. Typically, this percentage is set to 85%, but other values can be used depending on the application. The spectral roll-off point effectively separates the lower energy part of the spectrum from the higher energy part, providing insights into the signal's frequency distribution. It is particularly useful for distinguishing between harmonic and non-harmonic content in audio signals.

MFCC: Mel-Frequency Cepstral Coefficient (MFCC) is extensively utilized in speech recognition and musical classification. The frequency components derived from the STFT calculation are transformed using a non-linear Mel frequency scale and filtered into triangular frequency bands (bins), typically around 20. This process yields a concise representation of the spectral features of an audio signal, which closely resembles the human auditory system compared to raw spectral features.

2.4. ML Algorithms

This research was conducted using the Classification Learner App from Matlab version 2022a. Thus, this study was restricted to the ML algorithms provided by the software. Those algorithms are Decision Tree, Discriminant Analysis, Naïve Bayes, Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Ensemble and Neural Network with predefined kernel settings.

2.4.1. Decision Tree

The Decision Tree is a tree-like model where nodes represent tests on features (attributes), edges represent outcomes of these tests, and leaves represent the final decision or prediction. The root node is the top node representing the entire dataset, which is split into subsets based on the most significant feature. Splitting is the process of dividing a node into two or more sub-nodes to improve the homogeneity of the target variable within the sub-nodes. Decision nodes split into further sub-nodes, while leaf or terminal nodes do not split further and represent the final output. Each branch or sub-tree is a subsection of the entire tree. Decision trees offer several advantages, including interpretability, ease of visualization, and being non-parametric, making no assumptions about the data distribution. They also provide insights into feature importance. However, they have disadvantages, such as being prone to overfitting, especially with noisy data, and instability, where small changes in the data can result in different trees. Despite these drawbacks, decision trees are widely used in medical diagnosis, spam detection, customer segmentation, financial forecasting, pricing models, and risk assessment. They also serve as the foundation for more complex ensemble methods like Random Forests and Gradient Boosting Machines, which enhance predictive performance and mitigate some of the algorithm's limitations [28] [29].

2.4.2. Discriminant Analysis

Discriminant Analysis is a supervised machine learning technique primarily used for classification tasks. Its goal is to model the differences between classes by identifying the best combination of features that separates them. The algorithm prioritizes the probabilities of each class based on the training data. The algorithm then calculates these mean and covariance matrices according to two main types of approaches: Linear Discriminant Analysis (LDA), where these matrices are distributed across all classes, or Quadratic Discriminant Analysis (QDA), where these matrices are calculated separately for each class. Finally, for a given input, the algorithm calculates the value of the discriminant function for each class and identifies the class with the highest discriminant score [30].

2.4.3. Support Vector Machine

SVM is a prevalent supervised ML model. The model's reputation comes from its robustness and versatility, particularly its capability to discriminate high-dimensional spaces. The principle of the algorithm is to find the best boundary that separates different classes. This boundary tends to be a hyperplane where the dimensionality increases, especially in nonlinear conditions. The Support Vectors are the closest data points to hyperplanes influencing their position and orientation. The margin is a term that describes the distance between the hyperplane and the nearest support vectors from either class. SVM aims to maximize this margin to ensure the best separation between classes. This model uses different kernels to create higher dimensional space to separate classes. Common kernels include linear, polynomial, radial basis function (RBF), and sigmoid [31].

2.4.4. K-Nearest Neighbor

KNN is another popular supervised ML algorithm for classification and regression tasks. This algorithm discriminates data by measuring the distance between the instances. Common distance metrics include Euclidean distance, Manhattan distance, and Minkowski distance. The “k” symbolizes the closest training examples (neighbors) in the feature space for given data. The smaller k may indicate higher data noise and overfitting, yet the larger one causes underfitting. The algorithm consists of four main steps; first, it holds the instances from training data, then calculates distances between training and testing instances and finds the parameter “k” for each instance [32]. This algorithm has been applied in many tasks in medical applications, such as tuberculosis classification using x-ray images [33], heart disease determination [34], engineering; localization indoors [35], agricultural studies [36], music and speech classification [37], music genre classification [38].

2.4.5. Ensemble

Ensemble algorithms are conceptualized by combining multiple ML models to improve overall performance. There are several types of ensemble methods; the most common are bagging (Random Forest), boosting (AdaBoosting, Gradient Boosting), and stacking (Base Learners, Meta-Learner) [39]. The bagging technique combines statistical methods called bootstrapping, where random sampling (equal-sized subsets) from the training data is taken, and the aggregation method allows the regeneration of the training result of the whole model with the results obtained from the training data. Random forest is a featured bagging type algorithm based on decision trees [40]. The AdaBoost algorithm was proposed by Freund and Schapire to enhance previous boasting algorithms [41]. This algorithm transforms weak learners into strong learners, even without prior accurate information on the weak hypotheses. This method improved the implementation of various multiclass classification and regression problems [42]. The Gradient Boosting method proposed by Friedman function estimation/approximation is a numerical optimization problem in function space. It generalizes boosting by using gradient descent techniques to optimize any fitting criterion, not just for binary classification but also for regression and multiclass classification problems. Specific algorithms for least-squares, least absolute deviation, Huber-M loss functions for regression, and multiclass logistic likelihood for classification are presented, with a special focus on using regression trees as the base learners. This approach enhances robustness, interpretability, and performance, especially for handling less-than-clean data [43].

2.4.6. Neural Network

Neural networks are a category of machine learning architectures that draw inspiration from the configuration and operation of the human brain. Neural networks are composed of interconnected layers of neurons that analyze input data to generate an output. These networks provide a great degree of adaptability and can represent intricate connections within data, rendering them appropriate for a diverse array of tasks including classification, regression, and pattern recognition. Neurons, the fundamental components of a neural network, receive input, do a weighted sum, and then transmit the outcome through an activation function. Neural networks consist of layers, with the input layer receiving the input data, hidden layers processing the inputs from the previous layer, and the output layer generating the final prediction. Activation functions such as sigmoid, tanh, and ReLU (Rectified Linear Unit) provide non-linear behavior in the network, allowing it to acquire knowledge of intricate patterns. Training refers to improving the weights and biases of a network using methods such as backpropagation and gradient descent [44].Different hyperparameters may be preferred during the building of neural network-based ML models. The version of Matlab that we used in this study provides narrow, medium, wide, bi-layered, and tri-layered hyperparameters.

A narrow neural network has fewer neurons in its layers, making it simpler and faster to train with lower computational requirements. However, narrow networks may not capture complex patterns as effectively as wider networks and are suitable for simpler tasks or when computational resources are limited [45]. A medium neural network strikes a balance between complexity and computational efficiency, with a moderate number of neurons in its layers. This configuration provides a good trade-off between capturing patterns and avoiding overfitting, making it commonly used for a wide range of tasks, including classification and regression problems [46]. A bi-layered neural network features two hidden layers, allowing it to model more complex patterns than single-layer networks. Although the complexity of training increases, it remains manageable. Bi-layered networks are suitable for tasks where a single hidden layer is insufficient but additional layers do not significantly improve performance. In contrast, a tri-layered neural network has three hidden layers and can learn even more complex patterns than bi-layered networks. While this configuration incurs higher computational costs and a greater risk of overfitting, it is used in more complex tasks where deeper representations are needed, such as deep learning applications in image and speech recognition [47].

The wide neural network (thereafter referred to as WNN) is characterized by having many neurons in each layer. This type of network can capture more complex patterns in the data due to its higher capacity but requires more computational resources and is prone to overfitting if not regularized properly. WNNs are used in scenarios where the dataset is large and complex, such as image recognition and natural language processing [48]. The principle of WNN has the number of neurons in each layer, as opposed to the depth, which refers to the number of layers in the network. In the context of wide neural networks, the emphasis is on increasing the number of neurons per layer rather than stacking many layers allowing for more complex representations of the input data. WNN uses the Tangent type kernel method for the model creation. Advantages of the model are dealing with more complex data structures providing better representation of the input data, consistency in classification stabile training phase, connections to kernel methods and effective interpolation of training data [49].

2.5. Feature Selection Algorithms

Generally, most datasets consist of high dimensional features that do not have the same importance to create distinction about data. Therefore, feature selection is an approach to determining the most related features to represent data. Guyon and Elisseeff [27] stated the potential benefits of feature selection. In this study, five feature selection algorithms, mRMR (Maximum Relevance and Minimum Redundancy), Chi2, ReliefF, ANOVA, and Kruskal Wallis, were examined.

2.5.1. mRMR

The mRMR is considered a robust filtering method and was initially developed for the classification of DNA microarray data. Some implementations of the method are anomaly detection, eye movement analysis, gender classification, and analysis of satellite images [28]. This method aims to determine a subset of features that are highly important for a specific job while minimizing redundancy among them. Relevance measures the importance of each feature with respect to the target variable, often using mutual information. Redundancy measures the similarity among the features, again often using mutual information. Features are selected iteratively by adding the one that maximizes the difference between relevance and redundancy. Various research has emphasized the benefits of the mRMR algorithm. Tang et al. highlighted that mRMR is recognized for its rapid computation speed and resilience, as it automatically identifies essential characteristics by considering maximum correlation and lowest redundancy criteria [50]. In addition, Alshamlan et al. examined the efficacy of integrating mRMR with genetic algorithms and particle swarm optimization algorithms for gene selection in cancer classification [51]. Their study showcased the adaptability of mRMR in improving classification tasks.

2.5.2. Chi-Square

The Chi2 (Chi-Square) is a two phased x2 statistical method where data discretization occurs [52]. This method examines if there is a significant connection between category variables and the target variable. It evaluates whether the observed frequency distribution of a feature differs from the expected distribution. To perform feature selection using the Chi-Square statistic, start by calculating the Chi-Square statistic for each feature, which measures the discrepancy between the observed and expected frequencies. This involves creating a contingency table for each feature with respect to the target variable and computing the Chi-Square value for each table. Next, rank the features based on their Chi-Square statistics in descending order, highlighting those with the most significant discrepancies. Finally, select the top-ranked features based on a predefined threshold or the desired number of features. This selection process ensures that the most informative features are retained for further analysis or model building. Chi2 feature selection algorithm has been widely used in various domains such as computer security, sentiment analysis, malware detection, and health management. Studies have shown that the Chi2 algorithm effectively evaluates the dependency level of feature sets on class labels [53]. It has been used with other techniques like TFIDF for sentiment analysis [54], and Multinomial Naïve Bayes for question classification [55]. Additionally, the algorithm has been applied in feature selection for audio-related tasks, such as emotion regulation music recommendation Gong et al. [56] and malware detection systems [57].

2.5.3. ReliefF

ReliefF was originally developed by Kira and Rendell [58]. This algorithm can handle multi-class problems and is robust to noisy and incomplete data. It estimates the quality of features by considering the difference between feature values of nearest neighbors from the same and different classes. This feature selection algorithm basically searches two nearest hit and miss neighbors in an instance. The algorithm starts the selection process by setting all the feature weights to zeros. Afterward, it searches for nearest neighbors from the same class (hits) and different classes (misses) of instances that random samples from the dataset. The algorithm updates the feature weights in each process based on the difference between the sampled instance and its neighbors. Consequently, the features are selected according to their weights. The ReliefF algorithm, including its variations like I-ReliefF and SURF, has shown efficiency in estimating feature quality, especially in scenarios with strong dependencies among features [59], [60], [61]. It has been successfully applied in diverse fields such as biology, medicine, and computer science, showcasing its versatility and effectiveness [62], [63], [64]. ReliefF has also been combined with other techniques like Random Forest and Support Vector Machines to enhance classification models and improve feature subset selection [62].

2.5.4. ANOVA

The ANOVA (Analysis of Variance) is useful to check a hypothesis state whether it is null or not. The process begins by calculating the F-statistic for each feature, which measures the variance ratio between groups to the variance within groups. This helps identify features that contribute significantly to the variance in the target variable. Next, rank the features based on their F-statistic, prioritizing those with higher values. Finally, select the top-ranked features based on a predefined threshold or the desired number of features, ensuring that the most influential features are included in the model. ANOVA is utilized to reduce computations and time complexity while enhancing accuracy by overcoming the curse of dimensionality [65]. This method is widely used in many fields such as agriculture, biology [66], [67], medical research [68], [69], and engineering [70], [71].

2.5.5. Kruskal Wallis

The Kruskal Wallis is a nonparametric test that determines if all k populations are identical or if at least one of the populations tends to give observations different from those of other populations. The Kruskal Wallis test is a non-parametric alternative to ANOVA, designed to compare the medians of multiple groups without assuming data normality, making it suitable for ordinal or non-normal continuous data. The process begins by ranking all data points across groups. Next, calculate the H-statistic, which measures the discrepancy between the ranks of the groups. After calculating the H-statistic for each feature, rank the features based on their H-statistic values. Finally, select the top-ranked features based on a predefined threshold or the desired number of features. The Kruskal Wallis test offers several advantages, including not assuming data normality, suitability for ordinal or non-normal continuous data, and robustness to outliers. However, it is less powerful than ANOVA when the assumptions of normality and homogeneity of variances are met and are sensitive to ties in the data [72].

3. Results and Discussion

Model training, using a 5-fold cross-validation method, was preferred, and then the dataset was trained by 28 supervised ML algorithms with their default settings. Consequently, the WNN algorithm provided the best classification result at 93.2% before feature selection had been applied. The detailed ML model results are given in Table 2.

Table 2. The Tested Supervised ML Algorithms

Model	Hyperparameter	Accuracy
Tree	Fine	86.1%
	Medium	83.3%
	Coarse	76.0%
Discriminant	Linear	85.2%
Naïve Bayes	Gaussian	74.5%
	Kernel	65.6%
SVM	Linear	87.3%
	Quadratic	91.9%
	Cubic	92.6%
	Fine Gaussian	89.2%
	Medium Gaussian	87.4%
	Coarse Gaussian	80.9%

KNN	Fine	91.3%
	Medium	87.3%
	Coarse	79.9%
	Cosine	88.0%
	Cubic	87.1%
	Weighted	90.3%
Ensemble	Boosted	89.1%
	Bagged Trees	91.5%
	Subspace Discriminant	84.1%
	Subspace KNN	92.6%
	RUSBoosted Trees	87.8%
Neural Network	Narrow	91.7%
	Medium	92.2%
	Wide	93.2%
	Bi-layered	90.9%
	Tri-layered	90.7%

The determined top 20 audio features by mRMR, Chi2, ReliefF, ANOVA, and Kruskal Wallis algorithms are given in Table 3. The roll-off median has a higher grade in the list. It is also the primary feature according to mRMR and Chi2 algorithms in this experiment. For simplicity, the word "Spectral" is omitted in Tables 3 and 4, and the band index (bin) of the MFCC attribute is indicated by the number next to it.

Table 3. The Top 20 Audio Features Ranged by the Feature Selection Algorithms

	mRMR	**Chi2**	**ReliefF**	**ANOVA**	**Kruskal Wallis**
1	Roll-off median	Roll-off median	Bandwidth std	Roll-off mean	Flatness median
2	Contrast var	ZCR median	Roll-off median	Roll-off median	Flatness mean
3	MFCC 11	ZCR mean	Roll-off mean	Centroid mean	Roll-off mean
4	Bandwidth var	Centroid median	Centroid mean	Centroid median	Roll-off median
5	Flatness median	Flatness median	Centroid median	ZCR mean	Centroid median
6	MFCC 14 mean	Roll-off mean	Bandwidth var	ZCR median	Centroid mean
7	Contrast mean	Centroid mean	Bandwidth median	Flatness mean	MFCC 7 mean
8	ZCR median	Flatness mean	ZCR mean	Bandwidth mean	MFCC 6 mean
9	Flatness mean	MFCC 6 mean	Roll-off std	ZCR std	MFCC 8 mean
10	MFCC 7 mean	MFCC 7 mean	ZCR median	Flatness std	Flatness var
11	ZCR var	MFCC 8 mean	Bandwidth mean	Bandwidth median	Flatness std
12	Roll-off mean	ZCR std	Flatness std	MFCC 4 std	ZCR mean
13	MFCC 4 var	ZCR var	Contrast mean	Flatness var	MFCC 9 mean
14	Flatness var	MFCC 6 std	Contrast median	MFCC 7 mean	ZCR median
15	MFCC 9 mean	MFCC 6 var	Roll-off var	MFCC 5 std	MFCC 5 mean
16	Centroid mean	MFCC 2 std	ZCR std	MFCC 8 mean	MFCC 10 mean
17	Bandwidth median	MFCC 2 var	Flatness mean	MFCC 6 mean	ZCR var
18	MFCC 18 mean	MFCC 5 std	RMS std	MFCC 3 std	ZCR std
19	MFCC 6 mean	MFCC 5 var	Flatness var	MFCC 6 std	MFCC 7 median
20	ZCR mean	MFCC 4 std	RMS mean	MFCC 9 mean	MFCC 8 median

With the dataset and the WNN classification algorithm, it was observed that the common audio features listed by the feature selection algorithms are roll-off mean, roll-off median, ZCR mean, ZCR median, centroid mean, and flatness mean (in Table 4). MFCC 7 mean, flatness var, MFCC 6 mean, centroid median, and ZCR std are mutual features listed by four selection algorithms. Flatness median, ZCR var, MFCC 9 mean, bandwidth median, and MFCC 8 mean are common in three selection algorithm lists. bandwidth var, contrast mean, MFCC 6 std, MFCC 5 std, MFCC 4 std, and flatness std take place only in two selection lists. Eventually, 23 features are listed by only a single selection algorithm.

Table 4. The Top 20 Common Features and Uncommon Features Listed by the Selection Algorithm(s)

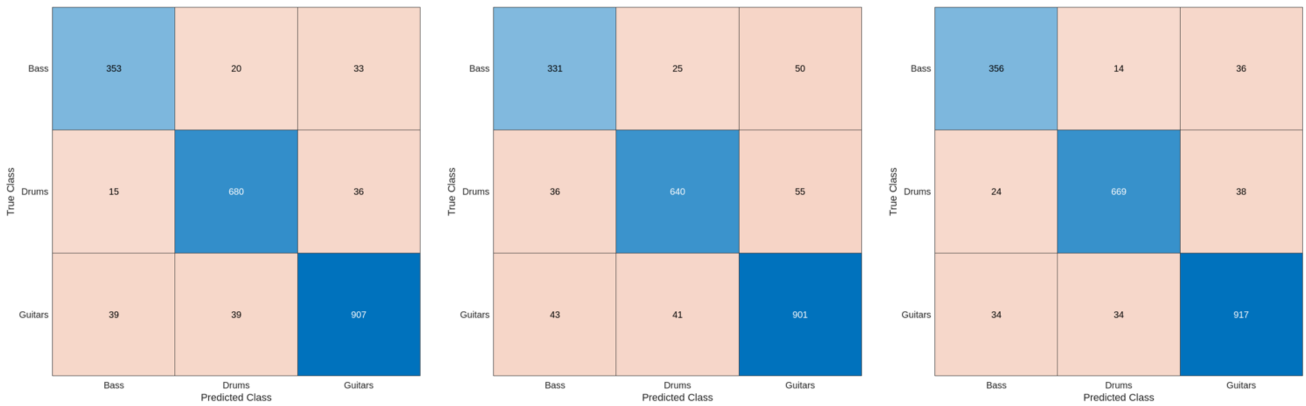
Common Selected Features by the	**Audio Features Statistics**	**Number of features**
All algorithms	Roll-off mean, roll-off median, ZCR mean, ZCR median, centroid mean, flatness mean	6
4 algorithms	MFCC 7 mean, flatness var, MFCC 6 mean, centroid median, ZCR std	5
3 algorithms	Flatness median, ZCR var, MFCC 9 mean, bandwidth median, MFCC 8 mean	5
2 algorithms	Bandwidth var, contrast mean, MFCC 6 std, MFCC 5 std, MFCC 4 std, flatness std	6
Listed only by 1 selection algorithm	Contrast var, MFCC 11, MFCC 14 mean, MFCC 4 var, MFCC 18 mean, MFCC 6 var, MFCC 2 std, MFCC 2 var, MFCC 5 var, bandwidth std, roll-off std, bandwidth mean, flatness std, contrast median, roll-off var, RMS std, RMS mean, MFCC 3 std, MFCC 10 mean, ZCR var, MFCC 5 mean, MFCC 7 median, MFCC 8 median	23

Further experiments have been performed to check ML model performances with the selected features. The best performance rate of WNN was 93.2% with 108 audio features, and this score degrades according to different feature lists by the selection algorithms. In Table 5, performance changes of the WNN model are given for each algorithm's top 20 list. According to the table, mRMR and ReliefF share the same performance rates at 91.4%, even though their feature list contents are different. The performance results are 89.3% for ANOVA, 88.4%, for Chi2 and 88.3% for Kruskal Wallis (Table 5). Also, the confusion matrix of the WNN model is given for each algorithm's top 20 lists in Figure 1. Figure 2 shows the ROC curve of the WNN model for each algorithm's top 20 lists. Additionally, it was observed that the performance result dropped to 84.5% rate, after the ML model was trained by common features (roll-off mean, roll-off median, ZCR mean, ZCR median, centroid mean, flatness mean).

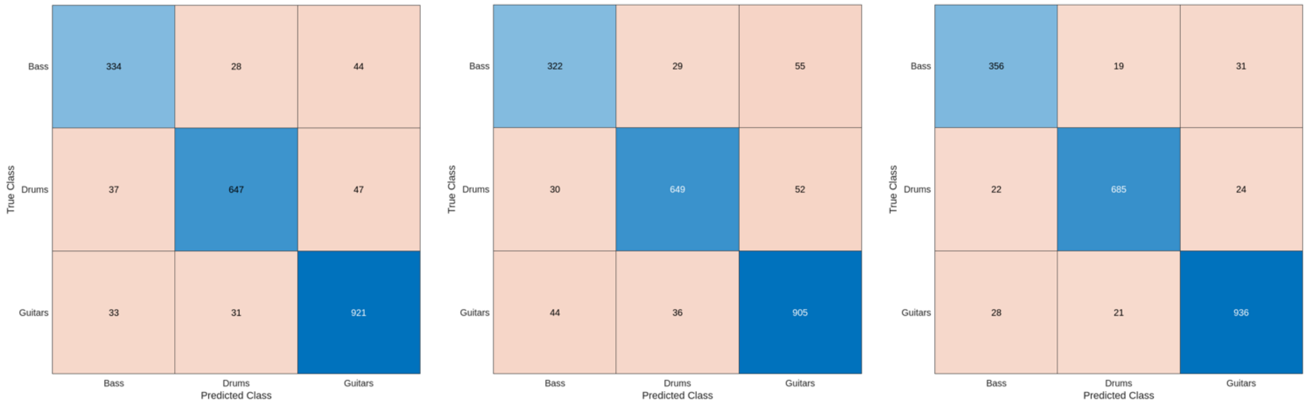
Table 5. Classification Success Rates of WNN Algorithm with Selected the First 20 Features

mRMR's top 20	**Chi2's top 20**	**ReliefF's top 20**	**ANOVA's top 20**	**Kruskal Wallis's top 20**
91.4%	88.4%	91.4%	89.3%	88.3%

WNN algorithm models were created using the first 20 features selected with different feature selection methods. The confusion matrices created to evaluate the performance of these models using different feature selection methods are shown in Figure 1. When the confusion matrices are analyzed, it is observed that high accuracy rates are obtained in general, but the error rates on classes vary depending on the methods used.



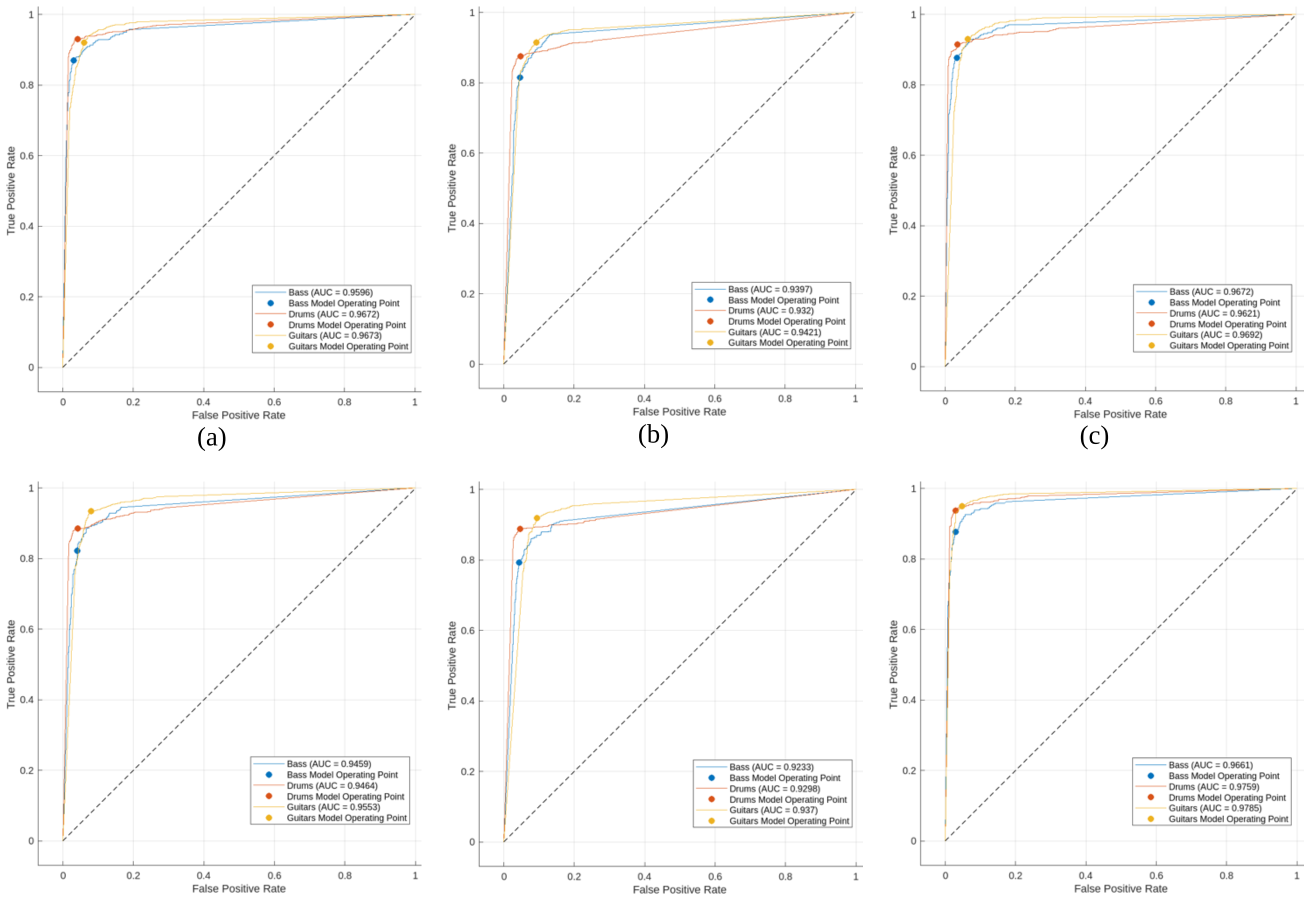
(a) (b) (c)



(d) (e) (f)

Figure 1. Confusion Matrixes of WNN Algorithm with Selected the First 20 Features a) mRMR b) Chi2 c) ReliefF d) ANOVA e) Kruskal Wallis f) No Selection

Figure 2 shows the ROC curves of the WNN algorithm models created with different feature selection methods. Each ROC curve shows the classification performance of the model in the 'Bass', 'Drums' and 'Guitars' classes separately. In particular, high AUC values were obtained in the 'Guitars' class, indicating that this class can be better classified than the others. In general, high AUC values were obtained with all feature selection methods, indicating that the classification performance of the models is high. When different feature selection methods are analyzed, similar to the confusion matrix, it is observed that the AUC values of the models created with mRMR and ReliefF methods give similar results when compared with the model performance using all features.



(a) (b) (c)

(d) (e) (f)

Figure 2. ROC Curves of WNN Algorithm with Selected the First 20 Features a) mRMR b) Chi2 c) ReliefF d) ANOVA e) Kruskal Wallis f) No Selection

It is worth mentioning that MFCC should be interpreted as a whole feature group even if the algorithms ranked its bins differently. Considering the results there is no direct relationship between selected features even though some of them are available in more than one selection algorithm list. In this experiment, when the model performance was analyzed after the feature selection process, there was no significant feature statistic that could be used to achieve good classification results.

At that point, further investigations are required to understand how different statistics affect the classification results. Timbre is the fundamental parameter of how human beings discriminate sound. Thus, the effort of estimating a sound source strongly depends on the timbral representation of the source, yet it is hard to model only a couple of audio features. Therefore, strict feature selection may not be suitable for audio classification especially in MII tasks. A specific MII approach exemplified in this study (single-hit and monophonic resources) may not be suitable for MII models designed with longer and poly-sourced audio files because those types of models will diversify the requirements of audio preprocessing, feature sets, ML algorithm and feature selection. Another point is that the description of a sound source from the perspective of audio production and music creation varies broadly. The functionality of an instrument in a musical arrangement cannot be explained only by its organologic root but also by the intention of the musician while creating the music.

4. Conclusion

In this study, various machine learning algorithms and feature selection methods were evaluated for their effectiveness in musical instrument identification (MII) using monophonic and single-hit sound sources. The experiment was constrained to three instrument families—Drums, Basses, and Guitars—within the context of popular music production scenarios. The dataset consisted of 27 audio features (with MFCCs having 20 bins) and their four statistical variations. Following the training of the dataset with seven machine learning algorithms and their various hyperparameters, the Wide Neural Network (WNN) demonstrated the highest classification accuracy at 93.2%. Upon further examination with five feature selection algorithms (mRMR, Chi2, ReliefF, ANOVA, and Kruskal-Wallis), it was found that common features selected by these algorithms, such as spectral roll-off mean and median, zero-crossing rate mean and median, spectral flatness mean, and spectral centroid mean, played a crucial role in classification performance. Both mRMR and ReliefF achieved a performance rate of 91.4% with the WNN, highlighting the importance of these common features.

However, this study also highlights several areas for future research and acknowledges its limitations. The current study focused on a controlled experimental setup with specific instruments and recording conditions, which may not fully represent the diversity of real-world music production scenarios. Future research should explore how these algorithms perform in different contexts, such as polyphonic recordings or real-time music production environments, to extend the applicability of the findings. Additionally, the limitations of traditional feature selection methods suggest that there is room for exploring more sophisticated techniques or integrating feature selection directly into deep learning models. Expanding the dataset to include a broader range of instruments, styles, and recording conditions, potentially through data augmentation, could enhance the generalizability of the models developed. Moreover, investigating the usability and performance of these models in actual music production software would provide valuable insights into their real-world applications. Finally, future studies could benefit from a cross-disciplinary approach, involving collaboration between experts in music production, machine learning, and digital signal processing, to develop more sophisticated models that can better understand and categorize musical instruments.

By addressing these areas, future research can build upon the findings of this study, leading to more robust and generalizable models for musical instrument identification, ultimately contributing to advancements in both music technology and artificial intelligence.

References

[1] A. Ghosh, A. Pal, D. Sil, and S. Palit, "Music Instrument Identification Based on a 2-D Representation," in *3rd International Conference on Electrical, Electronics, Communication, Computer Technologies and Optimization Techniques, ICEECCOT 2018*, Institute of Electrical and Electronics Engineers Inc., Dec. 2018, pp. 509–513. doi: 10.1109/ICEECCOT43722.2018.9001486.

[2] U. Shukla, U. Tiwari, V. Chawla, and S. Tiwari, "Instrument classification using image based transfer learning," in *Proceedings of the 2020 International Conference on Computing, Communication and Security, ICCCS 2020*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020. doi: 10.1109/ICCCS49678.2020.9277366.

[3] I. Kaminskyj and A. Materka, "AUTOMATIC SOURCE IDENTIFICATION OF MONOPHONIC MUSICAL INSTRUMENT SOUNDS," *Proceedings of the Australian and New Zealand Conference on Intelligent Information Systems*, 1995.

[4] I. Kaminskyj and P. Voumard, "Enhanced automatic source identification of monophonic musical instrument sounds," *Proceedings of the Australian and New Zealand Conference on Intelligent Information Systems*, no. November, pp. 76–79, 1996.

[5] K. D. Martin and Y. E. Kim, "2pMU9. Musical instrument identification: A pattern-recognition approach *," in *Presented at the 136th meeting of the Acoustical Society of America*, Newyork, 1998.

[6] P. Herrera-Boyer, G. Peeters, and S. Dubnov, "Automatic classification of musical instrument sounds," in *International Journal of Phytoremediation*, Journal of New Music Research, 2003, pp. 3–21.

[7] S. K. Banchhor and A. Khan, "Musical Instrument Recognition using Spectrogram and Autocorrelation," *Soft comput*, no. 1, pp. 1–4, 2012.

[8] H. Mukherjee, S. M. Obaidullah, S. Phadikar, and K. Roy, "SMIL - A Musical Instrument Identification System," Springer, Singapore, 2017, pp. 129–140. doi: 10.1007/978-981-10-6427-2_11.

[9] Y. Han, J. Kim, and K. Lee, "Deep Convolutional Neural Networks for Predominant Instrument Recognition in Polyphonic Music," *IEEE/ACM Trans Audio Speech Lang Process*, vol. 25, no. 1, pp. 208–221, Jan. 2017, doi: 10.1109/TASLP.2016.2632307.
[10] T. Kitahara, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno, "Instrument identification in polyphonic music: Feature weighting with mixed sounds, pitch-dependent timbre modeling, and use of musical context," *ISMIR 2005 - 6th International Conference on Music Information Retrieval*, no. January, pp. 558–563, 2005.
[11] S. Chang, S. Member, T. Sikora, S. Member, and A. Puri, "Overview of the MPEG-7 Standard," vol. 11, no. 6, pp. 688–695, 2001.
[12] M. R. Bai, A. Member, and C. Chen, "Intelligent Preprocessing and Classification of Audio Signals*."
[13] P. Wei, F. He, L. Li, and J. Li, "Research on sound classification based on SVM," *Neural Comput Appl*, vol. 32, no. 6, pp. 1593–1607, Mar. 2020, doi: 10.1007/s00521-019-04182-0.
[14] F. Alías, J. C. Socoró, and X. Sevillano, "A review of physical and perceptual feature extraction techniques for speech, music and environmental sounds," *Applied Sciences*, vol. 6, no. 5. Balkan Society of Geometers, 2016. doi: 10.3390/app6050143.
[15] J. D. Deng, C. Simmermacher, and S. Cranefield, "A study on feature analysis for musical instrument classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 38, no. 2, pp. 429–438, 2008, doi: 10.1109/TSMCB.2007.913394.
[16] A. Aljanaki and M. Soleymani, "A data-driven approach to mid-level perceptual musical feature modeling," Jun. 2018, [Online]. Available: http://arxiv.org/abs/1806.04903
[17] J. L. Fernández-Martínez and Z. Fernández-Muñiz, "The curse of dimensionality in inverse problems," *J Comput Appl Math*, vol. 369, 2020, doi: 10.1016/j.cam.2019.112571.
[18] J. Osmalskyj, M. Van Droogenbroeck, and J. J. Embrechts, "Performances of low-level audio classifiers for large-scale music similarity," in *International Conference on Systems, Signals, and Image Processing*, 2014, pp. 91–94.
[19] Z. Fu, G. Lu, K. M. Ting, and D. Zhang, "On feature combination for music classification," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2010, pp. 453–462. doi: 10.1007/978-3-642-14980-1_44.
[20] M. Chmulik, R. Jarina, M. Kuba, and E. Lieskovska, "Continuous music emotion recognition using selected audio features," in *2019 42nd International Conference on Telecommunications and Signal Processing, TSP 2019*, 2019. doi: 10.1109/TSP.2019.8768806.
[21] J. Grekow, "Audio features dedicated to the detection of arousal and valence in music recordings," in *Proceedings - 2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications, INISTA 2017*, 2017, pp. 40–44. doi: 10.1109/INISTA.2017.8001129.
[22] J. Mitra and D. Saha, "An Efficient Feature Selection in Classification of Audio Files," pp. 29–38, 2014, doi: 10.5121/csit.2014.4303.
[23] M. Liu and C. Wan, "Feature selection for automatic classification of musical instrument sounds," *Proceedings of the ACM International Conference on Digital Libraries*, pp. 247–248, 2001, doi: 10.1145/379437.379663.
[24] S. R. Gulhane, S. S. Badhe, and S. D. Shirbahadurkar, "Cepstral (MFCC) Feature and Spectral (Timbral) Features Analysis for Musical Instrument Sounds," *Proceedings - 2018 IEEE Global Conference on Wireless Computing and Networking, GCWCN 2018*, pp. 109–113, 2018, doi: 10.1109/GCWCN.2018.8668628.
[25] P. S. Jadhav, "Classification of Musical Instruments sounds by Using MFCC and Timbral Audio Audio Descriptors," 2015.
[26] J. Lee, T. Kim, J. Park, and J. Nam, "Raw Waveform-based Audio Classification Using Sample-level CNN Architectures," no. Nips, 2017.
[27] K. Avramidis, A. Kratimenos, C. Garoufis, A. Zlatintsi, and P. Maragos, "Deep convolutional and recurrent networks for polyphonic instrument classification from monophonic raw audio waveforms," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2021-June, pp. 3010–3014, 2021, doi: 10.1109/ICASSP39728.2021.9413479.
[28] T. M. Hehn, J. F. P. Kooij, and F. A. Hamprecht, "End-to-End Learning of Decision Trees and Forests," *Int J Comput Vis*, vol. 128, no. 4, 2020, doi: 10.1007/s11263-019-01237-6.
[29] Z. ÇETİNKAYA and F. HORASAN, "Decision Trees in Large Data Sets," *Uluslararası Muhendislik Arastirma ve Gelistirme Dergisi*, vol. 13, no. 1, 2021, doi: 10.29137/umagd.763490.
[30] A. Araveeporn, "Comparison of Logistic Regression and Discriminant Analysis for Classification of Multicollinearity Data," *WSEAS Trans Math*, vol. 22, 2023, doi: 10.37394/23206.2023.22.15.
[31] A. Saini, "Guide on Support Vector Machine (SVM) Algorithm," *Analytics Vidhya*, 2024.
[32] S. Uddin, I. Haque, H. Lu, M. A. Moni, and E. Gide, "Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction," *Sci Rep*, vol. 12, no. 1, Dec. 2022, doi: 10.1038/S41598-022-10358-X.
[33] R. A. Rizal, N. O. Purba, L. A. Siregar, K. P. Sinaga, and N. Azizah, "Analysis of Tuberculosis (TB) on X-ray Image Using SURF Feature Extraction and the K-Nearest Neighbor (KNN) Classification Method," *Jaict*, vol. 5, no. 2, p. 9, Oct. 2020, doi: 10.32497/JAICT.V5I2.1979.
[34] B. Akalin, Ü. Veranyurt, and O. Veranyurt, "Classification of Individuals at Risk of Heart Disease Using Machine

Learning," *Cumhuriyet Medical Journal*, 2020, doi: 10.7197/cmj.vi.742161.
[35] X. Peng, R. Chen, K. Yu, F. Ye, and W. Xue, "An improved weighted k-nearest neighbor algorithm for indoor localization," *Electronics (Switzerland)*, vol. 9, no. 12, 2020, doi: 10.3390/electronics9122117.
[36] H. K. Karthikeya, K. Sudarshan, and D. S. Shetty, "Prediction of Agricultural Crops using KNN Algorithm," *Int J Innov Sci Res Technol*, vol. 5, no. 5, 2020.
[37] R. Thiruvengatanadhan, "Speech/Music Classification using MFCC and KNN," 2017.
[38] X. Mu, "Implementation of Music Genre Classifier Using KNN Algorithm," *Highlights in Science, Engineering and Technology*, vol. 34, 2023, doi: 10.54097/hset.v34i.5439.
[39] I. D. Mienye and Y. Sun, "A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects," *IEEE Access*, vol. 10. 2022. doi: 10.1109/ACCESS.2022.3207287.
[40] A. Verikas, A. Gelzinis, and M. Bacauskiene, "Mining data with random forests: A survey and results of new tests," *Pattern Recognit*, vol. 44, no. 2, 2011, doi: 10.1016/j.patcog.2010.08.011.
[41] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *J Comput Syst Sci*, vol. 55, no. 1, 1997, doi: 10.1006/jcss.1997.1504.
[42] R. E. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," *Mach Learn*, vol. 37, no. 3, 1999, doi: 10.1023/A:1007614523901.
[43] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann Stat*, vol. 29, no. 5, 2001, doi: 10.1214/aos/1013203451.
[44] S. Joshi, A. Gera, and S. Bhadra, "Neural Networks and Their Applications," in *Evolving Networking Technologies: Developments and Future Directions*, 2023. doi: 10.1002/9781119836667.ch13.
[45] G. Alfonso and D. R. Ramirez, "Neural networks in narrow stock markets," *Symmetry (Basel)*, vol. 12, no. 8, 2020, doi: 10.3390/SYM12081272.
[46] M. Saglam, C. Spataru, and O. A. Karaman, "Forecasting Electricity Demand in Turkey Using Optimization and Machine Learning Algorithms," *Energies (Basel)*, vol. 16, no. 11, 2023, doi: 10.3390/en16114499.
[47] J. Chen, K. Li, K. Bilal, X. Zhou, K. Li, and P. S. Yu, "A Bi-layered parallel training architecture for large-scale convolutional neural networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 5, 2019, doi: 10.1109/TPDS.2018.2877359.
[48] J. Xi, O. K. Ersoy, J. Fang, T. Wu, X. Wei, and C. Zhao, "Parallel Multistage Wide Neural Network," *IEEE Trans Neural Netw Learn Syst*, vol. 34, no. 8, 2023, doi: 10.1109/TNNLS.2021.3120331.
[49] A. Radhakrishnan, M. Belkin, and C. Uhler, "Wide and deep neural networks achieve consistency for classification," *Proc Natl Acad Sci U S A*, vol. 120, no. 14, 2023, doi: 10.1073/pnas.2208779120.
[50] X. Tang, Q. He, X. Gu, C. Li, H. Zhang, and J. Lu, "A Novel Bearing Fault Diagnosis Method Based on GL-mRMR-SVM," *Processes*, vol. 8, no. 7, Jul. 2020, doi: 10.3390/PR8070784.
[51] H. Alshamlan, G. Badr, and Y. Alohali, "mRMR-ABC: A Hybrid Gene Selection Algorithm for Cancer Classification Using Microarray Gene Expression Profiling," *Biomed Res Int*, vol. 2015, 2015, doi: 10.1155/2015/604910.
[52] H. Liu and R. Setiono, "Chi2: feature selection and discretization of numeric attributes," *Proceedings of the International Conference on Tools with Artificial Intelligence*, pp. 388–391, 1995, doi: 10.1109/tai.1995.479783.
[53] T. D. Diwan *et al.*, "Feature Entropy Estimation (FEE) for Malicious IoT Traffic and Detection Using Machine Learning," *Mobile Information Systems*, vol. 2021, 2021, doi: 10.1155/2021/8091363.
[54] U. I. Larasati, M. A. Muslim, R. Arifudin, and A. Alamsyah, "Improve the Accuracy of Support Vector Machine Using Chi Square Statistic and Term Frequency Inverse Document Frequency on Movie Review Sentiment Analysis," *Scientific Journal of Informatics*, vol. 6, no. 1, pp. 138–149, May 2019, doi: 10.15294/SJI.V6I1.14244.
[55] N. Yusliani, S. A. Q. Aruda, M. D. Marieska, D. M. Saputra, and A. Abdiansah, "The effect of Chi-Square Feature Selection on Question Classification using Multinomial Naïve Bayes," *Sinkron*, vol. 7, no. 4, pp. 2430–2436, Oct. 2022, doi: 10.33395/SINKRON.V7I4.11788.
[56] X. Gong, R. Yuan, H. Qian, Y. Chen, and A. G. Cohn, "Emotion Regulation Music Recommendation Based on Feature Selection," *Frontiers in Artificial Intelligence and Applications*, vol. 337, pp. 486–495, Sep. 2021, doi: 10.3233/FAIA210047.
[57] A. Tripathi, N. Bhoj, M. Khari, and B. Pandey, "Feature Selection and Scaling for Random Forest Powered Malware Detection System", doi: 10.21203/RS.3.RS-778333/V1.
[58] K. Kira and L. A. Rendell, "The Feature Selection Problem: Traditional Methods and a New Algorithm," in *AAAI'92: Proceedings of the tenth national conference on Artificial intelligence*, 1992, pp. 129–134.
[59] C. Zhang, M. Ye, L. Lei, and Y. Qian, "Feature Selection for Cross-Scene Hyperspectral Image Classification Using Cross-Domain I-ReliefF," *IEEE J Sel Top Appl Earth Obs Remote Sens*, vol. 14, pp. 5932–5949, 2021, doi: 10.1109/JSTARS.2021.3086151.
[60] Y. Zhou, R. Zhang, S. Wang, and F. Wang, "Feature Selection Method Based on High-Resolution Remote Sensing Images and the Effect of Sensitive Features on Classification Accuracy," *Sensors*, vol. 18, no. 7, Jul. 2018, doi: 10.3390/S18072013.
[61] L. Sun, X. Kong, J. Xu, Z. Xue, R. Zhai, and S. Zhang, "A Hybrid Gene Selection Method Based on ReliefF and Ant Colony Optimization Algorithm for Tumor Classification," *Sci Rep*, vol. 9, no. 1, Dec. 2019, doi:

10.1038/S41598-019-45223-X.
[62] H. Ding and L. Huang, "Extraction of soybean planting areas based on multi-temporal Sentinel-1/2 data," *Third International Conference on Computer Vision and Pattern Analysis (ICCPA 2023)*, p. 8, Aug. 2023, doi: 10.1117/12.2684169.
[63] R. Togo *et al.*, "Cardiac sarcoidosis classification with deep convolutional neural network-based features using polar maps," *Comput Biol Med*, vol. 104, pp. 81–86, Jan. 2019, doi: 10.1016/J.COMPBIOMED.2018.11.008.
[64] C. S. Greene, N. M. Penrod, J. Kiralis, and J. H. Moore, "Spatially Uniform ReliefF (SURF) for computationally-efficient filtering of gene-gene interactions," *BioData Min*, vol. 2, no. 1, 2009, doi: 10.1186/1756-0381-2-5.
[65] H. Nasiri and S. A. Alavi, "A Novel Framework Based on Deep Learning and ANOVA Feature Selection Method for Diagnosis of COVID-19 Cases from Chest X-Ray Images," *Comput Intell Neurosci*, vol. 2022, 2022, doi: 10.1155/2022/4694567.
[66] M. O. Arowol, S. O. Abdulsalam, R. M. Isiaka, and K. A. Gbolagade, "A Hybrid Dimensionality Reduction Model for Classification of Microarray Dataset," *International Journal of Information Technology and Computer Science*, vol. 9, no. 11, pp. 57–63, Nov. 2017, doi: 10.5815/IJITCS.2017.11.06.
[67] G. F. Dong, L. Zheng, S. H. Huang, J. Gao, and Y. C. Zuo, "Amino Acid Reduction Can Help to Improve the Identification of Antimicrobial Peptides and Their Functional Activities," *Front Genet*, vol. 12, Apr. 2021, doi: 10.3389/FGENE.2021.669328.
[68] B. Thakur, N. Kumar, and G. Gupta, "Machine learning techniques with ANOVA for the prediction of breast cancer," *International Journal of Advanced Technology and Engineering Exploration*, vol. 9, no. 87, pp. 232–245, Feb. 2022, doi: 10.19101/IJATEE.2021.874555.
[69] F. A. Putra, S. Mandala, and M. Pramudyo, "A Study of Feature Selection Method to Detect Coronary Heart Disease (CHD) on Photoplethysmography (PPG) Signals," *Building of Informatics, Technology and Science (BITS)*, vol. 4, no. 2, Sep. 2022, doi: 10.47065/BITS.V4I2.2259.
[70] S. Suresh and V. P. S. Naidu, "Mahalanobis-ANOVA criterion for optimum feature subset selection in multi-class planetary gear fault diagnosis," *Journal of Vibration and Control*, vol. 28, no. 21–22, pp. 3257–3268, Nov. 2022, doi: 10.1177/10775463211029153.
[71] M. J. Siraj, T. Ahmad, and R. M. Ijtihadie, "Analyzing ANOVA F-test and Sequential Feature Selection for Intrusion Detection Systems," *International Journal of Advances in Soft Computing and Its Applications*, vol. 14, no. 2, pp. 185–194, 2022, doi: 10.15849/IJASCA.220720.13.
[72] P. E. McKight and J. Najab, "Kruskal-Wallis Test," *The Corsini Encyclopedia of Psychology*, pp. 1–1, Jan. 2010, doi: 10.1002/9780470479216.CORPSY0491.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical Approval and Informed Consent

It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Availability of Data and Material

Not applicable.

Plagiarism Statement

This article has been scanned by iThenticate ™.