Statistics and Its Interface Volume 13 (2020) 361–371

A sequential naïve Bayes method for music genre classification based on transitional information from pitch and beat

Tunan Ren, Feifei Wang*, and Hansheng Wang

Due to the rapid development of digital music market, online music websites are widely available in our daily life. There is a practical need to develop automatic music genre classification algorithms to manage a huge amount of music. In this regard, the transitional information contained in pitches and beats should be very useful. Particularly, the transition in pitches produces a melody, and the transition in beats produces a rhythm. They both decide the music genre. To take these valuable information into consideration, we propose here a sequential naïve Bayes method for music genre classification. This method can be viewed as an novel extension of the classical naïve Bayes classifier, but takes the transitional information between pitches and beats into consideration. To reduce the number of estimated parameters, we propose a BIC-type criterion and develop a computationally efficient algorithm for model selection. The selection consistency of the BIC method is theoretically proved and numerically investigated. The finite sample performance of the proposed methods are assessed through both simulations and a real music dataset.

Keywords and phrases: BIC, Music genre classification, Pitch and beat, Selection consistency, Sequential naïve Bayes.

1. INTRODUCTION

Information technology makes digital music widely available over the Internet [12, 2]. Under the flourishment of digital music market, many online music websites and applications emerge. They focus on providing digital music services to users, such as downloading, subscription streaming, and so on [26]. For example, *iTunes Store*, a software-based online digital media store operated by Apple Inc. has offered about 40 million songs available for download [30]. *QQ Music*, one of the top music service providers in China, has the copyrights of 17 million songs and reaches over 700 million users by 2017. How to manage such a huge amount of music files becomes a great challenge.

To handle this challenge, music files are often divided into various classes to annotate its characteristics [19, 21]. For example, the rock song *Beat It* has been classified into classes *Michael Jackson* for its artist and *English* for its language. The Beethoven's famous light music *Dance of the Little Swans* has been classified into the class *Piano* for its played instrument. Such classes are easy to identify, since their classification standards are relatively objective. However, many other classes, especially the music *genres*, have no clear classification standards. In practice, the corresponding classification work mainly relies on human efforts, which is labor intensive, time consuming, and error prone [19, 17]. Hence, it is of great need to develop an automatic algorithm for music genre classification.

To this end, various information sources (e.g., lyrics, audio files, and music scores) have been considered. Classification using lyrics is essentially a text classification problem. Therefore, various classification methods developed for texts can be readily applied for music [16, 14, 13]. However, there are a lot of music without lyrics, such as the classical piano music. This fact limits the application of lyrics-based classification methods. Since the audio files contain all the audio signals of music, classification based on audio files becomes very useful. In this regard, various features should be extracted from audio files and standard classification methods can be applied [12, 7, 11, 18]. In practice, however, there also exist scenarios, where both audio files and lyrics are inaccessible. See for examples the *Musescore* (https://musescore.com) and *Chinese Score* (http://www.qupu123.com) websites, which are online communities for music score creation and online sharing. In this case, only music scores (usually formatted in MusicXML) are available for music genre classification. Therefore, how to extract high-quality symbolic features from music scores to well represent music content becomes important.

A variety of symbolic features have been taken into consideration, covering dimensions of music, such as timbre, pitch, harmony and rhythm [21, 5, 3, 4, 24, 1, 15]. See [6] for an overview. The symbolic features can provide high-level music representations, since they are extracted from the basic elements of music (e.g., note) and present music in a way supposed for performers. The combination of audio-based features and symbolic features are also adopted in music genre classification to gain benefits from both representations. One notable work is [5], in which the authors

*Corresponding author.

extracted symbolic and audio features from MIDI data and found the use of both types of features can improve the classification accuracy.

In this work, we focus on the situation where only music scores are available and try to develop genre classification methods based on the transitional information of pitches and beats only. It is generally believed that the transition in pitches produces a melody, and the transition in beats produces a rhythm [9]. Music in different genre classes usually have different styles of melody and rhythm, and thus exhibit different transitional patterns of pitch and beat. For example, Figure 1 shows two examples of music scores for a piece of folk music and a piece of classical music, respectively. It is obvious that the transitional patterns of pitches and beats for the folk music are more mild than those for the classic music. In addition, compared to audio files, the pitch and beat information extracted from music score is smaller in storage and easier to process. It is also free from influences caused by instrument change. Therefore, it is of great interest to take advantages of transitional information of pitches and beats to address the problem of music genre classification.

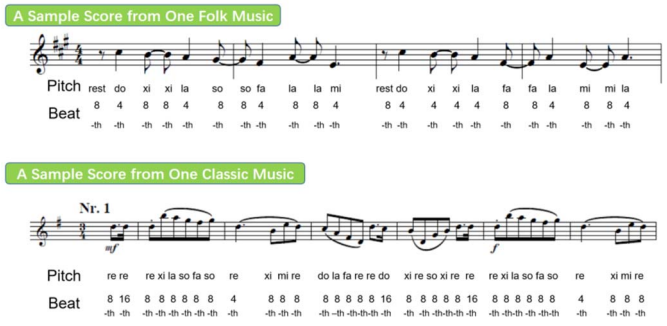

Figure 1. Examples of music scores for a piece of folk music (the top) and a piece of classical music (the bottom).

To this end, we propose a sequential naïve Bayes method for music genre classification by making use of the transitional information of pitches and beats only. Under assumptions that the pitch sequence and beat sequence are independent and follow the Markov property, we model the transitions of pitches and beats in a sequential manner. Our method can be viewed as an novel extension of the classical naïve Bayes classifier, but takes the transitional information of pitches and beats into consideration. Practically, we often have a large pitch space, which results in a large set of pitch transitional probabilities to be estimated. To reduce the number of parameters, a BIC-type criterion is proposed for model selection. For a practical implementation, a computationally efficient algorithm is also developed.

The rest of this paper is organised as follows. Section 2 introduces the sequential naïve Bayes method and the BIC method. Section 3 demonstrates the finite sample performance of our method through simulation studies. Section 4 conducts real data analysis. Section 5 concludes the paper with a brief discussion.

2. THE METHODOLOGY

2.1 Model and notations

We consider a total of n music score files, denoted by $\mathcal{S}_i$ $(i = 1, \cdots, n)$. There are two pieces of information contained in $\mathcal{S}_i$, i.e., the pitch sequence and beat sequence, respectively. Thus, we can write $\mathcal{S}_i = \{(X_{it}, B_{it}) : 1 \le t \le T_i\}$, where X_{it} represents the pitch, B_{it} represents the beat, and T_i is the total number of pitches and beats contained in $\mathcal{S}_i$. We further define $\mathcal{X}$ and $\mathcal{B}$ as the state space of pitch and beat, respectively. Each state in $\mathcal{X}$ corresponds to one key position on a piano keyboard. In this work, we consider a standard piano keyboard with 88 keys, plus the *rest*. As a result, a total of $88 + 1 = 89$ states are contained in $\mathcal{X}$. As for $\mathcal{B}$, we consider six typical beats, i.e., $\mathcal{B} = \{\text{whole, half, quarter, 8th, 16th, 32nd}\}$. Figure 2 illustrates all the states in $\mathcal{X}$ and $\mathcal{B}$. Lastly, let $Y_i \in \mathcal{Y} = \{1, ..., K\}$ represent the class label of $\mathcal{S}_i$.

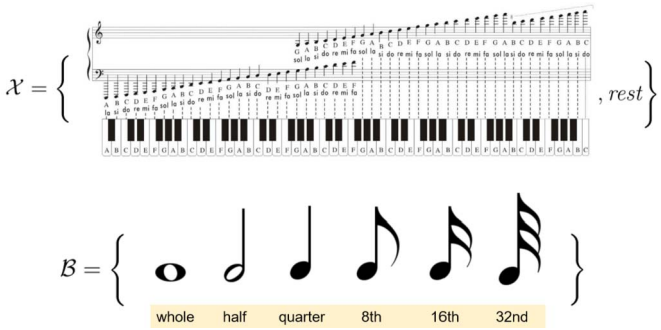

Figure 2. Illustration of the state space of pitch (the top) and beat (the bottom).

To model the relationship between $\mathcal{S}_i$ and Y_i, we propose the following sequential naïve Bayesian (SNB) model. Assume each Y_i is independently generated according to $\pi_k = P(Y_i = k) > 0$ and $\sum_{k\in\mathcal{Y}} \pi_k = 1$. Given Y_i, the music score $\mathcal{S}_i = \{(X_{it}, B_{it}) : 1 \le t \le T_i\}$ is generated in a sequential manner. Specifically, for $t = 1$, the first pitch X_{i1} and beat B_{i1} are generated according to

$$P(X_{i1} = x|Y_i = k) = p_{x,k},$$
$$P(B_{i1} = b|Y_i = k) = q_{b,k},$$

where $x \in \mathcal{X}$, $b \in \mathcal{B}$, $p_{x,k}$ and $q_{b,k}$ are probabilities satisfying $\sum_{x\in\mathcal{X}} p_{x,k} = 1$ and $\sum_{b\in\mathcal{B}} q_{b,k} = 1$. For any $1 < t \le T_i$, denote $\mathcal{F}_{it}$ to be the σ-field generated by $\{(X_{is}, B_{is}) : s \le t\}$. We first assume the sequences of pitch and beat follow

the Markov property. That is, $P(X_{it}, B_{it}|\mathcal{F}_{it-1}, Y_i = k) = P(X_{it}, B_{it}|X_{it-1}, B_{it-1}, Y_i = k)$. In addition, we assume the pitch sequence and beat sequence are independent from each other. Therefore, the pitch X_{it} and beat B_{it} are generated according to

$$\begin{aligned} &P(X_{it}|X_{it-1}, B_{it-1}, Y_i = k) \\ &= P(X_{it} = x_2|X_{it-1} = x_1, Y_i = k) = \omega_{x_1x_2,k}, \end{aligned}$$

and

$$\begin{aligned} &P(B_{it}|X_{it-1}, B_{it-1}, Y_i = k) \\ &= P(B_{it} = b_2|B_{it-1} = b_1, Y_i = k) = \tau_{b_1b_2,k}, \end{aligned}$$

where $x_1, x_2 \in \mathcal{X}$, $b_1, b_2 \in \mathcal{B}$, $\omega_{x_1x_2,k}$ and $\tau_{b_1b_2,k}$ are transitional probabilities satisfying $\sum_{x_2\in\mathcal{X}} \omega_{x_1x_2,k} = 1$ and $\sum_{b_2\in\mathcal{B}} \tau_{b_1b_2,k} = 1$. Then the conditional probability of $\mathcal{S}_i$ given $Y_i = k$ can be derived as

$$\begin{aligned} &P(\mathcal{S}_i|Y_i = k) = P(X_{i1}, B_{i1}|Y_i = k) \\ &\times \prod_{t=2}^{T_i} P(X_{it}, B_{it}|X_{it-1}, B_{it-1}, Y_i = k) \\ &= (p_{X_{i1},k})(q_{B_{i1},k}) \left\{ \prod_{t=2}^{T_i} (\omega_{X_{it}X_{it-1},k})(\tau_{B_{it}B_{it-1},k}) \right\} \\ &= \Big(\prod_{x\in\mathcal{X}} p_{x,k}^{I(X_{i1}=x)} \Big) \Big(\prod_{b\in\mathcal{B}} q_{b,k}^{I(B_{i1}=b)} \Big) \\ &\times \prod_{t=2}^{T_i} \Big(\prod_{x_1,x_2\in\mathcal{X}} \omega_{x_1x_2,k}^{I(X_{it-1}=x_1, X_{it}=x_2)} \Big) \\ &\times \prod_{t=2}^{T_i} \Big(\prod_{b_1,b_2\in\mathcal{B}} \tau_{b_1b_2,k}^{I(B_{it-1}=b_1, B_{it}=b_2)} \Big), \end{aligned} \tag{2.1}$$

where $I(\cdot)$ denotes the indicator function. In the fourth line of (2.1), we rewrite the probabilities $p_{X_{i1},k}$ and $q_{B_{i1},k}$ using all possible pitch values and beat values; while in the fifth and sixth lines, we rewrite the probabilities $\omega_{X_{it}X_{it-1},k}$ and $\tau_{B_{it}B_{it-1},k}$ using all possible pitch pairs and beat pairs, respectively. Given $P(\mathcal{S}_i|Y_i = k)$, we discuss how to conduct music genre classification in the next section.

2.2 Music genre classification

To conduct music genre classification, we first derive $P(Y_i = k|\mathcal{S}_i)$ according to the Bayes' theorem as

$$\begin{aligned} P(Y_i = k|\mathcal{S}_i) &= \frac{P(Y_i = k)P(\mathcal{S}_i|Y_i = k)}{\sum_{y\in\mathcal{Y}} P(Y_i = y)P(\mathcal{S}_i|Y_i = y)} \\ &= \pi_k \Big(\prod_{x\in\mathcal{X}} p_{x,k}^{I(X_{i1}=x)} \Big) \Big(\prod_{b\in\mathcal{B}} q_{b,k}^{I(B_{i1}=b)} \Big) \\ &\times \prod_{t=2}^{T_i} \Big(\prod_{x_1,x_2\in\mathcal{X}} \omega_{x_1x_2,k}^{I(X_{it-1}=x_1, X_{it}=x_2)} \Big) \\ &\times \prod_{t=2}^{T_i} \Big(\prod_{b_1,b_2\in\mathcal{B}} \tau_{b_1b_2,k}^{I(B_{it-1}=b_1, B_{it}=b_2)} \Big) \\ &\times \Big[\sum_{y\in\mathcal{Y}} \pi_y \Big\{ \prod_{x\in\mathcal{X}} p_{x,y}^{I(X_{i1}=x)} \prod_{b\in\mathcal{B}} q_{b,y}^{I(B_{i1}=b)} \\ &\times \prod_{t=2}^{T_i} \Big(\prod_{x_1,x_2\in\mathcal{X}} \omega_{x_1x_2,y}^{I(X_{it-1}=x_1, X_{it}=x_2)} \Big) \\ &\times \prod_{t=2}^{T_i} \Big(\prod_{b_1,b_2\in\mathcal{B}} \tau_{b_1b_2,y}^{I(B_{it-1}=b_1, B_{it}=b_2)} \Big) \Big\} \Big]^{-1}. \end{aligned} \tag{2.2}$$

If all the probabilities (e.g., $\pi_k, p_{x,k}, q_{b,k}, \omega_{x_1x_2,k}, \tau_{b_1b_2,k}$) in (2.2) are known, then we can predict the class label of $\mathcal{S}_i$ according to the maximum posterior probability. However, since these probabilities are unknown, we need to estimate them first and thus regard them as unknown parameters. Let $\theta = \{\pi_k, p_{x,k}, q_{b,k}, \omega_{x_1x_2,k}, \tau_{b_1b_2,k} : k \in \mathcal{Y}, x, x_1, x_2 \in \mathcal{X}, b, b_1, b_2 \in \mathcal{B}\}$ be the full parameter set. To estimate θ, we derive the log-likelihood function as follows,

$$\begin{aligned} \ell(\theta) &= \log \Big\{ \prod_{i=1}^{n} P(\mathcal{S}_i, Y_i) \Big\} = \sum_{i=1}^{n} \log \{P(Y_i)P(\mathcal{S}_i|Y_i)\} \\ &= \sum_{i=1}^{n} \sum_{k\in\mathcal{Y}} I(Y_i = k) \log \{P(Y_i = k)P(\mathcal{S}_i|Y_i = k)\} \\ &= \sum_{i=1}^{n} \sum_{k\in\mathcal{Y}} I(Y_i = k) \Big\{ \log(\pi_k) + \sum_{x\in\mathcal{X}} I(X_{i1} = x) \\ &\times \log(p_{x,k}) + \sum_{b\in\mathcal{B}} I(B_{i1} = b) \log(q_{b,k}) \\ &+ \sum_{t=2}^{T_i} \sum_{x_1,x_2\in\mathcal{X}} I(X_{it-1} = x_1, X_{it} = x_2) \log(\omega_{x_1x_2,k}) \\ &+ \sum_{t=2}^{T_i} \sum_{b_1,b_2\in\mathcal{B}} I(B_{it-1} = b_1, B_{it} = b_2) \log(\tau_{b_1b_2,k}) \Big\}. \end{aligned} \tag{2.3}$$

In addition, these unknown probabilities should satisfy some constraints. They are, $\sum_{k=1}^{K} \pi_k = 1$, $\sum_{x\in\mathcal{X}} p_{x,k} = 1$, $\sum_{b\in\mathcal{B}} q_{b,k} = 1$, $\sum_{x_2\in\mathcal{X}} \omega_{x_1x_2,k} = 1$, and $\sum_{b_2\in\mathcal{B}} \omega_{b_1b_2,k} = 1$ for any k. Thus, by maximizing (2.3) under these con-

straints, we can obtain the maximum likelihood estimator:

(2.4)

$$\widehat{\pi}_k = n^{-1}\sum_{i=1}^{n} I(Y_i = k),$$

$$\widehat{p}_{x,k} = \frac{\sum_{i=1}^{n} I(Y_i = k, X_{i1} = x)}{\sum_{i=1}^{n} I(Y_i = k)},$$

$$\widehat{q}_{b,k} = \frac{\sum_{i=1}^{n} I(Y_i = k, B_{i1} = b)}{\sum_{i=1}^{n} I(Y_i = k)},$$

$$\widehat{\omega}_{x_1x_2,k} = \Big\{\sum_{i=1}^{n}\sum_{t=2}^{T_i} I(Y_i = k)I(X_{it-1} = x_1, X_{it} = x_2)\Big\} \times \Big\{\sum_{i=1}^{n}\sum_{t=2}^{T_i} I(Y_i = k)I(X_{it-1} = x_1)\Big\}^{-1},$$

$$\widehat{\tau}_{b_1b_2,k} = \Big\{\sum_{i=1}^{n}\sum_{t=2}^{T_i} I(Y_i = k)I(B_{it-1} = b_1, B_{it} = b_2)\Big\} \times \Big\{\sum_{i=1}^{n}\sum_{t=2}^{T_i} I(Y_i = k)I(B_{it-1} = b_1)\Big\}^{-1}.$$

It is notable that, the maximum likelihood estimates in (2.4) happen to be equal to the empirical estimates using histogram statistics. By substituting these estimates into (2.2), we can get the estimated likelihood function $\widehat{P}(Y_i = k|\mathcal{S}_i)$. Then, we can make prediction by $\widehat{y}_i = \text{argmax}_{k\in\mathcal{Y}}\widehat{P}(Y_i = k|\mathcal{S}_i)$.

2.3 A BIC criterion

In SNB, the number of parameters to be estimated is $K \times (1 + M_1 + M_2 + M_1^2 + M_2^2)$, where M_1 and M_2 are the number of states in $\mathcal{X}$ and $\mathcal{B}$, respectively. In practice, the states space of pitch (i.e., $\mathcal{X}$) is relatively large. For example, we have $M_1 = 89$ states in $\mathcal{X}$ in this work. As a result, the number of parameters to be estimated would be large. A sufficient number of parameters can make the model more flexible, but also unstable [27, 10]. Thus, it is of great importance to reduce the number of parameters without sacrificing the classification accuracy.

Note that the large parameter set is mainly resulted from the pitch transitional probabilities, i.e., $\Omega = \{\omega_{x_1x_2,k} : k \in \mathcal{Y}, x_1, x_2 \in \mathcal{X}\}$, which has KM_1^2 parameters in total. Thus, we focus on Ω to conduct feature selection. Let $\mathcal{C}_F = \{(x_1, x_2) : x_1, x_2 \in \mathcal{X}\}$ be the set of all pitch pairs, $\mathcal{C}_0 = \{(x_1, x_2) : \omega_{x_1x_2,k_1} \neq \omega_{x_1x_2,k_2}, \text{for some } k_1 \neq k_2\}$ be the set of pitch pairs that can distinguish between classes, and $\overline{\mathcal{C}}_0 = \{(x_1, x_2) : \omega_{x_1x_2,k_1} = \omega_{x_1x_2,k_2}, \text{for any } k\}$ be the set of pitch pairs having no distinguishing power. As a results, we have $\mathcal{C}_F = \mathcal{C}_0 \bigcup \overline{\mathcal{C}}_0$. Since the pitch pairs in $\overline{\mathcal{C}}_0$ are class-independent, the transitional probabilities associated with $\overline{\mathcal{C}}_0$ can be simplified as $\overline{\Omega}_0 = \{\omega_{x_1x_2} : (x_1, x_2) \in \overline{\mathcal{C}}_0\}$. We can remove $\overline{\Omega}_0$ from the calculation of (2.2) and (2.3), since they are not helpful for classification. This can make the full parameter set largely reduced and the associated computation easier.

To find $\mathcal{C}_0$ from $\mathcal{C}_F$, we propose a BIC-type selection criterion [22, 23]. Specifically, assume all pitch pairs in $\mathcal{C}^*$ are class-dependent, and those in $\overline{\mathcal{C}}^* = \mathcal{C}_F \setminus \mathcal{C}^*$ are class-independent. The BIC criterion given $\mathcal{C}^*$ is written as

$$\text{BIC}(\mathcal{C}^*) = n^{-1}\{-2\ell(\mathcal{C}^*)\} + df \times \log(n)/n,$$

where $\ell(\mathcal{C}^*)$ is the log-likelihood function, and df is the number of parameters involved in $\ell(\mathcal{C}^*)$. To calculate $\text{BIC}(\mathcal{C}^*)$, we need to calculate $\ell(\mathcal{C}^*)$ first. Note that, $\ell(\mathcal{C}^*)$ is the simplified version of (2.3) by partitioning all pitch paries into $\mathcal{C}^*$ and $\overline{\mathcal{C}}^*$, i.e.,

$$\begin{aligned}\ell(\mathcal{C}^*) = &\sum_{i=1}^{n}\sum_{k\in\mathcal{Y}} I(Y_i = k)\Big\{\log(\pi_k) \\ &+ \sum_{x\in\mathcal{X}} I(X_{i1} = x)\log(p_{x,k}) + \sum_{b\in\mathcal{B}} I(B_{i1} = b)\log(q_{b,k}) \\ &+ \sum_{t=2}^{T_i}\sum_{b_1,b_2\in\mathcal{B}} I(B_{it-1} = b_1, B_{it} = b_2)\log(\tau_{b_1b_2,k}) \\ &+ \sum_{t=2}^{T_i}\sum_{(x_1,x_2)\in\mathcal{C}^*} I(X_{it-1} = x_1, X_{it} = x_2)\log(\omega_{x_1x_2,k}) \\ &+ \sum_{t=2}^{T_i}\sum_{(x_1,x_2)\in\overline{\mathcal{C}}^*} I(X_{it-1} = x_1, X_{it} = x_2)\log(\omega_{x_1x_2})\Big\}.\end{aligned}$$

Then, a new set of maximum likelihood estimates can be obtained; see equations in (2.5). By comparing (2.4) and (2.5), we find that $\widetilde{\pi}_k = \widehat{\pi}_k$, $\widetilde{p}_{x,k} = \widehat{p}_{x,k}$, $\widetilde{q}_{b,k} = \widehat{q}_{b,k}$, and $\widetilde{\tau}_{b_1b_2,k} = \widehat{\tau}_{b_1b_2,k}$. For any $(x_1, x_2) \in \mathcal{C}^*$, $\widetilde{\omega}_{x_1x_2,k} = \widehat{\omega}_{x_1x_2,k}$. However, for any $(x_1, x_2) \in \overline{\mathcal{C}}^*$, $\widetilde{\omega}_{x_1x_2} \neq \widehat{\omega}_{x_1x_2,k}$. In summary, when using a simplified version of log-likelihood function, only the transitional probabilities for class-dependent pitch pairs have different estimates. In contrast, the class-independent pitch pairs should have the same estimates.

By substituting (2.5) into $\ell(\mathcal{C}^*)$, we can compute $\text{BIC}(\mathcal{C}^*)$. The final selected set of class-dependent pitch pairs can be obtained by $\widehat{\mathcal{C}} = \arg\min_{\mathcal{C}^*} \text{BIC}(\mathcal{C}^*)$. Before investigating the asymptotic property of the proposed BIC method, the following condition is needed.

(C1) For any $\mathcal{C}$ and its corresponding parameter space $\Omega_{\mathcal{C}} = \{\omega_{x_1x_2,k} : (x_1, x_2) \in \mathcal{C}\} \cup \{\omega_{x_1x_2} : (x_1, x_2) \in \overline{\mathcal{C}}\}$, assume there exists two finite constants $\delta_{\min}$ and $\delta_{\max}$, satisfying $0 < \delta_{\min} \leq \min(\Omega_{\mathcal{C}}) \leq \max(\Omega_{\mathcal{C}}) \leq \delta_{\max} < 1$.

Based on condition (C1), the following theorem confirms the selection consistency [23] of the BIC method, whose detailed proof is left to Appendix A.

Theorem 2.1. *Under the condition (C1), we have* $P(\widehat{\mathcal{C}} = \mathcal{C}_0) \to 1$, *as* $n \to \infty$.

(2.5)

$$\widetilde{\pi}_k = n^{-1}\sum_{i=1}^{n} I(Y_i = k),$$

$$\widetilde{p}_{x,k} = \frac{\sum_{i=1}^{n} I(Y_i = k, X_{i1} = x)}{\sum_{i=1}^{n} I(Y_i = k)},$$

$$\widetilde{q}_{b,k} = \frac{\sum_{i=1}^{n} I(Y_i = k, B_{i1} = b)}{\sum_{i=1}^{n} I(Y_i = k)},$$

$$\widetilde{\tau}_{b_1b_2,k} = \Big\{\sum_{i=1}^{n}\sum_{t=2}^{T_i} I(Y_i = k)I(B_{it-1} = b_1, B_{it} = b_2)\Big\}$$
$$\times\Big\{\sum_{i=1}^{n}\sum_{t=2}^{T_i} I(Y_i = k)I(B_{it-1} = b_1)\Big\}^{-1},$$

$$\widetilde{\omega}_{x_1x_2,k} = \Big\{\sum_{i=1}^{n}\sum_{t=2}^{T_i} I(Y_i = k)I(X_{it-1} = x_1, X_{it} = x_2)\Big\}$$
$$\times\Big\{\sum_{i=1}^{n}\sum_{t=2}^{T_i} I(Y_i = k)I(X_{it-1} = x_1)\Big\}^{-1}, \textit{for } (x_1, x_2) \in \mathcal{C}^*,$$

$$\widetilde{\omega}_{x_1x_2} = \Big\{\sum_{i=1}^{n}\sum_{t=2}^{T_i} I(X_{it-1} = x_1, X_{it} = x_2)\Big\}$$
$$\times\Big\{\sum_{i=1}^{n}\sum_{t=2}^{T_i} I(X_{it-1} = x_1)\Big\}^{-1}, \textit{for } (x_1, x_2) \in \overline{\mathcal{C}}^*.$$

2.4 Fast computation of the BIC criterion

Since $\mathcal{C}_F$ contains M_1^2 pitch pairs, we have a total of $2^{M_1^2}$ candidate models in consideration. Note that M_1 is usually a large number in practice, e.g., $M_1 = 89$ in this work. Computing the BIC criterion for every possible candidate model is computationally infeasible. This motivates us to develop an efficient computation algorithm to find $\widehat{\mathcal{C}}$. To this end, let $\mathcal{C}^*$ be an arbitrary candidate model. For any $(x_1, x_2) \in \mathcal{C}^*$, let $\mathcal{C}^*_{-x_1x_2} = \mathcal{C}^* \setminus \{(x_1, x_2)\}$. Then we have

(2.6)

$$\begin{aligned}
&\mathrm{BIC}(\mathcal{C}^*) - \mathrm{BIC}(\mathcal{C}^*_{-x_1x_2})\\
&= -2n^{-1}\Big\{\ell(\mathcal{C}^*) - \ell(\mathcal{C}^*_{-x_1x_2})\Big\} + K\log(n)/n\\
&= -2n^{-1}\sum_{i=1}^{n}\sum_{k\in\mathcal{Y}}\Big[I(Y_i = k)\sum_{t=2}^{T_i} I(X_{it-1} = x_1, X_{it} = x_2)\\
&\times\big\{\log(\omega_{x_1x_2,k}) - \log(\omega_{x_1x_2})\big\}\Big] + K\log(n)/n.
\end{aligned}$$

Surprisingly, we find (2.6) is independent of $\mathcal{C}^*$, but only depends on the pitch pair (x_1, x_2). This enables us to write $\mathrm{BIC}(\mathcal{C}^*) - \mathrm{BIC}(\mathcal{C}^*_{-x_1x_2})$ as $\Delta(x_1, x_2)$. We can verify that $(x_1, x_2) \in \widehat{\mathcal{C}}$ if and only if $\Delta(x_1, x_2) < 0$; see Theorem 2.2. The detailed proof can be found in Appendix B.

Theorem 2.2. *Define* $\widetilde{\mathcal{C}} = \{(x_1, x_2) : \Delta(x_1, x_2) < 0\}$, *then* $P(\widetilde{\mathcal{C}} = \mathcal{C}_0) \to 1$, *as* $n \to \infty$.

This theorem suggests that the selected model can be computed by comparing $\Delta(x_1, x_2)$ against zero for all pitch pairs in $\mathcal{C}_F$. It results in a total of M_1^2 calculations, making this algorithm computationally feasible.

3. SIMULATION STUDIES

3.1 Data generation

To examine the finite sample performance of the proposed BIC method, we present here a number of simulation studies. For a fixed sample size n, assume there are two classes of music, i.e., $Y_i \in \{0, 1\}$ for $1 \le i \le n$. Further assume $P(Y_i = 1) = P(Y_i = 0) = 0.5$. For simplicity, we only consider the pitch information in music score files. Assume a pitch state space $\mathcal{X} = \{j : 1 \le j \le 30\}$. For each music score file $\mathcal{S}_i$, its initial pitch $X_{i1} = j \in \mathcal{X}$ is generated according to $p_{j,Y=1} = p_{j,Y=0} = 1/30$. Then, for a fixed number of pitches T, we generate the pitch sequence $\{X_{it} : 2 \le t \le T\}$ in a sequential manner, according to a pitch transitional probabilities Ω_{Y_i}.

We design Ω_{Y_i} as follows. Based on $\mathcal{X}$, the set of all pitch pairs is $\mathcal{C}_F = \{(j_1, j_2) : j_1, j_2 \in \mathcal{X}\}$. First, we generate a basic pitch transitional probabilities $\Omega^* = (\omega^*_{j_1j_2})$ with each row generated from a homogeneous Dirichlet distribution with parameter $\alpha = 10$. Next, based on Ω^*, we compute the pitch transitional probabilities $\Omega_1 = (\omega_{j_1j_2,1})$ and $\Omega_0 = (\omega_{j_1j_2,0})$ for score files associated with class label equal to 1 and 0, respectively. Specifically, define $\mathcal{C}_0 = \{(j_1, j_2) : 1 \le j_1, j_2 \le 20\}$ as the true model. For any $(j_1, j_2) \in \mathcal{C}_0$ and j_2 is odd, define $\omega_{j_1j_2,1} = \omega^*_{j_1j_2} + 0.9\omega^*_{j_1j_2+1}$ and $\omega_{j_1j_2+1,1} = 0.1\omega^*_{j_1j_2+1}$. Also, define $\omega_{j_1j_2,0} = 0.1\omega^*_{j_1j_2}$ and $\omega_{j_1j_2+1,0} = 0.9\omega^*_{j_1j_2} + \omega^*_{j_1j_2+1}$. One can verify that $\omega_{j_1j_2,1} - \omega_{j_1j_2,0} = 0.9(\omega^*_{j_1j_2} + \omega^*_{j_1j_2+1}) \neq 0$ and $\omega_{j_1j_2+1,1} - \omega_{j_1j_2+1,0} = -0.9(\omega^*_{j_1j_2} + \omega^*_{j_1j_2+1}) \neq 0$. Therefore, when the pair $(j_1, j_2) \in \mathcal{C}_0$, it can help distinguish between classes. For any pitch pair $(j_1, j_2) \notin \mathcal{C}_0$, let $\omega_{j_1j_2,1} = \omega_{j_1j_2,0} = \omega^*_{j_1j_2}$. Therefore, the pitch pair $(j_1, j_2) \notin \mathcal{C}_0$ is class-independent.

After generating Y_i and $\mathcal{S}_i = \{X_{it}, 1 \le t \le T\}$ for $1 \le i \le n$, we apply the SNB method with the BIC criterion for classification. We consider different settings of T, i.e., $T = 500, 800, 1000$. Under each setting, different sample sizes (i.e., $n = 100$, 200, 500) are evaluated, and the experiment is replicated for $M = 500$ times.

3.2 Performance measurement and simulation results

Let $\widehat{\mathcal{C}}^{(m)}$ $(1 \le m \le M)$ be the BIC model obtained in the m-th replication. To demonstrate the finite sample performance of the BIC model, the following measures are defined. First, we calculate the percentages of underfit (Underfit) and overfit (Overfit) [25] as follows, Underfit $= (M)^{-1}\sum_{m=1}^{M} I\big(\mathcal{C}_0 \setminus \widehat{\mathcal{C}}^{(m)} \neq \varnothing\big)$ and Overfit $=$

Table 1. Simulation results of the BIC *method under different sample sizes and number of pitches. The measures Underfit, Overfit, Correct-Fit, RMS, Correct-Zeros and Incorrect-Zeros are reported for each experiment setting.*

T	n	Underfit	Overfit	Correct-Fit	RMS	Correct-Zeros	Incorrect-Zeros
500	100	0.726	0.186	0.088	0.999	498.946	1.364
	200	0.002	0.390	0.608	1.001	499.502	0.002
	500	0.000	0.168	0.832	1.001	499.820	0.000
800	100	0.036	0.606	0.358	1.002	498.976	0.036
	200	0.000	0.388	0.612	1.001	499.510	0.000
	500	0.000	0.160	0.840	1.000	499.832	0.000
1000	100	0.002	0.666	0.332	1.003	498.918	0.002
	200	0.000	0.362	0.638	1.001	499.536	0.000
	500	0.000	0.144	0.856	1.000	499.840	0.000

$(M)^{-1}\sum_{m=1}^{M} I\big(\mathcal{C}_0 \subset \widehat{\mathcal{C}}^{(m)}\big) \times I\big(\widehat{\mathcal{C}}^{(m)} \neq \mathcal{C}_0\big)$. To measure whether the selected model could perfectly recover the true model, we calculate the percentage of correctly fit (Correct-Fit) as Correct-Fit $= (M)^{-1}\sum_{m=1}^{M} I\big(\widehat{\mathcal{C}}^{(m)} = \mathcal{C}_0\big)$.

To compare the sizes of the BIC model versus the true model, we calculate the relative model size (RMS), i.e., RMS $= M^{-1}\sum_{m=1}^{M}\big(|\widehat{\mathcal{C}}^{(m)}|/|\mathcal{C}_0|\big)$. It is remarkable that, under the Theorem 2.1, RMS should be close to 1 when n is sufficiently large. Lastly, to further examine the performance of the BIC model, we calculate the number of correct zeros (Correct-Zeros) and incorrect zeros (Incorrect-Zeros), from the perspective of model size [29]. Specifically, Correct-Zeros $= (M)^{-1}\sum_{m=1}^{M}\sum_{(j_1,j_2)} I\big((j_1,j_2) \notin \mathcal{C}_0\big) \times I\big((j_1,j_2) \notin \widehat{\mathcal{C}}^{(m)}\big)$, and Incorrect-Zeros $= (M)^{-1}\sum_{m=1}^{M}\sum_{(j_1,j_2)} I\big((j_1,j_2) \in \mathcal{C}_0\big) \times I\big((j_1,j_2) \notin \widehat{\mathcal{C}}^{(m)}\big)$.

The simulation results are summarized in Table 1. As the sample size n increases, both Underfit and Overfit decrease, but Correct-Fit approaches to 1. This finding verifies the selection consistency of the BIC method, which is in accordance with Theorem 2.2. As for model sizes, Correct-Zeros are close to its true value 500, while Incorrect-Zeros are close to 0. Accordingly, RMS are all close to 1. These findings demonstrate that the BIC method could help to select the right pitch pairs.

4. REAL DATA ANALYSIS

4.1 Data description

To further demonstrate the performance of our proposed SNB model, we present here a real music score dataset. It contains three different genre classes, i.e., *Classic*, *Jazz* and *Folk*. The corresponding sample sizes for each class are 198, 210 and 199, respectively. The original music scores are in the MIDI storage format. We first transferred each music into MUSICXML format using the software PDFTOMUSIC PRO [8]. Then we extracted the pitch sequence and beat sequence from the MUSICXML format of each music. The music data and corresponding computational codes are available on https://github.com/rtngsm/SequenceMusicScore.

To have a better understanding of music scores, we provide some descriptive analysis results related to pitches. We first calculate the occurring frequency of each pitch or pitch pair in the dataset. Then we take the top ten pitches or pitch pairs for illustration and compare their occurring frequencies in different classes. As shown in Figure 3, music in different genre classes have different habits of using pitches and pitch pairs. However, pitch pairs change more dramatically than pitches across the three classes, which suggests the transitional information of pitch pairs could be more helpful in music genre classification.

4.2 Music genre classification using the SNB model

We then apply the SNB method for music genre classification. We use "SNB+FULL" to represent the model based on the full parameter set, and "SNB+BIC" to represent the selected model using the BIC criterion. For comparison, we consider $4 \times 2 = 8$ benchmark methods. They are constructed as follows. First, we consider 4 most popularly used classification methods. They are, respectively, naïve Bayes (NB) classifier, decision tree (DT), support vector machine (SVM) and neural network (NN). The naïve Bayes classifier uses the Gaussian assumption. The decision tree uses the entropy splitting criterion. The SVM uses the polynomial kernel. The neural network has three hidden layers, in which the number of neurons are 20, 20 and 10, respectively. All the four classifiers are implemented using the SKLEARN library [20] in Python. Next, we consider two different types of features. The first type of features contains the marginal probabilities of pitch pairs and beat pairs calculated by their

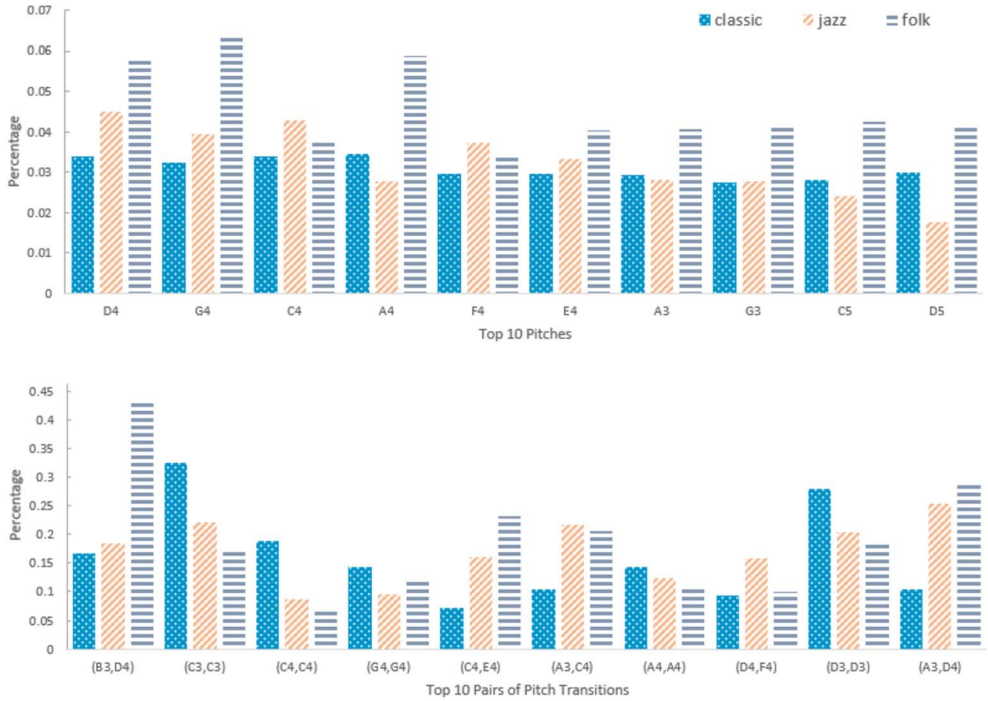

Figure 3. Occurring frequencies of the top ten pitches (top panel) and pitch pairs (bottom panel) in Classic, Jazz *and* Folk *music.*

occurring frequencies. The second type contains audio features extracted directly from the MIDI files following the method of [28]. To summarize, a total of 4 classification methods and 2 different types of features are considered. It is notable that each classification method is regarded as a multivariate model, in which all features of a particular type are considered together for classification.

Except for the 8 benchmark methods, we also make comparisons with the classification method in [5], which has shown good classification performance using symbolic and audio features extracted from MIDI. In particular, to get audio features, the MIDI data was transferred into audio files (in the WAV format) using different combinations of sample rates and sample sizes. After feature extraction, two classifiers, i.e., linear discriminant classifier (LDC) and k-nearest neighbor (KNN) were used on different feature sets for music genre classification. Finally, with a pool of classifiers, we applied two voting strategies, i.e., the constant weight (CW) and proportional weight (PW), on all classifiers to achieve better accuracy.

For validation purpose, we randomly split the data into a training dataset (with 304 observations) and a testing dataset (with 303 observations). The partition of data is repeated for 1000 times. Table 2 presents the detailed classification accuracy on testing datasets for our proposed SNB methods and different benchmark methods. For each method, the main descriptive statistics of out-sample classification accuracy on 1000 data partitions are reported. As shown in Table 2, the SNB methods have achieved better classification performance than all benchmark methods. For example, the mean value of classification accuracy obtained by SNB+BIC is 87.00%, while the best among all benchmark methods is only 85.72%.

In addition, based on the 1000 accuracies obtained by each method, we test whether the SNB methods have achieved significantly better performance than the benchmark methods. Specifically, let q_A denote the classification accuracy achieved by the SNB methods (i.e., A=SNB+FULL, SNB+BIC). Let q_B denote the classification accuracy achieved by each benchmark method. Then let $d_{AB} = q_A - q_B$ denote the difference. We then test the null hypothesis $H_0 : d_{AB} = 0$ versus $H_1 : d_{AB} > 0$ using the paired t-test. The test results are shown in Table 2, in which all tests have p.values smaller than 0.01. The test results suggest that, SNB+FULL and SNB+BIC have both achieved significantly higher classification accuracies than all benchmark methods. It is notable that, the outstanding performance of SNB results largely from its strength in better describing the natural generation process of music. That is, the pitches and beats are generated one by one under certain transitional probabilities. Therefore, the transitional information of pitches and beats are important features to music genres; see Figure 3 for an illustration. Accordingly, the SNB model taking advantage of these transitional information is helpful to music genre classification.

*Table 2. The detailed classification accuracy on testing datasets for different classification methods. We report the main descriptive statistics (i.e., min, median, mean, max and standard deviation) of out-sample classification accuracy on 1000 data partitions. In addition, the significance of paired t-test by comparing SNB methods with benchmark methods are also reported, where ** and *** indicate p.values smaller than 0.01 and 0.001, respectively.*

Feature Type/Method		Min	Median	Mean	Max	SD	Test with SNB	
							FULL	BIC
SNB	FULL	77.63%	86.51%	86.31%	93.42%	2.11%	–	–
	BIC	78.62%	87.17%	87.00%	93.75%	1.99%		
Marginal	NB	63.49%	75.99%	75.55%	84.54%	3.07%	***	***
	DT	66.78%	78.95%	78.97%	85.86%	2.44%	***	***
	SVM	55.26%	78.29%	77.78%	82.57%	3.20%	***	***
	NN	33.88%	87.50%	85.72%	95.39%	7.33%	**	***
Audio	NB	67.43%	75.99%	75.85%	83.22%	2.36%	***	***
	DT	73.03%	82.89%	82.85%	89.80%	2.33%	***	***
	SVM	71.05%	78.95%	78.90%	85.86%	2.49%	***	***
	NN	16.78%	67.43%	61.38%	82.24%	14.32%	***	***
Voting	CW	73.89%	79.93%	80.10%	86.62%	1.91%	***	***
	PW	73.89%	79.94%	80.12%	87.58%	1.97%	***	***

5. CONCLUDING REMARKS

In this work, we develop a sequential naïve Bayes model using transitional information of pitches and beats for music genre classification. To reduce the number of parameters to be estimated, we propose a BIC-type criterion, along with a computationally efficient algorithm for model selection. The selection consistency of the BIC method is theoretically proved. The finite sample performance of the methods are elaborated by both simulation studies and a real data example. To conclude this work, we consider several directions for future study. First, more pitch and beat information, such as the *chord*, can be considered for genre classification. Second, the transitional information of pitches and beats can be modeled dynamically, not fixed. Last, the Markov property of pitch sequence and beat sequence can be generalized to allow more complex situations.

APPENDIX A. PROOF OF THEOREM 2.1

To prove this theorem, we consider two cases, according to whether $\mathcal{C} \subset \mathcal{C}_F$ is an overfitted or underfitted model.

(Case 1. Overfitted Model) Let $\mathcal{C}$ be an overfitted model, i.e., $\mathcal{C} \in \mathcal{Q}_+ = \{\mathcal{C} : \mathcal{C}_0 \subset \mathcal{C}, \mathcal{C}_0 \neq \mathcal{C}\}$. Then we have $n\{\mathrm{BIC}(\mathcal{C}) - \mathrm{BIC}(\mathcal{C}_0)\} = 2\left\{\widetilde{\ell}(\mathcal{C}_0) - \widetilde{\ell}(\mathcal{C})\right\} + \{|\mathcal{C}| - |\mathcal{C}_0|\}\log(n)$, where $\widetilde{\ell}(\mathcal{C}_0)$ and $\widetilde{\ell}(\mathcal{C})$ are the maximum values of the corresponding log-likelihood functions, i.e.,

$$
\begin{aligned}
&\widetilde{\ell}(\mathcal{C}_0) - \widetilde{\ell}(\mathcal{C}) \\
=&\sum_{i=1}^{n}\sum_{k\in\mathcal{Y}} I(Y_i = k)\Big[\sum_{t=2}^{T_i}\Big\{\sum_{(x_1,x_2)\in\mathcal{C}_0} I(X_{it-1} = x_1, \\
&X_{it} = x_2)\log(\widetilde{\omega}_{x_1x_2,k}) \\
+&\sum_{(x_1,x_2)\in\overline{\mathcal{C}}_0} I(X_{it-1} = x_1, X_{it} = x_2)\log(\widetilde{\omega}_{x_1x_2}) \\
-&\sum_{(x_1,x_2)\in\mathcal{C}} I(X_{it-1} = x_1, X_{it} = x_2)\log(\widetilde{\omega}_{x_1x_2,k}) \\
-&\sum_{(x_1,x_2)\in\overline{\mathcal{C}}} I(X_{it-1} = x_1, X_{it} = x_2)\log(\widetilde{\omega}_{x_1x_2})\Big\}\Big] \\
=&\sum_{i=1}^{n}\sum_{k\in\mathcal{Y}} I(Y_i = k)\Big[\sum_{t=2}^{T_i}\sum_{(x_1,x_2)\in\mathcal{C}\setminus\mathcal{C}_0} \\
&I(X_{it-1} = x_1, X_{it} = x_2)\log(\frac{\widetilde{\omega}_{x_1x_2}}{\widetilde{\omega}_{x_1x_2,k}})\Big].
\end{aligned}
$$

When $(x_1, x_2) \in \mathcal{C} \setminus \mathcal{C}_0$, we have $\omega_{x_1x_2,k} = \omega_{x_1x_2}$, for any $k \in \mathcal{Y}$. According to the consistency property of MLE, we have $\widetilde{\omega}_{x_1x_2} - \omega_{x_1x_2} = O_p(1/\sqrt{n})$. Then under the condition (C1) and the Slusky Theorem, one can verify

that $\log(\widetilde{\omega}_{x_1x_2}/\widetilde{\omega}_{x_1x_2,k}) = O_p(1/\sqrt{n})$. Furthermore, define

$$W_i = \sum_{k\in\mathcal{Y}} I(Y_i = k)\Big[\sum_{t=2}^{T_i}\sum_{(x_1,x_2)\in\mathcal{C}\setminus\mathcal{C}_0} I(X_{it-1}=x_1, X_{it}=x_2)\log(\frac{\widetilde{\omega}_{x_1x_2}}{\widetilde{\omega}_{x_1x_2,k}})\Big],$$

and $W = \sum_{i=1}^n W_i$. One can verify that $\{W_i : i = 1,\cdots,n\}$ are independent and $W_i = O_p(1/\sqrt{n})$. This leads to $var(W) = \sum_{i=1}^n var(W_i) = O(1)$, which implies that $W = O_p(1)$. Hence, we have

$$\begin{aligned} n\big\{\mathrm{BIC}(\mathcal{C}) - \mathrm{BIC}(\mathcal{C}_{\mathbf{0}})\big\} &= 2\big\{\widetilde{\ell}(\mathcal{C}_0) - \widetilde{\ell}(\mathcal{C})\big\} + (|\mathcal{C}| - |\mathcal{C}_0|)\log n \\ &\geq O_p(1) + \log n \to \infty. \end{aligned}$$

Furthermore, since $|\mathcal{C}_F| = M_1^2 < \infty$, we have

$$P\big\{\inf_{\mathcal{C}\in\mathcal{Q}_+}\mathrm{BIC}(\mathcal{C}) > \mathrm{BIC}(\mathcal{C}_{\mathbf{0}})\big\} \to \mathbf{1}.$$

(Case 2. Underfitted Model) Let $\mathcal{C}$ be an underfitted model, i.e., $\mathcal{C} \in \mathcal{Q}_- = \{\mathcal{C} : \mathcal{C}_0 \not\subset \mathcal{C}\}$. Similarly, we have $n\{\mathrm{BIC}(\mathcal{C}) - \mathrm{BIC}(\mathcal{C}_{\mathbf{0}})\} = 2\left\{\widetilde{\ell}(\mathcal{C}_0) - \widetilde{\ell}(\mathcal{C})\right\} + \{|\mathcal{C}| - |\mathcal{C}_0|\}\log(n)$. In this case, we have

$$\begin{aligned} &\widetilde{\ell}(\mathcal{C}_0) - \widetilde{\ell}(\mathcal{C}) \\ =&\sum_{i=1}^n\sum_{k\in\mathcal{Y}} I(Y_i=k)\Big[\sum_{t=2}^{T_i}\sum_{(x_1,x_2)\in\mathcal{C}\setminus\mathcal{C}_0} I(X_{it-1}=x_1, X_{it}=x_2)\log(\frac{\widetilde{\omega}_{x_1x_2}}{\widetilde{\omega}_{x_1x_2,k}})\Big] \\ +&\sum_{i=1}^n\sum_{k\in\mathcal{Y}} I(Y_i=k)\Big[\sum_{t=2}^{T_i}\sum_{(x_1,x_2)\in\mathcal{C}_0\setminus\mathcal{C}} I(X_{it-1}=x_1, X_{it}=x_2)\log(\frac{\widetilde{\omega}_{x_1x_2,k}}{\widetilde{\omega}_{x_1x_2}})\Big] \\ =&L_1 + L_2. \end{aligned}$$

By proof of Case 1, we have

$$\begin{aligned} L_1 &= \sum_{i=1}^n\sum_{k\in\mathcal{Y}} I(Y_i=k)\Big[\sum_{t=2}^{T_i}\sum_{(x_1,x_2)\in\mathcal{C}\setminus\mathcal{C}_0} I(X_{it-1}=x_1, X_{it}=x_2)\log(\frac{\widetilde{\omega}_{x_1x_2}}{\widetilde{\omega}_{x_1x_2,k}})\Big] \\ &= O_p(1). \end{aligned}$$

Then we focus on

$$L_2 = \sum_{i=1}^n\sum_{k\in\mathcal{Y}} I(Y_i=k)\Big[\sum_{t=2}^{T_i}\sum_{(x_1,x_2)\in\mathcal{C}_0\setminus\mathcal{C}} I(X_{it-1}=x_1, X_{it}=x_2)\log(\frac{\widetilde{\omega}_{x_1x_2,k}}{\widetilde{\omega}_{x_1x_2}})\Big].$$

Further define

$$\begin{aligned} R_i &= \sum_{k\in\mathcal{Y}} I(Y_i=k)\Big[\sum_{t=2}^{T_i}\sum_{(x_1,x_2)\in\mathcal{C}_0\setminus\mathcal{C}} I(X_{it-1}=x_1, X_{it}=x_2)\log(\frac{\widetilde{\omega}_{x_1x_2,k}}{\widetilde{\omega}_{x_1x_2}})\Big] \\ &= \sum_{k\in\mathcal{Y}}\sum_{(x_1,x_2)\in\mathcal{C}_0\setminus\mathcal{C}} I(Y_i=k)\Big[\sum_{t=2}^{T_i} I(X_{it-1}=x_1, X_{it}=x_2)\log(\widetilde{\omega}_{x_1x_2,k})\Big] \\ &- \sum_{k\in\mathcal{Y}}\sum_{(x_1,x_2)\in\mathcal{C}_0\setminus\mathcal{C}} I(Y_i=k)\Big[\sum_{t=2}^{T_i} (X_{it-1}=x_1, X_{it}=x_2)\log(\widetilde{\omega}_{x_1x_2})\Big] \\ &= \sum_{k\in\mathcal{Y}}\sum_{(x_1,x_2)\in\mathcal{C}_0\setminus\mathcal{C}}\sum_{t=2}^{T_i} I(Y_i=k, X_{it-1}=x_1, X_{it}=x_2)\log(\widetilde{\omega}_{x_1x_2,k}) \\ &- \sum_{k\in\mathcal{Y}}\sum_{(x_1,x_2)\in\mathcal{C}_0\setminus\mathcal{C}}\sum_{t=2}^{T_i} I(Y_i=k, X_{it-1}=x_1, X_{it}=x_2)\log(\widetilde{\omega}_{x_1x_2}). \end{aligned}$$

Accordingly, we have $L_2 = \sum_{i=1}^n R_i$. When $(x_1,x_2) \in \mathcal{C}_0 \setminus \mathcal{C}$, we have $\omega_{x_1x_2,k} \neq \omega_{x_1x_2}$, for some $k \in \mathcal{Y}$. This leads to $\widetilde{\omega}_{x_1x_2,k} \neq \widetilde{\omega}_{x_1x_2}$ as $n \to \infty$. According to the property of MLE, it is obvious that $R_i > 0$. Let $c_{\min} = \min(R_i)$, which is a positive number. Then, we have $\mathrm{BIC}(\mathcal{C}) - \mathrm{BIC}(\mathcal{C}_{\mathbf{0}}) \geq 2c_{\min} + o_p(1)$, which is a positive number with probability tending to one. Consequently, we have $P\big\{\inf_{\mathcal{C}\in\mathcal{Q}_-}\mathrm{BIC}(\mathcal{C}) > \mathrm{BIC}(\mathcal{C}_0)\big\} \to 1$. Based on the results of Case 1 and Case 2, we have $P\big\{\inf_{\mathcal{C}\in\mathcal{Q}_-\cup\mathcal{Q}_+}\mathrm{BIC}(\mathcal{C}) > \mathrm{BIC}(\mathcal{C}_0)\big\} = P\big\{\widehat{\mathcal{C}} = \mathcal{C}_0\big\} \to 1$, which completes the entire proof.

APPENDIX B. PROOF OF THEOREM 2.2

To prove this theorem, we only need to verify that when $\widetilde{\mathcal{C}} = \big\{(x_1,x_2) : \Delta(x_1,x_2) < 0\big\}$, we have $\mathrm{BIC}(\widetilde{\mathcal{C}}) = \min_{\mathcal{C}\subset\mathcal{C}_{\mathcal{F}}}\mathrm{BIC}(\mathcal{C})$. We use the counter-evidence method to demonstrate the above conclusion. Assume $\mathcal{C}^* = \min_{\mathcal{C}\subset\mathcal{C}_F}\mathrm{BIC}(\mathcal{C})$, and there exists $(x_1^*,x_2^*) \in \mathcal{C}^*$, such that $\Delta(x_1^*,x_2^*) \geq 0$. Then we have $\mathcal{C}^*_{-(x_1,x_2)} = \mathcal{C}^*\setminus\{(x_1^*,x_2^*)\}$, and now we have $\mathrm{BIC}(\mathcal{C}^*) - \mathrm{BIC}(\mathcal{C}^*_{-(x_1,x_2)}) = \Delta(x_1^*,x_2^*) \geq 0$. This leads to contradiction. Then by Theorem 2.1, we have $P(\widetilde{\mathcal{C}} = \mathcal{C}_0) \to 1$ as $n \to \infty$.

ACKNOWLEDGEMENTS

Feifei Wang's research is supported by National Natural Science Foundation of China (No. 11971504), the Fundamental Research Funds for the Central Universities

and the Research Funds of Renmin University of China (No. 18XNLG02), National Key R&D Program of China (No. 2018YFC0830300), Ministry of Education Focus on Humanities and Social Science Research Base (Major Research Plan 17JJD910001). Hansheng Wang's research is partially supported by National Natural Science Foundation of China (No. 11831008, 11525101, 71532001). It is also supported in part by China's National Key Research Special Program (No. 2016YFC0207704).

Received 9 October 2019

REFERENCES

[1] Abesser, J., Lukashevich, H., and Bräuer, P. (2012). Classification of music genres based on repetitive basslines. *Journal of New Music Research*, 41(3):239–257.

[2] Aguiar, L. and Martens, B. (2016). Digital music consumption on the Internet: Evidence from clickstream data. *Information Economics and Policy*, 34:27–43.

[3] Ahonen, T. E., Lemström, K., and Linkola, S. (2011). Compression-based similarity measures in symbolic, polyphonic music. In *Proceedings of the 12th International Society for Music Information Retrieval Conference*, pages 91–96.

[4] Anan, Y., Hatano, K., Bannai, H., and Takeda, M. (2011). Music genre classification using similarity functions. In *Proceedings of the 12th International Society for Music Information Retrieval Conference*, pages 693–698.

[5] Cataltepe, Z., Yaslan, Y., and Sonmez, A. (2007). Music genre classification using midi and audio features. *EURASIP Journal on Advances in Signal Processing*, 2007:1–8.

[6] Corrêa, D. C. and Rodrigues, F. A. (2016). A survey on symbolic data-based music genre classification. *Expert Systems with Applications*, 60:190–210.

[7] Costa, Y. M. G., Oliveira, L. S., and Silla, C. N. (2017). An evaluation of convolutional neural networks for music classification using spectrograms. *Applied Soft Computing*, 52:28–38.

[8] Doerfler, G. and Beck, R. (2013). An approach to classifying four-part music. In *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV)*, pages 1–8.

[9] Dor, O. and Reich, Y. (2011). An evaluation of musical score characteristics for automatic classification of composers. *Computer Music Journal*, 35(3):86–97.

[10] Fan, J., Feng, Y., Jiang, J., and Tong, X. (2016). Feature augmentation via nonparametrics and selection in high-dimensional classification. *Journal of the American Statistical Association*, 111(513):275–287. MR3494659

[11] Ferretti, S. (2017). On the modeling of musical solos as complex networks. *Information Sciences*, 375:271–295.

[12] Fu, Z., Lu, G., Ting, K. M., and Zhang, D. (2011). Music classification via the bag-of-features approach. *Pattern Recognition Letters*, 32(14):1768–1777.

[13] Furuya, M., Huang, H. H. and Kawagoe, K. (2015). Evaluation of music classification method based on lyrics of English songs. In *Proceedings of the International MultiConference of Engineers and Computer Scientists*, pages 134–137.

[14] Hu, X., Downie, J. S., and Ehmann, A. F. (2009). Lyric text mining in music mood classification. In *10th International Society for Music Information Retrieval Conference*, pages 411–416.

[15] Hübler, S. and Hoffmann, R. (2013). Modelling drum patterns with weighted finite-state transducers. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 719–723.

[16] Laurier, C., Grivolla, J., and Herrera, P. (2008). Multimodal music mood classification using audio and lyrics. In *2008 Seventh International Conference on Machine Learning and Applications*, pages 688–693.

[17] Lee, C., Shih, J., Yu, K., and Lin, H. (2009). Automatic music genre classification based on modulation spectral analysis of spectral and cepstral features. *IEEE Transactions on Multimedia*, 11(4):670–682.

[18] Lee, J., Park, J., Kim, K. L., and Nam, J. (2018). SampleCNN: End-to-end deep convolutional neural networks using very small filters for music classification. *Applied Sciences*, 8(1):150–163.

[19] Li, T., Ogihara, M., and Li, Q. (2003). A comparative study on content-based music genre classification. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 282–289.

[20] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830. MR2854348

[21] Scaringella, N., Zoia, G., and Mlynek, D. (2006). Automatic genre classification of music content: A survey. *IEEE Signal Processing Magazine*, 23(2):133–141.

[22] Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464. MR0468014

[23] Shao, J. (1997). An asymptotic theory for linear model selection. *Statistica Sinica*, 7:221–242. MR1466682

[24] Simsekli, U. (2010). Automatic music genre classification using bass lines. In *2010 20th International Conference on Pattern Recognition*, pages 4137–4140.

[25] Speed, T. P. and Yu, B. (1993). Model selection and prediction: Normal regression. *Annals of the Institute of Statistical Mathematics*, 45(1):35–54. MR1220289

[26] Tang, D. and Lyons, R. (2016). An ecosystem lens: Putting China's digital music industry into focus. *Global Media and China*, 1(4):350–371.

[27] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B (Methodological)*, 58(1):267–288. MR1379242

[28] Tzanetakis, G. and Cook, P. R. (2002). Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302.

[29] Wang, H. (2009). Forward regression for ultra-high dimensional variable screening. *Journal of the American Statistical Association*, 104(488):1512–1524. MR2750576

[30] Wikipedia (2018). Wikipedia website. http://en.wikipedia.org//wiki//ITunes_Store.

Tunan Ren
Guanghua School of Management
Peking University
Beijing 100871
P. R. China
E-mail address: rtnpku@pku.edu.cn

Feifei Wang
Center for Applied Statistics
Renmin University of China
School of Statistics
Renmin University of China
Beijing, 100082
P. R. China
E-mail address: feifei.wang@ruc.edu.cn

Hansheng Wang
Guanghua School of Management
Peking University
Beijing 100871
P. R. China
E-mail address: hansheng@pku.edu.cn