Empirical Analysis of Effect of Higher Order Harmonics on Guitar Chord Classification

Nipun Sharma and Swati Sharma

Abstract Music notes classification is gathering exponential traction in machine learning. Guitar chords and piano chords classification has given thrust to research in this area. A chord is a collection of multiple notes played together, and if viewed mathematically, it is a collection of frequencies. To view these frequency components, numerous tools are available in SciPy Python library. Analysis of these harmonics reveals that they can be used as features to classify chord types. However, inclusion of all the harmonics as features in the classification problem does not necessarily mean that it would go on and increase the efficiency of the model in terms of its prediction accuracy. On the contrary, it may lead to increase in dimensionality and cause additional burden on the machine to learn and prompt need for dimensionality reduction. Therefore, selection of optimal number of features becomes important. In this paper, we are evaluating the effect of higher order harmonics on the model's efficiency.

Keywords Guitar chords · Machine learning · Pitch class profile · Harmonics · Chromagram

1 Introduction

A musical chord is said to be played if two or more notes are played at the same time. A chord is named after the root or the origin note. There are many chord variations depending on the scale and the order of the notes [1]. The most popular variation or distinction is based on whether the chord is a major chord or a minor chord. The study of chord recognition stems from the chroma features to recognize chords. A chroma vector is a 12×1 vector, where the values represent the energy of the 12 semitones. Pitch class profile (PCP) uses chroma features to recognize chords [2], and it discusses an implementational software that recognizes guitar chords from an actual wave file or the sound signal. It was able to identify the chords in orchestral sounds also. The

N. Sharma (✉) · S. Sharma
Presidency University, Bangalore, India
e-mail: nipun.sharma@presidencyuniversity.in

A. Joshi et al. (eds.), *Information and Communication Technology for Competitive Strategies (ICTCS 2022)*, Lecture Notes in Networks and Systems 623,
https://doi.org/10.1007/978-981-19-9638-2_56

software operated in a three-step procedure. In the first step, it performed the discrete Fourier transform (DFT) spectrum of the input sound signal. In the second step, it creates the PCP. The concluding step is to perform pattern matching on the created PCP to throw the result out in the form of the chord name. The basic understanding of the chord recognition system hints at making improvements in the second step that is to create better PCPs so that the classification is precise and robust. The following paragraph highlights the developmental progress in this regard.

PCP produces 12 feature extraction coefficients, and they essentially represent the power of fundamental frequencies that are present in the chord. It is established that feature extraction is an integral part of the chord recognition systems. There are multiple feature extraction techniques like improved chromagram [3], improved PCP [4], chroma DCT-reduced log pitch (CRP) enhanced PCP [5], segment averaging with SHPS and logarithmic scaling [6] have reduced the coefficients. The chord recognition system has been tested under two conditions, with and without simplified harmonic product spectrum (SHPS), and the results are compared. The use of SHPS has increased the recognition rate of the chords. Segment averaging and subsampling [7] proposed a feature extraction sub-system which can be used in a chord recognition system. The proposed work is based on segment averaging and subsampling. Recognition rate of above 98% category is achieved with 7 feature extraction coefficients. A calculation pipeline in the form of a CNN feature extractor is explored for synthesized MIDI data. It performs automatic chord recognition by combining deep CNN acoustic feature extractor and the bidirectional long short-term memory-conditional random fields (BLSTM-CRF) sequence decoder [8]. A novel process of detecting variations in harmonic content for sound and music signal uses harmonic change detection function (HCDF) [9]. However, the most important and researched feature is the chromagram, and a chroma extractor using deep neural network is also present [10]. The latest in guitar chord sensing and recognition utilizes multi-task learning using robotics [11]. All these techniques strive to reduce the features and increase the recognition rate. The last of the mentioned techniques has been able to reduce the number of feature extraction coefficients to 6 and also maintaining the recognition rate at 91.43%.

Contributions of the paper:

1. The paper briefly reviews the most important fundamental techniques, PCP and chromagram, in guitar chord recognition along with its improvements over time.
2. The paper presents the experimental results on a dataset of major and minor guitar chords. It implements feature engineering and dimensionality reduction of PCP and chromagram on guitar chord tonality classification problem.

1.1 Harmonics in Guitar Chords

The most intriguing property associated with a note is its frequency. Every note is composed of fixed frequencies that basically defines that note. To illustrate this fact, let's consider A note that can have frequencies like 55, 110, 220, 440 Hz and so on. Every time we double the frequency of the note, it moves to the next octave or we can say, we move one octave higher. Also, there are 12 notes in western music, viz., [A, A#, B, C, C#, D, D#, E, F, F#, G, G#] which are equally spaced on a logarithmic scale. To go from A note to A# note, we can simply multiply by a factor of $2^{1/12}$. Further, a chord can be thought of as several notes being played simultaneously instead of sequentially. Major chords and minor chords are the most popular ones, both involve playing exactly 3 notes simultaneously. A root note is selected first. To play a major chord, progression is 2 whole steps, followed by 1.5 steps from root note. On the other hand to play a minor chord, progression changes to 1.5 steps first and then 2 whole steps from the root node. In real life, musical sound waves do not exist as a single frequency but as harmonics which are integer multiples of base frequency. This implies that playing a particular note will contain all the integer multiple frequencies as its harmonics. Playing a chord will result in an overlapping of all the harmonics of individual notes thus creating music. Table 1 gives a clear distribution of harmonics (integral multiple of base frequency for a particular note) across 8 octaves.

Visualization of notes frequencies reveal that frequencies are present along with their harmonic frequencies. The presence of these harmonic frequencies provide symmetry to chords and additional features for a machine to learn, predict and classify. Usually, the classification of guitar chords employs analysis of intervals between the harmonics. However, the Fourier transform of the chords results in up to 38 harmonic values at the frequency spectrum output. The higher order harmonics are inconsistent amongst different chords and chord types. Higher order harmonics may or may not exist in all chords, which result in a lot of void values in the dataset. It thereby results in increased feature engineering efforts. To reduce that effort, it is advised to analyse only those harmonic intervals which are significant enough to contribute meaningfully to the machine learning process. This paper explores the effect of higher order harmonics on the guitar chord classification problem. We essentially start with the dataset that includes up to 4th harmonic and go all the way up to dataset with 9th harmonic inclusion. We evaluate the model's performance for major and minor chord classification problem for 6 different datasets that include harmonic values from 4 to 9th harmonic. The presence of harmonics in any sound or musical note signal is an inherent property. Even if we play A note that is supposed to be 220 Hz, the frequency spectrum of the note clearly depicts that it contains many other frequency components which are integral multiples of 220 Hz. Moreover, it is the presence of the harmonics that led to the evolution of harmonic notes and beauty in the sound of music. To appreciate this fact, Fig. 1 takes the time domain and frequency domain graphs of the note C. The harmonics of C note are obtained at [131, 165, 196, 262, 330, 392, 496, 525, 588, 659, 787, 826, 989, 1050, 1158, 1179, 1312, 1319, 1376, 1576, 1838] Hz.

Table 1 Frequency distribution of harmonics for musical notes

Octaves	Notes											
	A	A#	B	C	C#	D	D#	E	F	F#	G	G#
0	55	58.3	61.7	65.4	69.3	73.4	77.8	82.4	87.3	92.5	98	103.8
1	110	116.5	123.5	130.8	138.6	146.8	155.6	164.8	174.6	185	196	207.7
2	220	233.1	246.9	261.6	277.2	293.7	311.1	329.6	349.2	370	392	415.3
3	440	466.2	493.9	523.3	554.4	587.3	622.3	659.3	698.5	740	784	830.6
4	880	932.3	987.8	1046.5	1108.7	1174.7	1244.5	1318.5	1396.9	1480	1568	1661.2
5	1760	1864.7	1975.5	2093	2217.5	2349.3	2489	2637	2793.8	2960	3136	3322.4
6	3520	3729.3	3951.1	4186	4434.9	4698.6	4978	5274	5587.7	5919.9	6271.9	6644.9
7	7040	7458.6	7902.1	8372	8869.8	9397.3	9956.1	10,548.1	11,175.3	11,839.8	12,543.9	13,289.8

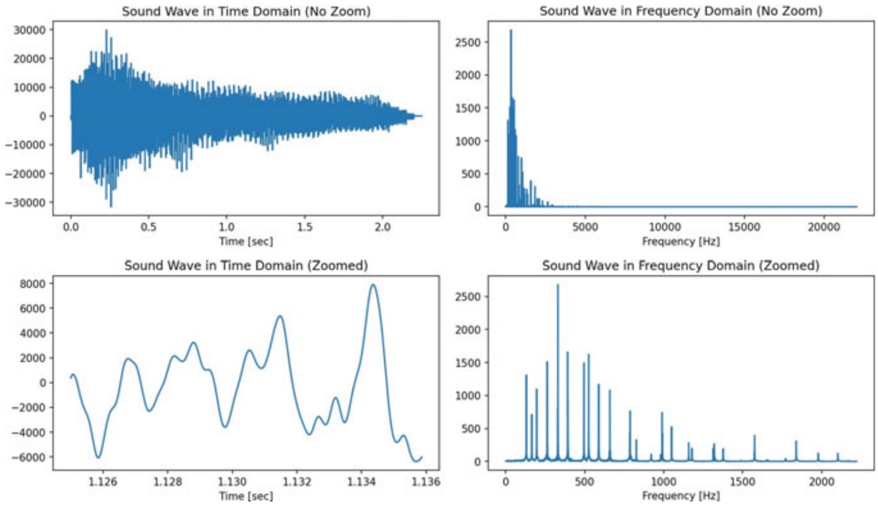

Fig. 1 Sound wave analysis for presence of harmonics

2 Methodology

Guitar chords are primarily classified into 2 major groups, major chords and minor chords. These chords are formed from the major and minor scales and the differentiated from each other by the notes distance from the root note which is discussed in the next paragraph. The tonality of a chord refers to the mood of the sound that it creates; the major chords are characterized by their happy sound, and the minor chords sound sad. Recognition of chords and their type has been the key research focus in this domain. The paper focuses on intuitive feature reduction based on the analysis of datasets that include varying number of harmonics. The guitar chord audio files in the dataset are labelled as major or minor chords. The model is trained with 70% of the data using 6 classification algorithms. The performance of all the six classification algorithms is monitored every time a feature in the form of a harmonic is added to the dataset. The initial model training is done with only the harmonic difference up to 4th harmonic, and with each model training iteration, a new harmonic difference is added. We read a wav file and save it in an array that is in the time domain. By applying the Fourier transform, we obtain an array in the frequency domain. Also, spectrogram is applied to obtain a 2D matrix that has both time and frequency information. The time domain array is not suitable for feature extraction. We analyse the frequency array, and the peak values in the frequency plot are harmonics which we are going to record and keep safe. The frequencies at which the peaks occur can be used to build a model. To do this, we use *find_peaks* method from SciPy library of Python which returns the indices of peaks, and when these indices are plugged into the frequency stamp array, harmonic frequencies are obtained. If a peak is seen at a really small value like 2 Hz, we can safely ignore that, and as a matter of fact,

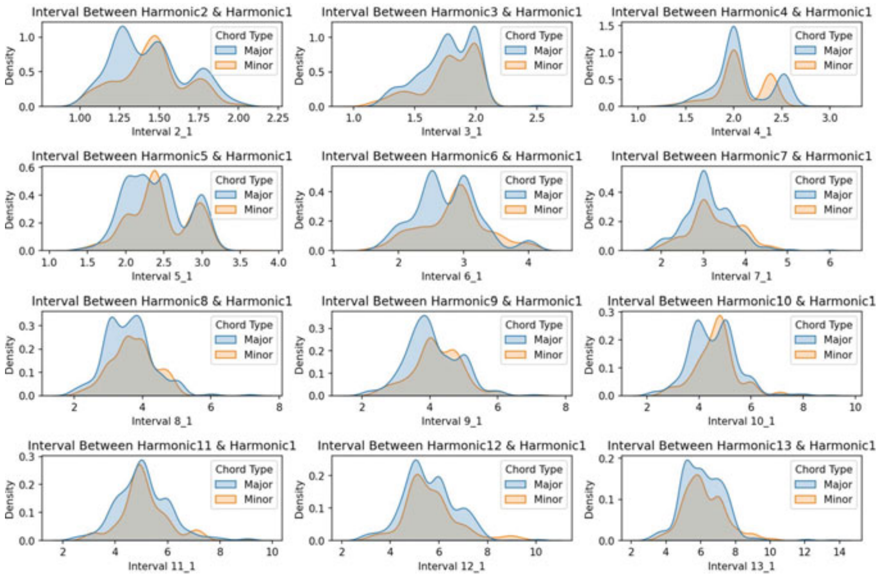

Fig. 2 Intervals between harmonics for major and minor chords with respect to harmonic 1

any peak less than 50 Hz can be ignored cause that might well as be noise in most probability. After the harmonics are extracted from all the wave files, the next phase is to prepare the datasets of these harmonics, and after the dataset is prepared, it is important to find the interval differences between harmonics. This is important because it is the pattern of the interval difference only that will distinguish a major chord from a minor chord. As mentioned in the music theory, a major chord is characterized by 2 full steps distance followed by a 1- and 1/2-step distance between the notes starting from the root note, and minor chord is characterized by 1- and 1/2-step distance followed by a 2-step distance between the notes starting from the root note. The machine is expected to learn from the supervised dataset of more than 800 major and minor chords with features that include the harmonic distances of the root notes. Figure 2 illustrates the distances/intervals between the harmonics with respect to harmonic 1. Although the graphs are plotted up to the 13th harmonic, but we are going to include only the intervals of harmonic 9 with respect to harmonic 1. These harmonic intervals are included in a pandas dataset, and the target variable is set to chord type.

2.1 Inclusion Criteria

The experiment is performed on the dataset with 859 chords classified as major and minor chords which is processed via Python's pandas to form a data frame. As the resultant data frame includes labelled targets, therefore, we have chosen six machine learning algorithms (Linear regression [12], K-nearest neighbours [13–16], support vector machine [17], Gaussian Naïve Bayes [18], decision tree classifier [19] and random forest classifier [20]) perfectly skilled for classification problems. The six machine learning algorithms are well documented in Python's sklearn library and are most popularly used in the classification problems. Each algorithm has its own pros and cons; each one is best suited to perform classification based on the type and size of the dataset. However, it is important to use all six of these on a similar dataset to find out which one works best for further experimentation. The best suited algorithms for chord recognition may be further explored for hyper-parameter tunings. The result analysis of a specific algorithm, for example KNN, in our case suggested that it performs best with the least number of features as discussed in the results section. It opens the door for the further exploration of the said algorithm in future scope.

2.2 Time Complexity of the Proposed Work

The work is performed on a regular Intel 5 core machine with 16-GB RAM, as the dataset consists of 859 chords, so the execution time of the entire notebook file is not significantly large. However, feature engineering requires careful calibration of the data frame.

3 Results

The models for six machine learning algorithms, viz., *Linear Regression, Support Vector Machine, Decision Tree Classifier, Random Forest Classifier, K-Nearest Neighbours and Gaussian Naïve Bayes* are trained on datasets that include varying number of features which depend on the harmonic difference with respect to the first harmonic. The comprehensive test results in Table 2 show that out of the 6 machine learning algorithms tested, 4 of them (Linear regression, support vector machine, decision tree classifier and random forest classifier) give peak performance when harmonic differences of 7th harmonic are included. The performance never increased or rather decreased in all the cases when higher order harmonics (8th, 9th and beyond) are taken. In the other 2 cases, (K-nearest neighbours and Gaussian Naïve Bayes) gave peak performance at inclusion of 4th harmonic and 5th harmonic, respectively. Most interesting picks from the results obtained are that of KNN, because it gives peak performance with the least number of features and random forest classifier, because it gives the highest performance out of all permutations and combinations. So, these two algorithms can be selected for future work in enhancing performance by tuning hyperparameters. However, the results also cement the fact that higher order harmonics are very unlikely to improve the accuracy of the guitar tonality classification.

Table 2 Accuracy score of 6 machine learning algorithms with different harmonic inclusions

Classifier	Harmonics					
	UPTO-4th harmonic	UPTO-5th harmonic	UPTO-6th harmonic	UPTO-7th harmonic	UPTO-8th harmonic	UPTO-9th harmonic
Linear regression	0.57	0.56	0.57	0.6	0.6	0.58
K-nearest neighbours	0.84	0.83	0.83	0.81	0.77	0.75
Support vector machine	0.61	0.57	0.59	0.61	0.6	0.58
Gaussian Naive Bayes	0.66	0.7	0.68	0.64	0.63	0.61
Decision tree classifier	0.89	0.9	0.91	0.92	0.91	0.92
Random forest classifier	0.91	0.91	0.92	0.93	0.92	0.92

Figure 3 clearly shows a more visually comprehensible result of the discussion of the results.

4 Conclusion

Guitar chord classification has been one of the most worked upon topic in music recognition field, and chromagram is extensively used for this purpose. Chromagram essentially utilizes the spectrum of frequencies in the chord, and this paper presents an extensive analysis of harmonics derived from chromagram concept. Utilization of harmonic intervals of guitar chords is presented for the purpose of classification problem. The paper clearly identifies that the classification accuracy of the guitar chord tonality does not increase when the higher order harmonics are included in the features list. All the machine learning algorithms peak at 7th harmonic inclusion and the higher order harmonics 8th and above can safely be excluded as features. The inclusion of higher order harmonics also comes with a problem of missing not a numeric (NaN) values that requires additional feature engineering efforts, so a concrete decision of their exclusion can help the data analysts speed up their workflow. The paper presents a novel way of utilizing the chords data for this observation that can be further explored also.

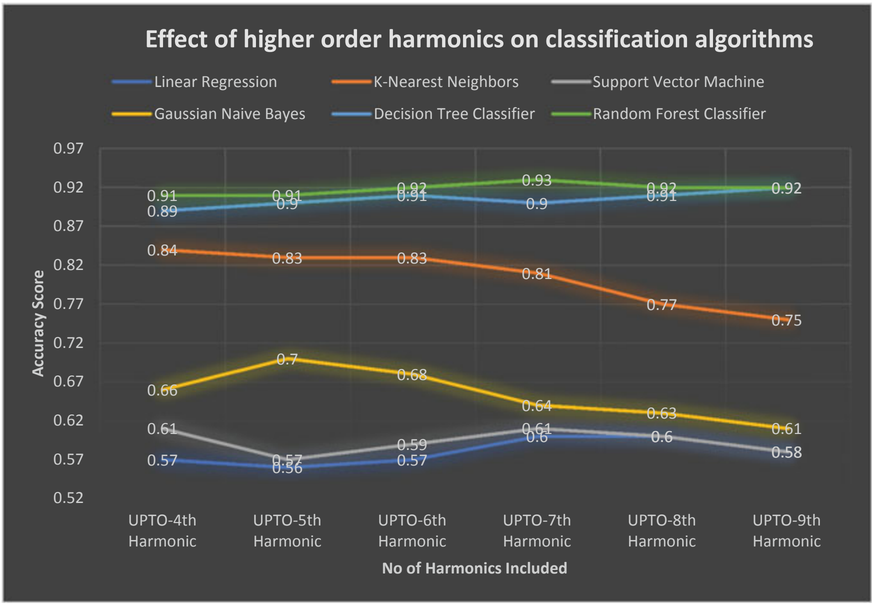

Fig. 3 Variation of accuracy with the inclusion of higher order harmonics

References

1. Obispo SL, Hartquist J (2012) Real-time musical analysis of polyphonic guitar audio
2. Fujishima T (1999) Realtime chord recognition of musical sound: a system using common lisp music. In: ICMC proceedings
3. Stark AM, Plumbley MD (2009) Realtime chord recognition for live performance. In: Proceedings of the international computer music conference (ICMC 2009)
4. Ma K Automatic chord recognition
5. Rajpurkar P, Girardeau B, Migimatsu T A supervised approach to musical chord recognition
6. Sumarno L (2018) Chord recognition using segment averaging feature extraction with simplified harmonic product spectrum and logarithmic scaling. Int J Electr Eng Inform 10(4):753–764. https://doi.org/10.15676/ijeei.2018.10.4.9
7. Sumarno L (2020) Chord recognition using FFT based segment averaging and subsampling feature extraction
8. Wu Y, Li W (2019) Automatic audio chord recognition with MIDI-trained deep feature and BLSTM-CRF sequence decoding model. IEEE/ACM Trans Audio Speech Lang Process 27(2):355–366. https://doi.org/10.1109/TASLP.2018.2879399
9. Harte C, Sandler M, Gasser M (2006) Detecting harmonic change in musical audio. In: Proceedings of the ACM international multimedia conference and exhibition, pp 21–26. https://doi.org/10.1145/1178723.1178727
10. Korzeniowski F, Widmer G Feature learning for chord recognition: the deep chroma extractor. (Online). Available: https://rodrigob.github.io/are
11. Byambatsogt G, Choimaa L, Koutaki G (2020) Guitar chord sensing and recognition using multi-task learning and physical data augmentation with robotics. Sensors (Switzerland) 20(21):1–17. MDPI AG. https://doi.org/10.3390/s20216077
12. Kumari K, Yadav S (2018) Linear regression analysis study. J Pract Cardiovasc Sci 4(1):33. https://doi.org/10.4103/jpcs.jpcs_8_18

13. Cunningham P, Delany SJ (2021) K-nearest neighbour classifiers—a tutorial. ACM Comput Surv 54(6). Association for Computing Machinery. https://doi.org/10.1145/3459665
14. Zhang H, Berg AC, Maire M, Malik J (2006) SVM-KNN: discriminative nearest neighbor classification for visual category recognition
15. Han E-H, Karypis G, Kumar V Text categorization using weight adjusted k-nearest neighbor classification. (Online). Available: http://www.cs.umn.edu/
16. Wettschereck D, Dietterich TG Locally adaptive nearest neighbor algorithms
17. Evgeniou T, Pontil M (2001) Support vector machines: theory and applications. In: Lecture notes in computer science (including subseries Lecture notes in artificial intelligence and lecture notes in bioinformatics), vol 2049 LNAI, pp 249–257. https://doi.org/10.1007/3-540-44673-7_12
18. Bustamante C, Garrido L, Soto R (2006) Comparing fuzzy Naive Bayes and Gaussian Naive Bayes for decision making in RoboCup 3D. In: Lecture notes in computer science (including subseries Lecture notes in artificial intelligence and lecture notes in bioinformatics), vol 4293 LNAI, pp 237–247. https://doi.org/10.1007/11925231_23
19. Patel HH, Prajapati P (2018) Study and analysis of decision tree based classification algorithms. Int J Comput Sci Eng 6(10):74–78. https://doi.org/10.26438/ijcse/v6i10.7478
20. Khan R, Ahmad N, Ali J, Maqsood I (2012) Random forests and decision trees automatic interior room color designing view project distributed deep learning view project random forests and decision trees. (Online). Available: www.IJCSI.org