12-5-2018

Statistical Aspects of Music Mining: Naive Dictionary Representation

Qiuyi Wu
qw9477@rit.edu

ROCHESTER INSTITUTE OF TECHNOLOGY

MASTER THESIS

Statistical Aspects of Music Mining: Naive Dictionary Representation

Author:
Qiuyi WU

Supervisor:
Dr. Ernest FOKOUÉ

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Applied Statistics

in the

College of Science
School of Mathematical Sciences

December 5, 2018

Committee Approval

Name	Signature	Date

Dr. Ernest Fokoué, Thesis Advisor, School of Mathematical Sciences

Dr. Joseph Voelkel, Committee Member, School of Mathematical Sciences

Dr. Robert Parody, Committee Member, School of Mathematical Sciences

Declaration of Authorship

I, Qiuyi WU, declare that this thesis titled, "Statistical Aspects of Music Mining: Naive Dictionary Representation" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

"All men have been created to carry forward an ever-advancing civilization."

Bahá'u'lláh

Rochester Institute of Technology

Abstract

Associate Professor Ernest Fokoué, Chair
School of Mathematical Sciences

Master of Science in Applied Statistics

**Statistical Aspects of Music Mining:
Naive Dictionary Representation**

by Qiuyi WU

Extensive studies have been conducted on both musical scores and audio tracks of western classical music with the finality of learning and detecting the key in which a particular piece of music was played. Both the Bayesian Approach and modern unsupervised learning via latent Dirichlet allocation have been used for such learning tasks. In this research work, we propose and develop the novel idea of treating musical sheets as literary documents in the traditional text analytics parlance, to fully benefit from the vast amount of research already existing in statistical text mining and topic modeling.

We specifically introduce the idea of representing any given piece of music as a collection of "musical words" that we codenamed "muselets", which are essentially musical words of various lengths. Given the novelty and therefore the extremely difficulty of properly forming a complete version of a dictionary of muselets, the present paper focuses on a simpler albeit naive version of the ultimate dictionary, which we refer to as a Naive Dictionary because of the fact that all the words are of the same length. We specifically herein construct a naive dictionary featuring a corpus made up of African American, Chinese, Japanese and Arabic music, on which we perform both supervised and unsupervised learning.

For the exploration of pattern recognition and topic modeling, we venture out of the traditional western classical music and embrace and explore other music genres. We consider the musical score sheets and audio tracks of some of the giants of jazz like Duke Ellington, Miles Davis, John Coltrane, Dizzie Gillespie, Wes Montgomery, Charlie Parker, Sonny Rollins, Louis Armstrong, Bill Evans, Dave Brubeck, Thelonious Monk. We specifically employ Bayesian techniques and modern topic modeling methods to explore tasks such as: automatic improvisation detection, genre identification, and key detection. Although some of the results based on the Naive Dictionary are reasonably good, we anticipate phenomenal predictive performances once we get around to actually build a full scale complete version of our intended dictionary of muselets.

Acknowledgements

First I want to sincerely thank my academic advisor & life mentor Dr. Ernest Fokoue, for his persistent guidance during my Master journey. Thank him for squeezing extra time out of his busy schedule offering me Independent Study in Advanced Statistical Methodology, which paves solid foundation for my further study and research work. Thank him for motivating me to sequentially conduct two research projects on Ensemble of Echo State Networks and Topic Modeling in Music Mining. Thank him for telling me the sky is my limit on the first day of school back in 2016 and keeping bringing world-class scholars to Data Science Research Group every Friday.

I especially express my gratitude to Dr. Robert Parody and Dr. Joseph Voelkel, for their acceptance of being my committee members and generous guidance during my Master study. Dr. Parody as the most welcome professor never fails to bring students a lot of joy in his class. The rigorous training given by Dr. Voelkel helps me to pursue everything with high standard. I also thank all other faculties in College of Science in RIT, specifically Dr. Seshavadhani Kumar, Dr. Dhireesha Kudithipudi, Dr. Matthew Hoffman, Dr. Raluca Felea for their edifying help.

I deliberately push myself into the hard mode of my life and it turns out to be the best thing ever. If I had to end my life now, there would be five people I must give credits to: Uncle William Hui, Miss Yefei Ma, Dr. Ernest P. Fokoué, Dr. David L. Banks, Dr. Edward I. George. My life would never be the same without them. *'If you are a teacher your words can be meaningful, but if you are a compassionate teacher, your words can be especially meaningful.'* Uncle William, Dr. Ernest Fokoué, Dr. David L. Banks and Dr. Edward I. George, are all compassionate teachers letting their hearts burn with love for all who may come cross their lives. I thank Miss Ma, as my soulmate, who witnesses numerous failures behind my every single glorious moment, and persistently embraces the worst version of me.

Special big THANK goes to musicians: thank Lizhu Lu from Eastman School of Music and Gankun Zhang from Brandon University School of Music, for their patience to answer my tons of never-ending music theory questions. Thank Dr. Carl Atkins from Department of Performance Arts & Visual Culture, who is also the president and chair and dean for bunch of departments and organizations, and Professor Kwaku Kwaakye Obeng from Brown University, for their encouragement and technical supports all the time. I would like to show my gratitude to Dr. Jonathan Kruger, Dr. Evans Gouno, Mrs. Rebecca Ann Finnangan Kemp, Dr. David Guidice for sharing their pearls of wisdom during the personal communication on music lexicon.

I sincerely thank professors and postdoctoral scholars and PhDs I met in SAMSI and Duke this year, who nicely treat me like their family, extend my vision to see the possibility of life, and deepen my experience and the resoluteness of my determination to reach the highest heights of excellence. I want to especially thank Dr. James Berger, Dr. Alan Gelfand, Dr. Jong-Min Kim, Dr. Rui Paulo, Dr. Anabel Forte Deltell, Dr. Pierre Barbillon, Dr. Dongchu Sun, Dr. Zhuoqiong He, Dr. Huang Haung, Dr. Cheng Cheng, Dr. Lei Yang, Dr. Chenyang Tao, Dr. Yawen Guan, Dr. Whitney Huang, Dr. Matthias Sachs, Dr. Mikael Kuusela, Dr. Hyungsuk Tak, Dr. Maggie Johnson, Dr. Xinyi Li, Dr. Pulong Ma, Dr. Wenjia Wang, Dr. Christine Peijinn Chai, Hanyu Song, Dr. Akihiko Nishimura, Chi Feng, Federico Felpari Ferrari, Chenyuan

Song, Cong Lin, Hyunjung Lee.

I would like to offer my greatest gratitude to the friends who share memory with me in Rochester: Seyed Hamed Fatemi Langroudi, Anisia Jabin, Shiyang Ma, Zichen Ma, Xingchen Yu, Kenneth Tyler Wilcox, Bohan Liu, Shiteng Yang, Wenbo Sun, Matthew Williams, Mohammad Aqil Azad, Ahmed Sultan Aldhaheri, Shahd Saad Alnofaie, Marcos Michael Soriano Almanzar, Chen Feng, Xiaowen Zhou, Guenadie Nibbs, Manèl Chaîb, Preeti Sah, Glenn Egli, Tom Howe, Debbie & Bob, PohLeng Teo, Shuhuan Zhang, Yuchen Zhong, Yuan Yao, Xingnan Zhang, Yichen Jin, Lujun Yin, Limin He, Nan Mo, Xiaoshan Lin, Muyun Liu.

Most importantly, I want to thank wholeheartedly to my family, especially my parents and grandparents, for their unconditional love and understanding, for freeing me with the audacious imagination, which I believe will go down in history as one of the most significant moves of my career and my life.

Finally, I thank RIT Research & Creativity Reimbursement Program for partially sponsoring my work to have it possibly presented in Joint Statistical Meetings (JSM) this year in Vancouver. I appreciate supports from International Conference on Advances in Interdisciplinary Statistics and Combinatorics (AISC) for a conference registration waiver and NC Young Researcher Award this year. I thank 7th Annual Conference of the Upstate New York Chapters of The American Statistical Association (UP-STAT) for recognizing my work and offering me Gold Medal for Best Student Research Award this year.

Contents

List of Figures

List of Tables

For the memory in Schapiro Hall, the eternal present.

Chapter 1

Introduction

1.1 Motivation

FIGURE 1.1: Titanic in Music piece and Text Body

Music plays a big part of our lives but have you ever think of questions like:
How does music have the power to provoke different emotions? What's the similarity between music from different culture, or composers, or different genres?

Music piece and text articles are very similar in the sense that both carry the information to narrate a certain story. Musicians express their feelings through music while writers record events through words. Take the tragedy *Titanic* in Figure 1.1 as an example, we learn the tragedy from the newspaper and feel anguished, but we can also get the mourning from the song *My Heart Will Go On*. The melody contains a lot of minor keys (e.g. $D\flat, F\sharp, A\flat$), which are more likely to trigger the dissonance via two closely spaced notes hitting the ear simultaneously and thus to make people feel sad.

Here this psychoacoustical topic was transformed into the statistical question. Suppose the melody we hear based on the feeling we gain from the music is denoted as "X_{feel}", the true melody of the song is denoted as "X_{real}", then we would like to obtain the melody as complete as possible. We want to get close as much as possible to the truth:

$$\underset{real}{\operatorname{argmax}}\, p(X_{real}|X_{feel}) \tag{1.1}$$

From Bayesian inference, we can rewrite the posterior probability as:

$$p(X_{real}|X_{feel}) = \frac{p(X_{feel}|X_{real})p(X_{real})}{p(X_{feel})} \tag{1.2}$$

The probability of true melody given the melody we feel is the same as the right side. Since the overall feeling towards the music would not change over time, we can simplify the formula by removing the denominator:

$$\underset{real}{\operatorname{argmax}}\, p(X_{real}|X_{feel}) = \underset{real}{\operatorname{argmax}}\, p(X_{feel}|X_{real})p(X_{real}) \tag{1.3}$$

Now to get the most probable true melody based on our feeling towards the melody, we need to get the likelihood, which is the probability of the feeling for every melody; and the prior probability of the true melody from the current knowledge. Afterwards we can maximize the product to get the melody as close to the real melody as possible.

Of course Bayesian modeling is one of the approaches that works very well in key-detection. Here I use probabilistic topic model and pattern recognition techniques to detect the key in which a particular piece of music is played.

TABLE 1.1: Comparison between Text and Music in Topic Modeling

Text	letter	word	topic	document	corpus
Music	note	notes*	melody	song	album

* **a series of notes in one bar can be regarded as a "word"**

FIGURE 1.2: Piece of Music Melody

Compared with the role of text in Topic Modeling as showed in Table 1.1, we treat a series of notes as "word", can also be called as "term", as single note could not hold enough information for us to interpret, specifically, we treat notes in one bar[3] as one "term". Melody[4] plays the role of "topic", and the melodic materials give the

[3]In musical notation, a bar (or measure) is a segment of time corresponding to a specific number of beats in which each beat is represented by a particular note value and the boundaries of the bar are indicated by vertical bar lines.

[4]Harmony is formed by consecutive notes so that the listener automatically perceives those notes as a connected series of notes.

shape and personality of the music piece. "Melody" is also referred as "key-profile" by Hu and Saul (2009a) in their paper, and this concept was based on the key-finding algorithm from Krumhansl and Schmuckler (1990) and the empirical work from Krumhansl and Kessler (1982). The whole song is regarded as "document" in text mining, and a collection of songs called album in music could be regarded as "corpus" in text mining.

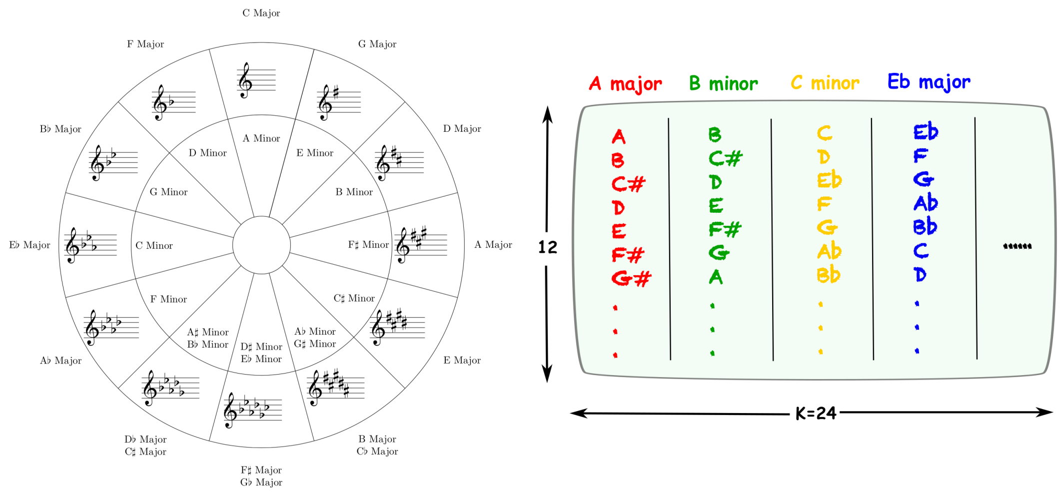


FIGURE 1.3: Circle of Fifths (left) and Key-profiles (right)

Specifically, "key-profile" is chromatic scale showed geometrically in Figure 1.3 Circle of Fifths plot containing 12 pitch classes in total with major key and minor key respectively, thus there are totally 24 key-profiles, each of which is a 12-dimensional vector. The vector in the earliest model in Longuet-Higgins and Steedman (1971) uses indicator with value of 0 and 1 to simply determine the key of a monophonic piece. E.g. C major key-profile:

$$[1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1]$$

As showed in the figures below, Krumhansl and Schmuckler (1990) judge the key in a more robust way. Elements in the vector indicate the stability of each pitch-class corresponding to each key. Melody in the same key-profile would have similar set of notes, and each key-profile is a distribution over notes.

Figure 1.4 (left part) shows the pitch-class distribution of C Major *Piano Sonata No.1, K.279/189d (Mozart, Wolfgang Amadeus)* using K-S key-finding algorithm, and we can see all natural notes: C, D, E, F, G, A, B have high probability to occur than other notes. Figure 1.4 (right part) shows the pitch-class distribution of C Minor *BWV.773 No. 2 in C minor (Bach, Johann Sebastian)* and again we can see specific notes typical for C Minor with higher probability: C, D, D♯, F, G, G♯, and A♯.

Usually different scales could bring different emotions. Generally, major scale arouse buoyant and upbeat feelings while minor scales create dismal and dim environment. Details for emotion and mood effects from musical keys would be presented in later section.

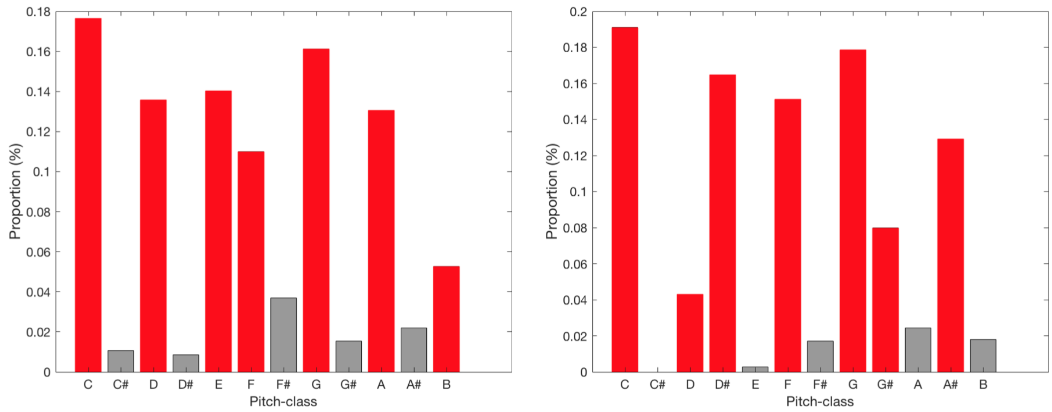


FIGURE 1.4: C minor key-profile (left) and C minor key-profile (right)

1.2 Thesis Scope

This thesis explores various aspects of statistical machine learning methods for music mining with a concentration on soundtracks from Jazz legends like Charlie Parker and Miles Davis. We attempt to create a Naive Lexicon analogy to the language dictionary. That is to say, when people hear a music piece, they are hearing the audio of an essay written with music words.

The target of this research work is to create homomorphism between musical and literature. Instead of decomposing music sheet into a collection of single notes, we attempt to employ direct seamless adaptation of canonical topic modeling on words in order to "topic model" music fragments.

One of the most challenging components is to define the basic unit of the information from which one can formulate a soundtrack as a document. Specifically, if a music soundtrack were to be viewed as a document made up of sentences and phrases, with sentences defined as a collection of words (adjectives, verbs, adverbs and pronouns), several topics would be fascinating to explore:

- What would be the grammatical structure in music?
- What would constitute the jazz lexicon or dictionary from which words are drawn?

All music is story telling as assumption, it is plausible to imagine every piece of music as a collection of words and phrases of variable lengths with adverbs and adjectives and nouns and pronouns.

$$\varphi : \textit{musical sheet} \rightarrow \textit{bag of music words}$$

The construction of the mapping φ is non-trivial and requires deep understanding of music theory. Here several great musicians offer insights on the complexity of φ from their perspectives, to explain about the representation of the input space, namely, creating a mapping from music sheet to collection of music "words" or "phrases":

- *"These are extremely profound questions that you are asking here. I can't answer them within any specific time-frame. I'm interested in trying — I think? But you have* *opened up a whole lot of bigger questions with this than you could possibly imagine."* (Dr. Jonathan Kruger, personal communication with Dr. Ernest Fokoue, November 24, 2018).

- *"Your music idea is fabulous but are you sure that nothing exists? Do you know 'band in a box'? It is a software in which you put a sequence of chords and you get an improvisation 'à la manière de'. You choose amongst many musicians so they probably have the dictionary to play as Miles, Coltrane, Herbie, etc."* (Dr. Evans Gouno, personal communication with Dr. Ernest Fokoue, November 05, 2018).
- *Rebecca Ann Finnangan Kemp mentioned building blocks of music when it comes to music words idea.* (personal communication with Dr. Ernest Fokoue, November 20, 2018).

So the concept of *notes* is equivalent to *alphabet*, which can be extended as below:

- literature word $\equiv$ mixture of the 26 alphabets
- music word $\equiv$ mixture of the 12 musical notes

Since notes are fundamental, one can reasonably consider input space directly isomorphic to the 12 notes.

Two types of dictionaries are crated for the study of music genres and musicians. One is note-based represented data, another is measure-based represented data. There are 7 Main musicians we focused to study: Duke Ellington, Miles Davis, John Coltrane, Charlie Parker, Louis Armstrong, Bill Evans, Thelonious Monk. There are also three different genres of music and compare them with Jazz respectively. I select songs from China, Japan and Arab due to their unique cultural characteristics.

1.3 Organization

This thesis creates two representations of music piece as "music words" or "muselets", and applies them to topic modeling and pattern recognition methods. The naive dictionary representation is homomorphism of musical arts based on literature arts. Chapter 1 sheds light on the idea of "building blocks of music" and introduces the whole scope of the work in this thesis. Chapter 2 reviews the relevant work in text mining and music mining. Specifically, for text mining section, it demonstrates two most common pattern recognition applications, digit recognition and speech recognition. Then it concentrates on topic models, with two examples using latent Dirichlet allocation model. For music mining section, it focuses on western classical music via key detection algorithm. In Chapter 3 I construct the music mining model based on the work in text mining shows in the previous Chapter. It also demonstrates the similarity and difference between two different sources in music models (sheet music and audio music). Chapter 4 develops two representations of music notes, also known as "muselets", and respectively employ these two representations into different models. Chapter 5 summarizes the whole research work and also paves road for the potential future work.

Chapter 2

Related Work

2.1 Text Analysis

Textual data containing rich resources sometimes can be very complicated. The way to efficiently extract most useful information with the minimum efforts are demanding. Text analytics (Figure 2.1), also referred as "text mining", include information retrieval, lexical analysis, pattern recognition, annotation/tagging, sentiment analysis, topic modeling techniques. Here in this section we mainly focus on pattern recognition and topic modeling, which also help us in the music analysis chapter thereafter.

FIGURE 2.1: Word Cloud Generated via `wordcloud` Package

2.1.1 Pattern Recognition

Statistical machine learning methods and techniques have been successfully applied to wide variety of important fields. In 1960s, nonparametric estimation gained its attention with the help of Tom Cover and Peter Hart, who showed the nearest neighbor with the error at most twice as often as the best possible discrimination method (Devroye, Györfi, and Lugosi, 2013).

$$\begin{aligned} L_{NN} &= \mathbb{E}\{2\eta(X)(1-\eta(X))\} \\ &= 2\mathbb{E}\{A(1-A)\} \quad (A = \min(\eta(X), 1-\eta(X))) \\ &\leq 2\mathbb{E}\{A\}\mathbb{E}\{1-A\} \quad \textit{(by Theorem B.1)} \\ &= 2L^\star(1-L^\star) \\ &\leq 2L^\star \end{aligned} \tag{2.1}$$

where $L^\star$ is the Bayes probability of error in the best rule:

$$L^\star = \mathbb{E}\{\min(\eta(X), 1 - \eta(X))\}$$

Handwritten Digit Recognition

The famous and ubiquitous technique is handwritten digit recognition. This data set is also known as `MNIST`, and is usually the first task in some Data Analytics competitions. Handwritten digit recognition captured the attention of the machine learning and neural network community for many years, and has remained a benchmark problem in the field. Below I show the example of a small sample of data using k nearest neighbors technique to detect the handwritten digits. The handwritten digits scanned from envelops by the U.S. Postal Service are normalized in 16×16 grayscale images (Le Cun et al., 1990). The label below each plot in Figure 2.2 is the test result learned from kNN algorithm.

TABLE 2.1: Confusion Matrix: KNN on MNIST Data

	Reference									
Prediction	0	1	2	3	4	5	6	7	8	9
0	355	0	6	3	0	2	0	0	5	0
1	0	255	1	0	3	1	0	1	0	0
2	2	0	183	2	1	2	1	1	1	1
3	0	0	2	154	0	4	0	1	6	0
4	0	6	1	0	182	0	2	4	1	2
5	0	0	0	5	1	145	3	0	1	0
6	0	2	0	0	2	2	164	0	0	0
7	1	1	2	0	2	0	0	139	1	4
8	0	0	3	0	1	3	0	0	148	1
9	1	0	0	2	8	1	0	1	3	169

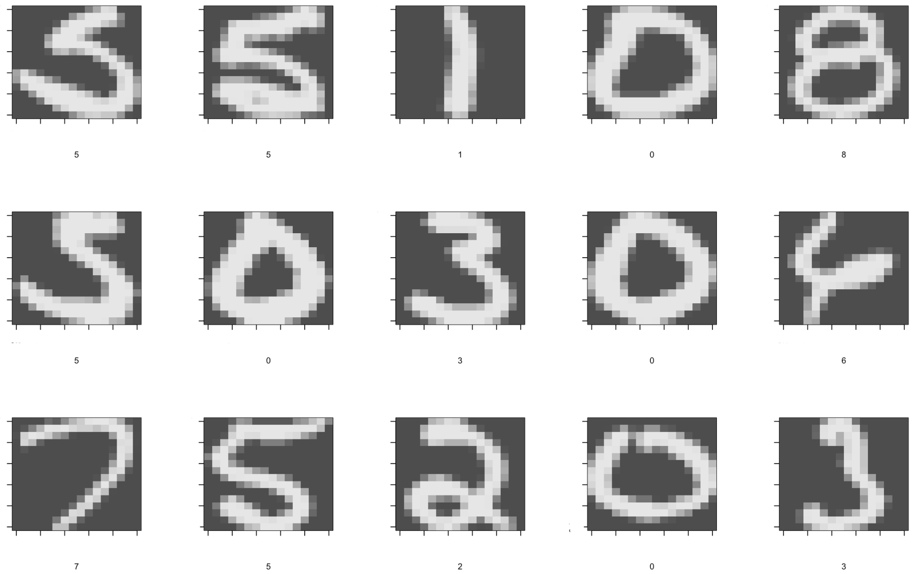


FIGURE 2.2: USPS Digit Recognition Dataset Using KNN

Speech Recognition

The triumph in speech recognition has been achieved using pattern recognition paradigms. It prevails in the world of speech recognition for utilizing terms or words as pattern and avoid the issue in phoneme level. Below is an example of transformed audio tracks of a total of 328 readings of the same English words by different speakers. Most of the readings are done by US born speakers of English while the remaining ones are done by speakers born outside the US.

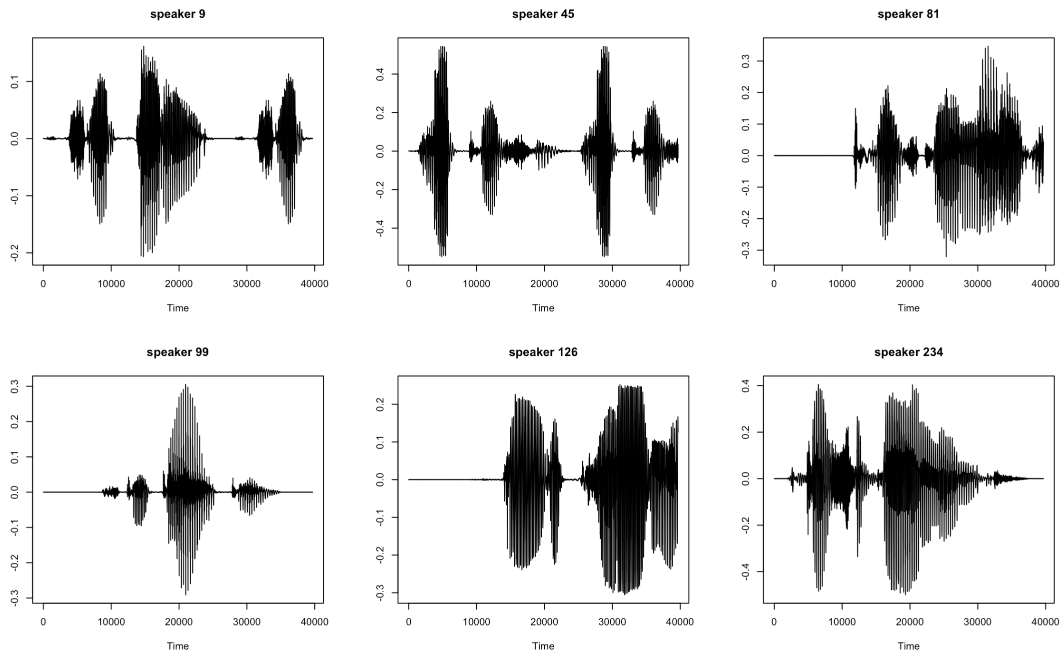


FIGURE 2.3: Audio Tracks of Selected Speakers

Consider $X_i = (x_{i1}, ..., x_{ip})^\top \in \mathbb{R}^p$ to be the time domain representation of his/her reading of an English sentence, and $Y_i \in \{1, 2, 3, 4, 5, 6\}$ is the response to distinguish the nationality of the speakers, and the set $\mathcal{D} = \{(X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n)\}$, we can detect the nationality of the speakers from their audio track pattern (Table 2.2).

TABLE 2.2: Confusion Matrix: Multi-class Support Vector Machine on Audio Track

	Reference					
Prediction	ES	FR	GE	IT	UK	US
ES	26	0	0	0	0	1
FR	0	25	0	0	0	0
GE	0	0	20	1	0	0
IT	0	0	1	24	2	1
UK	0	0	0	0	38	0
US	3	5	9	5	5	163

Using the binary classification task of US Born versus Non-US Born speakers. I compare the following methods of classification: (1) kNearest Neighbors (2) LDA (3) QDA (4) CART (5) Support Vector Machines (6) Naive Bayes in Figure 2.4. We can see different techniques have different predictive accuracy. While Naive Bayes has the largest test error, which is not surprising as it is not a robust classifier, kNN and SVM appear to be quite robust with lower test errors.

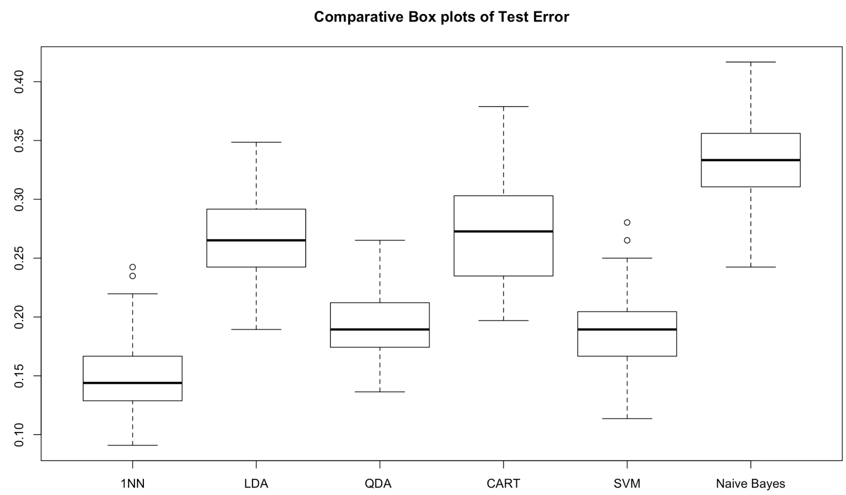


FIGURE 2.4: 6 Pattern Recognition Techniques for Audio Detection

2.1.2 Topic Modeling

Topic modeling as one of the most popular text mining techniques has been extensively studied and applied in many fields due to its intuitively easy concept of "discover hidden semantic structure in the text body". Before Latent Dirichlet Allocation

(LDA) topic model became the most common one, another model, probabilistic latent semantic analysis (PLSA), was proposed by Hofmann, 1999. This is an extension of the model latent semantic indexing (LSI), first created by Deerwester et al., 1990.

LSA and pLSA

Suppose we have D documents with N words.

$$\mathcal{D} = \{d_1, d_2, d_3, ..., d_D\}$$

$$\mathcal{W} = \{w_1, w_2, w_3, ..., w_N\}.$$

The assumption in latent semantic analysis is that words share similar meaning would appear in the same articles. So a matrix whose cell has word counts in per document is created:

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & x_{13} & \dots & x_{1D} \\ x_{21} & x_{22} & x_{23} & \dots & x_{2D} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & x_{N3} & \dots & x_{ND} \end{bmatrix}.$$

The matrix is factorized by SVD:

$$X = U\Sigma V^\top = \begin{bmatrix} \begin{bmatrix} \\ \mathbf{u}_1 \\ \\ \end{bmatrix} \dots \begin{bmatrix} \\ \mathbf{u}_l \\ \\ \end{bmatrix} \end{bmatrix} \cdot \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_l \end{bmatrix} \cdot \begin{bmatrix} [& \mathbf{v}_1 &] \\ & \vdots & \\ [& \mathbf{v}_l &] \end{bmatrix}.$$

The approximation of X in LSA is $\hat{X} = \hat{U}\hat{\Sigma}\hat{V}^\top$, and therefore it is computed by truncating the matrices. In pLSA, the approximation of X based on fixed number of topics $\mathcal{Z} = \{z_1, z_2, ..., z_K\}$ is:

$$X = P(d_i, f_j) = P(d_i|z_k)diag(P(z_k))P(f_j|z_k)^\top = \hat{U}\hat{\Sigma}\hat{V}^\top$$

Both are factorization methods with normalization while in SVD, it is the spectral norm by L_2 norm. And pLSA uses log-likelihood to maximize $\theta = (P(w_j|z_k), P(z_k|d_i))$.

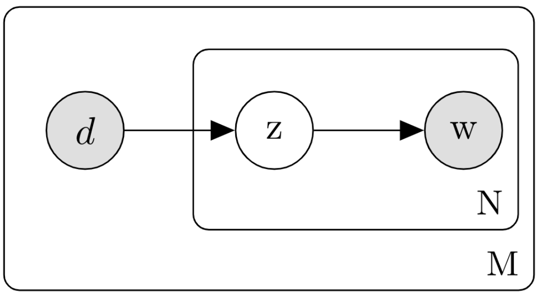


FIGURE 2.5: Graphical Model for pLSA

Figure 2.5 is the graphical model of pLSA. Nodes in the graphical model represent random variables with shaded ones refer to observed variables and blank ones refer to latent variables. Plates in the graphical model demonstrate the replicates of the process. Here we assume M number of documents in the corpus, and each document contains N words(Blei, Ng, and Jordan, 2003). The graphical model can be translated as the Equation (2.2).

$$P(d_j, w_i) = P(d_j)P(w_i|d_j) \tag{2.2}$$

$$P(w_i|d_j) = \sum_{k=1}^{K} P(z_k|d_j)P(w_i|z_k) \tag{2.3}$$

Based on the observed words and documents, we can gain the conditional probability $P(w_i|d_j)$ by marginalizing over topics. $P(z_k|d_j)$ is the probability of certain topic z_k appearing in certain document d_j, and $P(w_i|z_k)$ is the probability of the word w_i showing in a specific topic z_k. EM algorithm(Blei, Ng, and Jordan, 2003) is applied to get the optimal result.

Generative Process:

1. Determine the number of words in the documents
2. Choose a topic mixture for the document over a fixed set of topics
3. Generate the words in the document by
   (a) Picking a topic based on the document's multinomial distribution
   (b) Picking a word based on the topic's multinomial distribution

Latent Dirichlet Allocation

Because pLSA does not have a generative process to create documents from scratch, and thus could "spread out" with small weights on many topics to cause overfitting issue, LDA is proposed to avoid this situation. LDA learns the topic representation of topics in each document and the word distribution of each topic. It backtracks from the document level to identify topics that are likely to have generated the corpus.

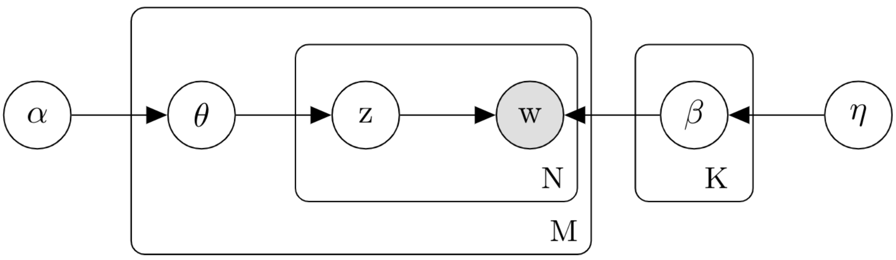


FIGURE 2.6: Graphical Model for LDA

Shaded nodes w is only observed variables in the graphical model (Figure 2.6). The model can be translated as the Equation (2.4). The posterior is intractable to compute, thus the common way to turn is to approximate the posterior via variational

EM algorithm, or Gibbs sampling.

$$P(\theta, \mathbf{z}, \mathbf{w}, \beta|\alpha, \eta) = \prod_{k=1}^{K} p(\beta_k|\eta) \prod_{d=1}^{D} p(\theta_d|\alpha) \left(\prod_{n=1}^{N} p(z_{d,n}|\theta_d) p(w_{d,n}|\beta_{1:K}, z_{d,n}) \right) \tag{2.4}$$

The topic distribution under each document is a Multinomial distribution $Mult(\theta)$ with its conjugate prior $Dir(\alpha)$. The word distribution under each topic is a Multinomial distribution $Mult(\beta)$ with its conjugate prior $Dir(\eta)$. For the n^{th} word in the certain document, first we select a topic z from from per document-topic distribution $Mult(\theta)$, then select a word under this topic $w|z$ from per topic-word distribution $Mult(\beta)$. Repeat for M documents: For M documents, there are M independent Dirichlet-Multinomial Distributions; for K topics, there are K independent Dirichlet-Multinomial Distributions.

Generative Process:

1. Randomly assign each word in each document to one of the K topics
2. For each document d
   (a) Assume all topic assignments except for the current one are correct
   (b) Calculate two proportions:
      i. Proportion of words in document d that are currently assigned to topic z: $P(\text{topic } z|\text{document } d)$
      ii. Proportion of assignments to topic k over all documents that come from this word w: $P(\text{word } w|\text{topic } z)$
      iii. Multiply the two proportions and assign a new topic based on the probability: $P(\text{word } w|\text{topic } z) \times P(\text{topic } z|\text{document } d)$
3. Until we reach a steady state

LDA Implementation: Short Story Analysis

Here I borrow the example from Silge (2018) to show how LDA topic model works in text analysis. The short stories is collected from `gutenbergr` package. After manipulating the raw text data by removing the stop words, indicating important words, we get the Figures 2.7. Individual story emphasis on different narrative elements and words. Some stories contain a lot of animals while others contain many family names.
After the data has been cleaned and Document-Term matrix (DTM) is created, we can feed it into the topic model. Figure 2.8 demonstrates the words with highest probabilities in each topic. Different topic is mixture of different words. For topic 4, it focuses on family relationship. For topic 2, it probably tells story about birds.
Figure 2.9 shows the document probabilities for each topic. We can see each topic is related to 1∼3 stories. We can also find that each short story only has one topic, which not commonly happen in text mining. Because in this scenario we have small number of documents with relatively large number of topics corresponding to the documents.

SLDA Implementation: Political Blog Post Analysis

Supervised LDA is an extension of the general LDA topic models (Mcauliffe and Blei, 2008). It enriches the model by associating each document with a label. The

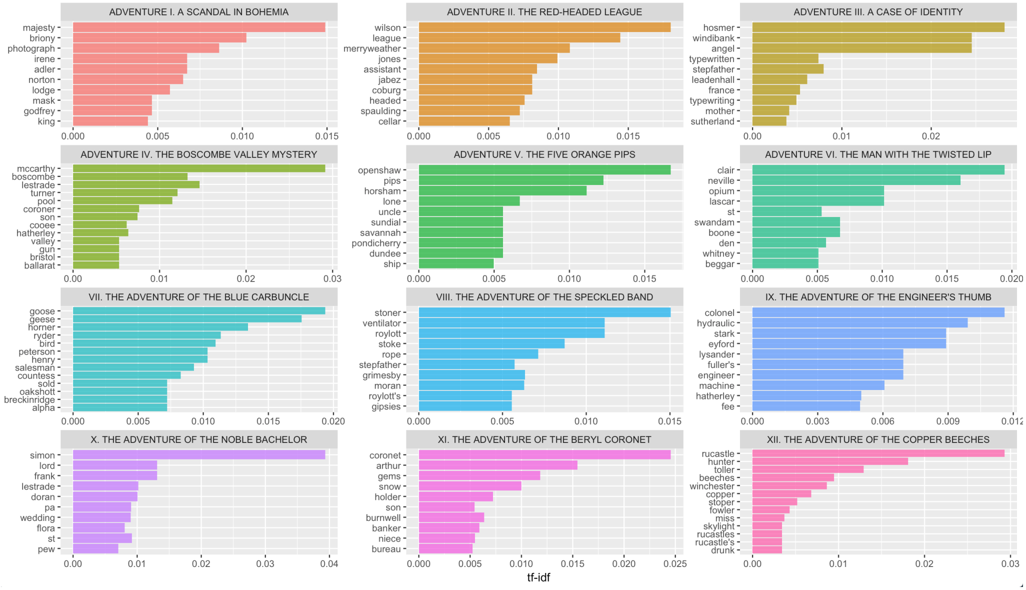


FIGURE 2.7: Highest TF-IDF Words in the Stories

response is usually the rating score(for movies or books), or the count (for websites or blogs views).

Generative Process

1. Draw topic proportions $(\theta|\alpha)$ from $Dir(\alpha)$
2. For each word
   (a) Draw topic assignments $(z|\theta)$ from $Mult(\theta)$
   (b) Draw word $(w|z, \beta_{1:K})$ from $Mult(\beta_z)$
3. Draw response variable $(y|z_{1:N}, \eta, \sigma^2)$ from $N(\eta^\top \bar{z}, \sigma^2)$

In this case I analysis 273 US political blogs with 71,654 blog posts ranked by Technorati score for the whole 2012 year. The higher Technorati score is, the more influential the blog post is, and more people would read the posts. The score ranked from 83 to 876, with the most frequent score 127 containing 8,135 blog posts through 366 days. We are going to predict the Technorati score from the topic proportions and *log(number of posts)* over the entire year.

I divided the total 71,654 blog posts into 10 categories with equal number of posts in each category, and labeled them from 0 to 9 consistently.

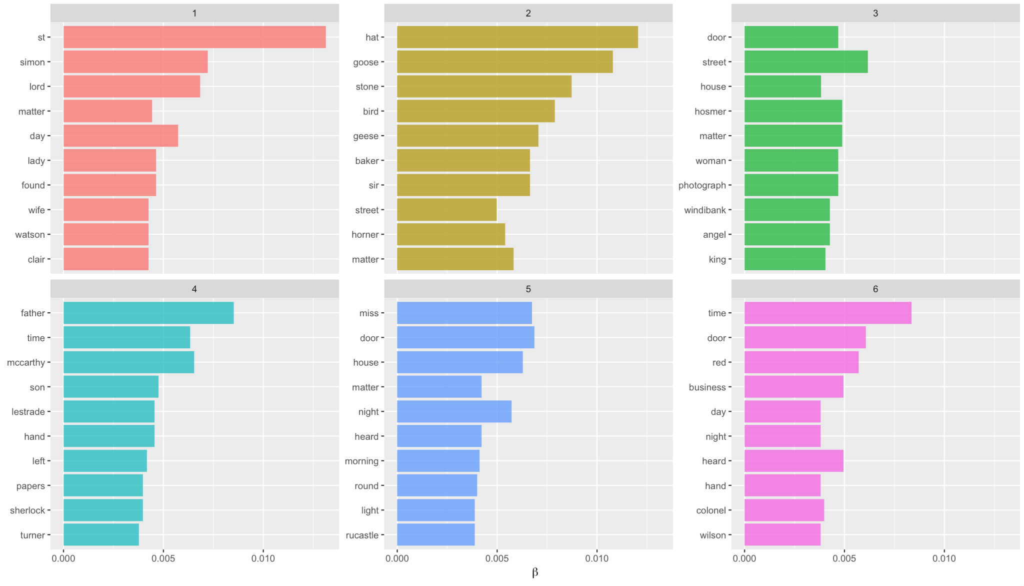


FIGURE 2.8: Highest Word Probabilities for Each Topic

TABLE 2.3: 10 Categories for Blog Posts

Score	83∼95	96∼110	111∼126	127∼444	445∼466
Annotation	0	1	2	3	4
Score	467∼553	554∼624	625∼657	658∼687	688∼876
Annotation	5	6	7	8	9

From the topic-document distribution in Figure 2.11 we can notice that some document such as *Doc 2* focus on Topic *Economics* with highly probable words such as "gold", "market", "economic". While other documents contain mixture of several topics such as *Doc 3*, *Doc 8*, *Doc 9*.

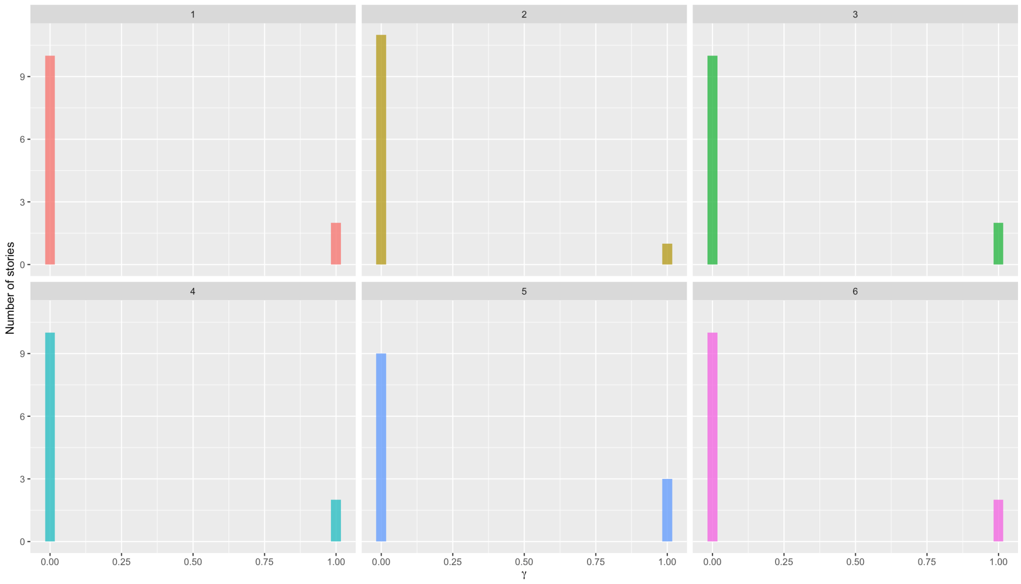


FIGURE 2.9: Distribution of Document Probabilities for Each Topic

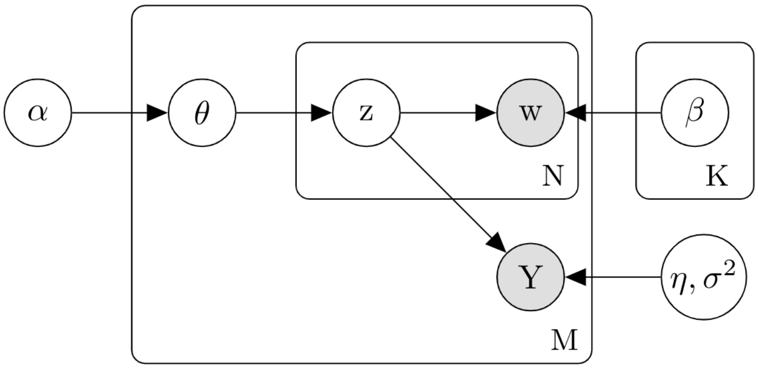


FIGURE 2.10: Graphical Model for SLDA

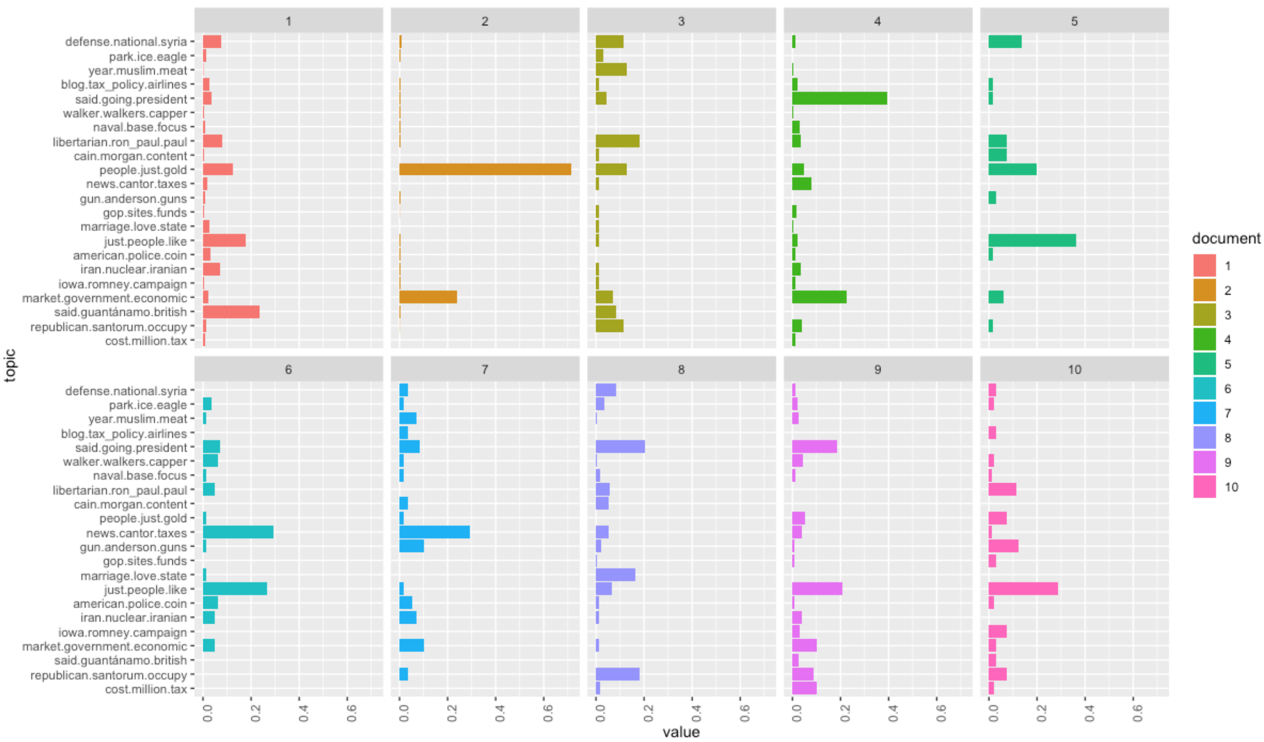


FIGURE 2.11: Topic Document Distribution

Given a particular blog *https://thinkprogress.org*, Technorati score for the blog *https://thinkprogress.org* is 127, which falls into the 3rd category.

TABLE 2.4: Particular Blog and its Score

Score	127∼444
Annotation	3

From sLDA model, I get the predicted annotation with the highest probability falling into Category 3, consistent with the label.

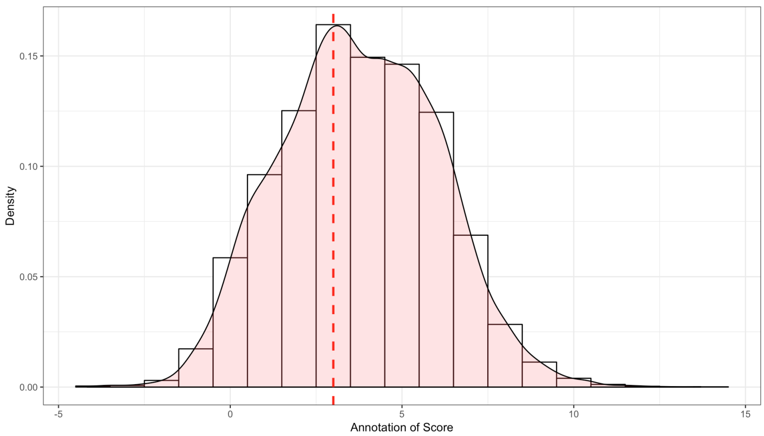


FIGURE 2.12: Prediction for blog *thinkprogress.org*

2.2 Music Analysis

Extensive studies have been conducted on both musical scores and audio tracks of western classical music with the finality of learning and detecting the key in which a particular piece of music was played. Both the Bayesian Approach and modern unsupervised learning via latent Dirichlet allocation have been used for such learning tasks. In this section I will give brief introduction to the relevant music mining techniques developed in recent years.

2.2.1 Topic Models

Probabilistic topic model has been employed in many fields. Hu (2009) shows in her paper using Latent Dirichlet Allocation in text, image and music. In music part she mainly focus on western classical music due to its clear and mature formation in music theory. She applied LDA to classical symbolic music for automatic harmonic analysis. Her work goes beyond bag-of-words representation and discard the order of notes in each segment with the idea of "bag of segments" where each segment is treated as "bag of notes". Figure 2.13 shows key judgments for *Bach's Prelude in C minor, WTC-II*. The top three keys in each measure segment are the judgments from LDA model, the bottom three keys are judgments from human experts.

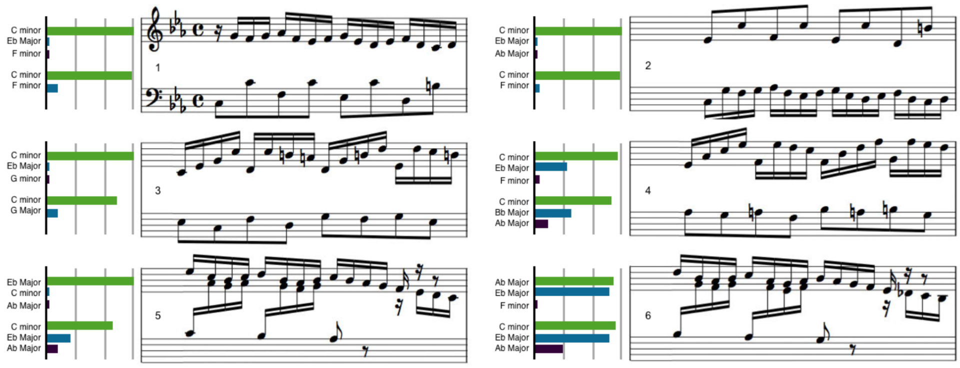


FIGURE 2.13: Note. Reprinted from "A probabilistic topic model for music analysis", by Hu, Diane J and Lawrence K Saul. , (2009).

2.2.2 Other Key-finding Algorithms

Except topic modeling approach, decades ago another method proposed by Krumhansl and Kessler (1982) in key detection is very influential. They used "flat" major profile by removing all keys that were not in the current melody. E.g. C major key-profile:

$$[1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1]$$

The key-profile they created were based on the experimental data. They conducted a series of experiments with listener hearing incomplete melody in separate trails. Later Krumhansl & Schmuckler created well-know KS key-finding algorithm in 1990 based on the empirical work from Krumhansl & Kessler (1982).

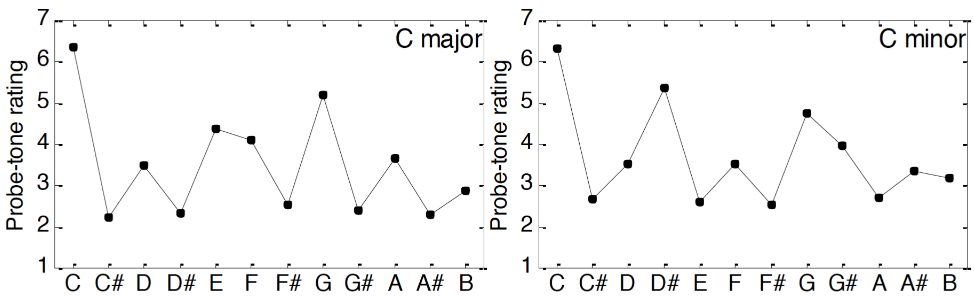


FIGURE 2.14: Note. Reprinted from "MIDI toolbox: MATLAB tools for music research", by Eerola, T. and Toiviainen, P. , (2004).

Figure 2.14 shows the probability of the tone for C major and C minor keys from (Krumhansl and Kessler, 1982). The approach has been examined and reached 83% accuracy rate on 48 preludes from Bach, 70% accuracy rate on Shostakovich's preludes.

Chapter 3

Music Mining

Music and text are similar in the way that both of them can be regraded as information carrier and emotion deliverer. People get daily information from reading newspaper, magazines, blogs etc., and they can also write diary or personal journal to reflect on daily life, let out pent up emotions, record ideas and experience. Same power could come from music! Composers express their feelings through music with different combinations of notes, diverse tempo[1], and dynamics levels[2], as another version of language. All these similarities drive people to ask questions like:

- Could music deliver information tantamount to text?
- Can we efficiently use text mining approach in music field?
- Why music from diverse culture can bring people so many different feelings?
- What's the similarity between music from different culture, or composers, or genres?
- To what extend do people grasp the meaning behind each piece of music expressed by the composer?

And more and more, just name a few. After all, the power of music and the meaning behind it have puzzled scientists for long time, though some relative researches has been studied, in comparatively low frequencies. Furthermore, the process of deeply digging into the music structure and decompose it appear to be tabu for many people, especially music enthusiasts who regard the natural integral attribution of music as sacred and inviolable. I personally encountered the difficulty during this research as one of my friends commented that *"Deciphering music in a mathematical way seems intriguing, but to me it is cruel as music itself embodies intuitively mysterious beauty."* I admit his philosophy point that "distance creates beauty", while we could not ignore the fact that the modern advancing techniques attract more and more researchers tend to study the complex system behind intuition, especially the fast-pacing development in Neuroscience recent decades, avails people to find the answer about how music stimulates our brain to reflect mixture of emotional and intellectual reaction.

This chapter is dedicated to using text mining tool in music field, specifically, applying Topic Modeling to Improvisational Jazz Music and other Music genres such as Chinese music and Japanese Music.

[1]In musical terminology, tempo ("time" in Italian), is the speed of pace of a given piece.

[2]In music, dynamics means how loud or quiet the music is.

3.1 Intuition Behind Model

Similar to the work from Blei (2012) in text mining, Figure 3.1 illustrates the intuition behind our model in music concept. We assume an album, as a collection of songs, are mixture of different topics (melodies). These topics are the distributions over a series of notes (left part of the figure). In each song, notes in every measure are chosen based on the topic assignments (colorful tokens), while the topic assignments are drawn from the document-topic distribution.

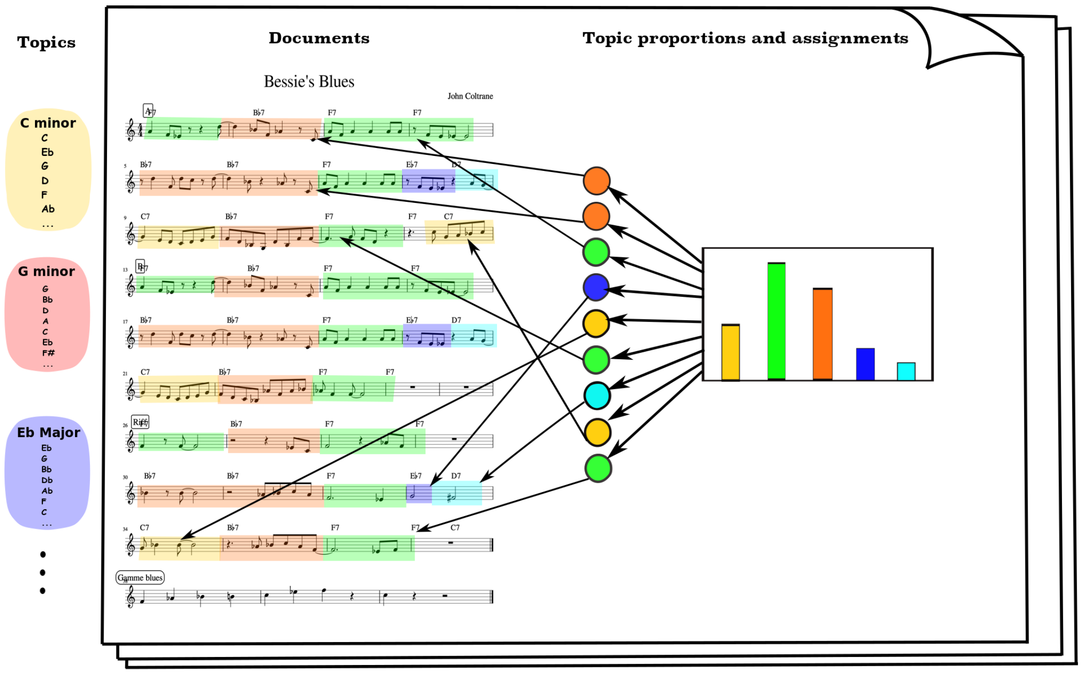
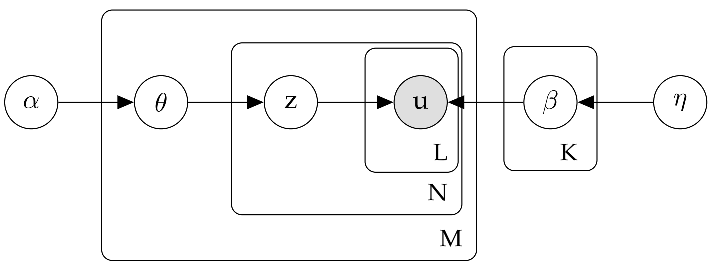


FIGURE 3.1: Intuition behind Music Mining

3.2 LDA for Sheet Music

In this section, I'll show the generative process of sheet music based on the graphical model as well as the corresponding computation.

3.2.1 Model



$$\textbf{Dirichlet:} \quad p(\theta|\alpha) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} \qquad p(\beta|\eta) = \frac{\Gamma(\sum_i \eta_i)}{\prod_i \Gamma(\eta_i)} \prod_{i=1}^{K} \theta_i^{\eta_i - 1} \tag{3.1}$$

$$\textbf{Multinomial:} \quad p(z_n|\theta) = \prod_{i=1}^{K} \theta_i^{z_n^i} \qquad p(x_n|z_n, \beta) = \prod_{i=1}^{K} \prod_{j=1}^{V} \beta_{ij}^{(z_n^i x_n^j)} \tag{3.2}$$

Notation

- u: notes (observed)
- z: chord per measure (hidden)
- θ chord proportions for a song (hidden)
- α: parameter controls chord proportions
- β: key profiles
- η: parameter controls key profiles

3.2.2 Generative Process

1. Draw $\theta \sim \text{Dirichlet}(\alpha)$
2. For each harmony $k \in \{1, ..., K\}$
   - Draw $\beta_k \sim \text{Dirichlet}(\eta)$
3. For each measure $\mathbf{u}_n$ (notes in nth measure) in song m
   - Draw harmony $z_n \sim \text{Multinomial}(\theta)$
   - Draw pitch in nth measure $x_n|z_n \sim \text{Multinomial}(\beta_k)$

Terms for single song:

$$p(\theta|\alpha) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} \tag{3.3}$$

$$p(\beta|\eta) = \frac{\Gamma(\sum_i \eta_i)}{\prod_i \Gamma(\eta_i)} \prod_{i=1}^{K} \theta_i^{\eta_i - 1} \tag{3.4}$$

$$p(z_n|\theta) = \prod_{i=1}^{K} \theta_i^{z_n^i} \tag{3.5}$$

$$p(x_n|z_n, \beta) = \prod_{i=1}^{K} \prod_{j=1}^{V} \beta_{ij}^{(z_n^i x_n^j)} \tag{3.6}$$

Joint Distribution for the whole album:

$$p(\theta, \mathbf{z}, \mathbf{x}|\alpha, \beta, \eta) = \prod_{k=1}^{K} p(\beta|\eta) \prod_{m=1}^{M} p(\theta|\alpha) \Big(\prod_{n=1}^{N} p(z_n|\theta) p(x_n|z_n, \beta) \Big) \tag{3.7}$$

Summary

- Assume there are M documents in the corpus.
- The topic distribution under each document is a Multinomial distribution $Mult(\theta)$ with its conjugate prior $Dir(\alpha)$.
- The word distribution under each topic is a Multinomial distribution $Mult(\beta)$ with the conjugate prior $Dir(\eta)$.
- For the n^{th} word in the certain document, first we select a topic z from per document-topic distribution $Mult(\theta)$, then select a word under this topic $x|z$ from per topic-word distribution $Mult(\beta)$.
- Repeat for M documents. For M documents, there are M independent Dirichlet-Multinomial Distributions; for K topics, there are K independent Dirichlet-Multinomial Distributions.

3.2.3 Estimation

For per-document posterior is

$$p(\beta, \mathbf{z}, \theta|\mathbf{x}, \alpha, \eta) = \frac{p(\theta, \beta, \mathbf{z}, \mathbf{x}|\alpha, \eta)}{p(\mathbf{x}|\alpha, \eta)} = \frac{p(\theta|\alpha)\prod_{n=1}^{N} p(z_n|\theta)p(x_n|z_n, \beta_{1:K})}{\int_\theta p(\theta|\alpha)\prod_{n=1}^{N}\sum_{z=1}^{K} p(z_n|\theta)p(x_n|z_n, \beta_{1:K})} \tag{3.8}$$

Here we use Variational EM (VEM) **??** instead of EM algorithm to approximate posterior inference because the posterior in E-step is intractable to compute.

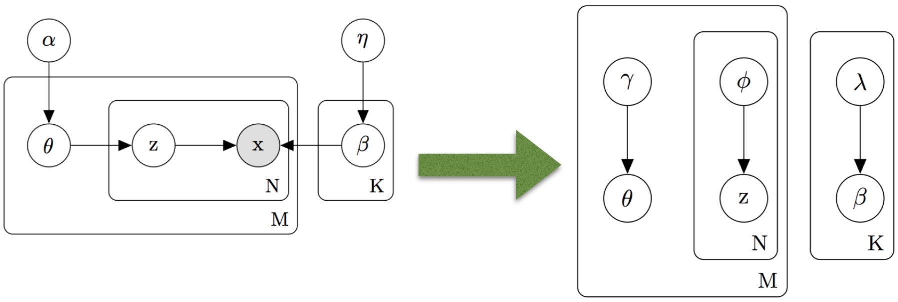


FIGURE 3.2: Variational EM Graphical Model

Blei, Ng, and Jordan (2003) proposed a way to use variational term $q(\beta, \mathbf{z}, \theta|\lambda, \phi, \gamma)$ (Eq.3.9) to approximate the posterior $p(\beta, \mathbf{z}, \theta|\mathbf{x}, \alpha, \eta)$ (Eq.3.10). That is to say, by removing certain connections in the graphical model in Figure 3.2, we obtain the tractable version of lower bounds on the log likelihood.

$$q(\beta, \mathbf{z}, \theta|\lambda, \phi, \gamma) = \sum_{k=1}^{K} \mathrm{Dir}(\beta_k|\lambda_k)\sum_{d=1}^{M}(q(\theta_d|\gamma_d)\sum_{n=1}^{N} q(z_{dn}|\phi_{dn})) \tag{3.9}$$

$$p(\beta, \mathbf{z}, \theta|\mathbf{x}, \alpha, \eta) = \frac{p(\theta, \beta, \mathbf{z}, \mathbf{x}|\alpha, \eta)}{p(\mathbf{x}|\alpha, \eta)} \tag{3.10}$$

With the simplified version of posterior distribution, we aim to minimize the KL Distance (Kullback–Leibler divergence) between the variational distribution $q(\beta, z, \theta|\lambda, \phi, \gamma)$

and the posterior $p(\beta, z, \theta|x, \alpha, \eta)$ to obtain the optimal value of the variational parameters γ, ϕ, and λ (Eq.3.12). That is to obtain the maximum lower bound $L(\gamma, \phi, \lambda; \alpha, \eta)$ (Eq.3.13).

$$\ln p(\mathbf{x}|\alpha, \eta) = L(\gamma, \phi, \lambda; \alpha, \eta) + D(q(\beta, \mathbf{z}, \theta|\lambda, \phi, \gamma)||p(\beta, \mathbf{z}, \theta|\mathbf{x}, \alpha, \eta)) \tag{3.11}$$

$$(\lambda^*, \phi^*, \gamma^*) = \underset{\lambda,\phi,\gamma}{\texttt{argmin}} D(q(\beta, \mathbf{z}, \theta|\lambda, \phi, \gamma)||p(\beta, \mathbf{z}, \theta|\mathbf{x}, \alpha, \eta)) \tag{3.12}$$

$$\begin{aligned} L(\gamma, \phi, \lambda; \alpha, \eta) = {} & E_q[\ln p(\theta|\alpha)] + E_q[\ln p(\mathbf{z}|\theta)] + E_q[\ln p(\beta|\eta)] + E_q[\ln p(\mathbf{x}|\mathbf{z}, \beta)] \\ & - E_z[\ln q(\theta|\gamma)] - E_q[\ln q(\mathbf{z}|\phi)] - E_z[\ln q(\beta|\lambda)] \end{aligned} \tag{3.13}$$

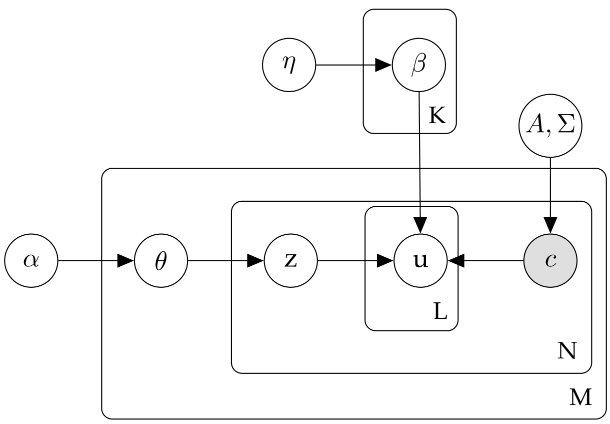
Algorithm 1 Variational EM for Smoothed LDA in Sheet Music

for $t \leftarrow 1 : T$ **do**
  E-step
  Fix model parameters α, η. Initialize $\phi^0_{ni} := \frac{1}{k}, \gamma^0_i := \alpha_i + \frac{N}{k}, \lambda^0_{ij} := \eta$
  for $n \leftarrow 1 : N$ **do**
    for $i \leftarrow 1 : k$ **do**
      $\phi^{t+1}_{ni} := \exp(\Psi(\gamma^t_i)) \prod_{j=1}^{V} \beta_{ij}^{x^j_n}$
    end for
    Normalize ϕ^{t+1}_n to sum to 1
  end for
  $\gamma^{t+1} := \alpha + \sum_{n=1}^{N} \phi^{t+1}_n$
  $\lambda^{t+1}_j := \eta + \sum_{d=1}^{M} \sum_{n=1}^{N_d} \phi^{t+1}_{dn} x^j_{dn}$
  M-step
  Fix the variational parameters γ, ϕ, λ
  Maximize lower bound with respect to model parameters η, α
  until converge
end for

3.3 LDA for Audio Music

In this section, I'll show the generative process of audio music based on the graphical model as well as the corresponding computation.

3.3.1 Model



Draw chroma-vector c_n from probability distribution (Hu and Saul, 2009b):

$$p(c_n|\mathbf{u}_n, A) = \frac{1}{\sqrt{(2\pi)^p|\Sigma|}} exp\left\{-\frac{1}{2}(c_n - A\mathbf{u}_n)^\top \sigma^{-1}(c_n - A\mathbf{u}_n)\right\} \tag{3.14}$$

Notation

- u: notes (observed)
- z: chord per measure (hidden)
- θ chord proportions for a song (hidden)
- α: parameter controls chord proportions
- β: key profiles
- c: chroma feature for certain time period
- A: $V \times V$ matrix
- Σ: covariance matrix

3.3.2 Generative Process

1. Draw $\theta \sim$ Dirichlet(α)
2. For each harmony $k \in \{1, ..., K\}$
   - Draw $\beta_k \sim$ Dirichlet(η)
3. For each measure $\mathbf{u}_n$ (notes in nth measure) in song m
   - Draw harmony $z_n \sim$ Multinomial(θ)
   - Draw pitch in nth measure $x_n|z_n \sim$ Multinomial(β_k)
   - Draw chroma vector $c_n \sim N(A\mathbf{u}_n, \Sigma)$ to infer the hidden notes

Terms for single song:

$$p(\theta|\alpha) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} \tag{3.15}$$

$$p(\beta|\eta) = \frac{\Gamma(\sum_i \eta_i)}{\prod_i \Gamma(\eta_i)} \prod_{i=1}^{K} \theta_i^{\eta_i - 1} \tag{3.16}$$

$$p(z_n|\theta) = \prod_{i=1}^{K} \theta_i^{z_n^i} \tag{3.17}$$

$$p(x_n|z_n, \beta) = \prod_{i=1}^{K} \prod_{j=1}^{V} \beta_{ij}^{(z_n^i x_n^j)} \tag{3.18}$$

$$p(c_n|\mathbf{u}_n, A) = \frac{1}{\sqrt{(2\pi)^p|\Sigma|}} exp\left\{-\frac{1}{2}(c_n - A\mathbf{u}_n)^\top \sigma^{-1}(c_n - A\mathbf{u}_n)\right\} \tag{3.19}$$

Joint Distribution for the whole album:

$$p(\theta, \mathbf{z}, \mathbf{x}, \beta, \mathbf{c}|\alpha, \eta, A) = \prod_{k=1}^{K} p(\beta|\eta) \prod_{m=1}^{M} p(\theta|\alpha)\Big(\prod_{n=1}^{N} p(z_n|\theta)p(x_n|z_n, \beta)p(c_n|x_n, A)\Big) \tag{3.20}$$

Summary

The input of the song in audio music is not notes but chroma feature $c \in \mathcal{R}^{12}$. So in audio music the notes are now latent variables and only the chroma vector gets observed. The extra step in generative process is to have chroma vector c_n drawn from probability distribution with additional parameters A and Σ to learn (Eq.3.14).

3.3.3 Estimation

Again here we use Variational Bayes to approximate the intractable posterior. To minimize the KL Divergence between the approximate posterior and the true probability (Eq.3.22), we need to maximize the lower bound $L(\gamma, \phi, \lambda, \omega; \alpha, \eta, A)$ (Eq.3.23).

$$\ln p(\mathbf{c}|\alpha, \eta, A) = L(\gamma, \phi, \lambda, \omega; \alpha, \eta, A) + D(q(\beta, \theta, \mathbf{z}, \mathbf{x}|\lambda, \phi, \gamma, \omega)||p(\beta, \theta, \mathbf{z}, \mathbf{x}|\mathbf{x}, \alpha, \eta, A)) \tag{3.21}$$

$$(\lambda^*, \phi^*, \gamma^*, \omega^*) = \underset{\lambda,\phi,\gamma}{\texttt{argmin}} D(q(\beta, \theta, \mathbf{z}, \mathbf{x}|\lambda, \phi, \gamma, \omega)||p(\beta, \theta, \mathbf{z}, \mathbf{x}|\mathbf{x}, \alpha, \eta, A)) \tag{3.22}$$

$$\begin{aligned} L(\gamma, \phi, \lambda, \omega; \alpha, \eta, A) = {} & E_q[\ln p(\theta|\alpha)] + E_q[\ln p(\mathbf{z}|\theta)] + E_q[\ln p(\beta|\eta)] \\ & + E_q[\ln p(\mathbf{c}|\mathbf{x}, A)] + E_q[\ln p(\mathbf{x}|\mathbf{z}, \beta)] - E_z[\ln q(\theta|\gamma)] \\ & - E_q[\ln q(\mathbf{z}|\phi)] - E_z[\ln q(\beta|\lambda)] - E_z[\ln q(\mathbf{x}|\omega)] \end{aligned} \tag{3.23}$$

Notice here x_n as binary vector indicating if certain pitch among the 12 pitches is available in nth measure, we can get the variational term for x_n:

$$p(x_n|w_n) = \prod_{j=1}^{V} w_{jn}^{x_n^j}(1 - w_{jn})^{1-x_n^j} \tag{3.24}$$

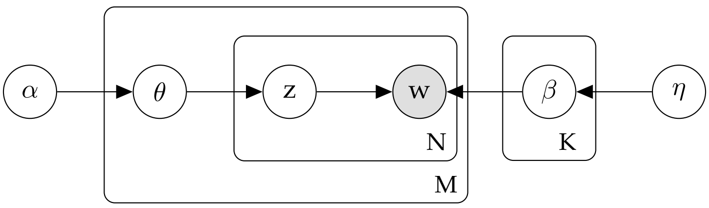
Algorithm 2 Variational EM for Smoothed LDA in Audio Music

for $t \leftarrow 1 : T$ **do**
  E-step
  Fix model parameters α, η, A.
  Initialize $\phi^0_{ni} := \frac{1}{k}$, $\gamma^0_i := \alpha_i + \frac{N}{k}$, $\lambda^0_{ij} := \eta$, $\omega^0_{jn} := z$
  for $n \leftarrow 1 : N$ **do**
    for $i \leftarrow 1 : k$ **do**
      $\phi^{t+1}_{ni} := \exp(\Psi(\gamma^t_i)) \prod_{j=1}^{V} \beta_{ij}^{\omega^j_n}$
    end for
    Normalize ϕ^{t+1}_n to sum to 1
  end for
  $\gamma^{t+1} := \alpha + \sum_{n=1}^{N} \phi^{t+1}_n$
  $\lambda_j := \eta + \sum_{d=1}^{M} \sum_{n=1}^{N_d} \phi^{t+1}_{dn} \omega^j_{dn}$
  $w_{jn} := c_{jn} \delta a - \frac{1}{2} a^2 \delta + z$
  M-step
  Fix the variational parameters γ, ϕ, λ, ω
  Maximize lower bound with respect to model parameters A, η, α
  until converge
end for

3.4 Model Comparison

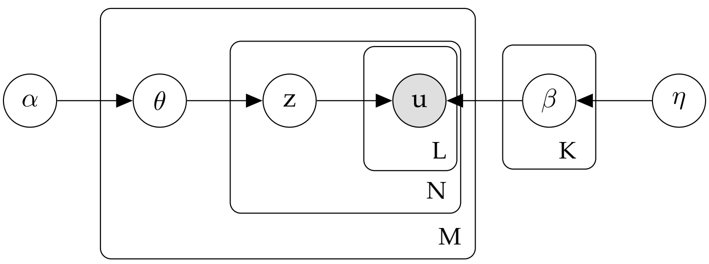
3.4.1 Text Mining vs. Music Mining

Text Mining:



$$p(\theta, \mathbf{z}, \mathbf{w} | \alpha, \beta, \eta) = \prod_{k=1}^{K} p(\beta|\eta) \prod_{m=1}^{M} p(\theta|\alpha) \Big(\prod_{n=1}^{N} p(z_n|\theta) p(w_n|z_n, \beta) \Big)$$

Music Mining:



$$p(\theta, \mathbf{z}, \mathbf{x}|\alpha, \beta, \eta) = \prod_{k=1}^{K} p(\beta|\eta) \prod_{m=1}^{M} p(\theta|\alpha) \Big(\prod_{n=1}^{N} p(z_n|\theta) p(x_n|z_n, \beta) \Big)$$

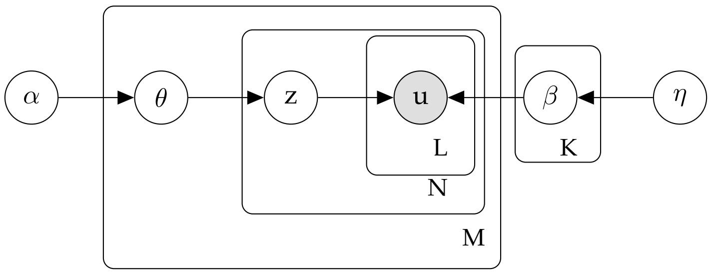
where

- x_n is a $V \times 1$ indicator vector for a series of notes from a certain pitch $\in \{A, A\sharp, B, ..., G\sharp\}$ among **12** in nth measure
- $z_n \in$ {A major, F minor,...,E♭ major} is a scalar given **24** key-profiles where $z_n^i = 1$ for a specific i.

The difference between Text Mining and Music Mining (for sheet music) is that in music model, we have one more plate on node "notes". We regard whole notes in one measure as one "term", so we have L number of notes in one measure, which can be regarded as the length of each "term". Due to the equal duration in music measure, we have terms with the same number of notes in Music Mining.

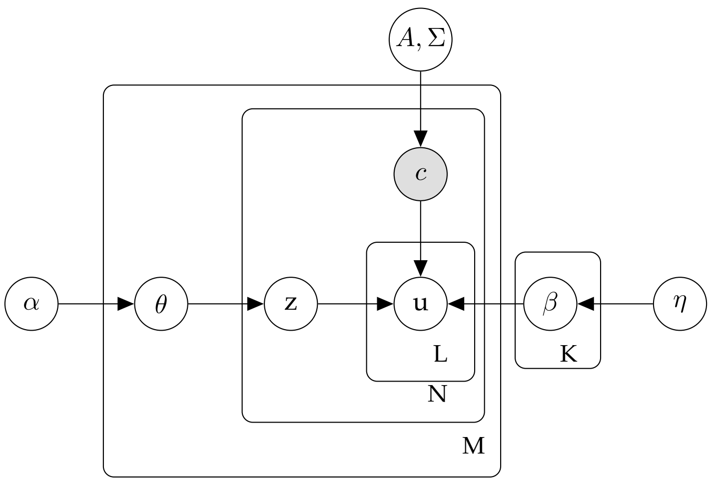
3.4.2 Sheet Music vs. Audio Music

Sheet Music



$$p(\theta, \mathbf{z}, \mathbf{x}|\alpha, \beta, \eta) = \prod_{k=1}^{K} p(\beta|\eta) \prod_{m=1}^{M} p(\theta|\alpha) \Big(\prod_{n=1}^{N} p(z_n|\theta) p(x_n|z_n, \beta) \Big)$$

Audio Music



$$p(\theta, \mathbf{z}, \mathbf{x}, \beta, \mathbf{c}|\alpha, \eta, A) = \prod_{k=1}^{K} p(\beta|\eta) \prod_{m=1}^{M} p(\theta|\alpha) \Big(\prod_{n=1}^{N} p(z_n|\theta) p(x_n|z_n, \beta) p(c_n|\mathbf{u}_n, A) \Big)$$

The observed notes in Sheet Music become hidden variables in Audio Music, so there is one more step in Audio Music compared with Sheet Music. That is to draw chroma vector from Gaussian distribution to infer the hidden notes.

Chapter 4

Application

4.1 Improvisational Learning

Extensive studies have been conducted on both musical scores and audio tracks of western classical music with the finality of learning and detecting the key in which a particular piece of music was played. Both the Bayesian Approach and modern unsupervised learning via latent Dirichlet allocation have been used for such learning tasks. In this research work, we venture out of the western classical genre and embrace and explore jazz music. We consider the musical score sheets and audio tracks of some of the giants of jazz like Duke Ellington, Miles Davis, John Coltrane, Dizzie Gillespie, Wes Montgomery, Charlie Parker, Sonny Rollins, Louis Armstrong (Instrumental), Bill Evans, Dave Brubeck, Thelonious Monk (Pianist). We specifically employ Bayesian techniques and modern topic modelling methods (and even occasionally a combination of both) to explore tasks such as: automatic improvisation detection, genre identification, key learning (how many keys do the giants of jazz tended to play in, and what are those keys) and even elements of the mood of the piece.

4.1.1 Why Jazz

Classical Music is one of the music genres that have been heavily studied due to its stability and complete notation system[1], which leaves less room for composers to improvise like other non-European music and popular music. Jazz as one of popular music style originated from African-American in the late 19th century is usually regarded as "America's classical music". We select Jazz mainly based on its unique traits listed below:

- Never the same, creative and innovative.
- Jazz is pretty improvisational and solo based.
- Jazz is flexible, though, still follow the music theory.
- Many variations in each chord, though not Jazz specifically.

We are intended to study both sheet music and audio music to track the improvisational part of Jazz. For any given song, we would assume the printed form and audio form to be consistent. Therefore any inconsistent part between the two forms would be regarded as solo part.

[1] Western notation is created to indicate the pitches, tempo, metre and rhythms for a piece of music

4.1.2 Data Preprocessing

There are two types of data format in our study: `mxl` file for sheet music, `mid` file for audio music. Both data are collected from MuseScore[2] containing music pieces from Duke Ellington, Miles Davis, John Coltrane, Charlie Parker, Louis Armstrong, Bill Evans, Thelonious Monk. These musicians were mainly active in Jazz music during last century (1900 $\sim$ 1999).

Sheet Music

- Transfer `mxl` file to `xml` file
- Use `xml` files to extract notes in each measure
- Create matrices based on the extracted notes (Appendix A)

	V1	V2	V3	V4	V5	V6	V7	V8
1	F	F	C	A	E	E	C	A
2	D	E	B	C	C sharp	B flat	G	G sharp
3	A	A	F	D	G	A	F	E
4	D	D flat	F	O	O	O	O	O
5	C	C	B flat	F	A flat	B flat	G	O
6	E flat	D flat	A flat	F	C	F	G	A
7	A	A	E	C	D	D	D flat	O
8	D	D	C flat	G flat	B flat	B flat	A flat	O
9	G	F	F	F	D	B flat	G	O
10	A	G	C	B flat	E flat	E flat	C	O
11	C	C	A	F	G	G	D	O
12	D	D	B flat	F	A	A	O	O

FIGURE 4.1: Transforming Notes from Music Sheets to Matrices

Based on the concept of duration (the length of time a pitch/ tone is sounded), and in each measure the duration is fixed, we can create Measure-Note matrices. In Measure-Note matrices, we use letter {C, D, E, F, G, A, B} to denote the notes from "Do" to "Ti", "flat" and "sharp" to denote ♭ and ♯, and "O" to denote rest[3].

Audio Music

Different from sheet music, which we created the data and developed the analysis from scratch, the audio music in the format of midi data were generated based on Toolbox in MATLAB (Toiviainen and Eerola, 2016). We use the Toolbox to visualize the audio music and track the improvisational or solo part in wave form.

Take song *Sarabande* from J.S. Bach's *Partita in A minor for Solo Flute* (BWV 1013) as an example: Figure 4.2 shows the movement of the key over time. The key moves from a minor to F major, further to b minor finally move back to a minor.

[2]MuseScore: https://musescore.org/en
[3]A rest is an interval of silence in a piece of music.

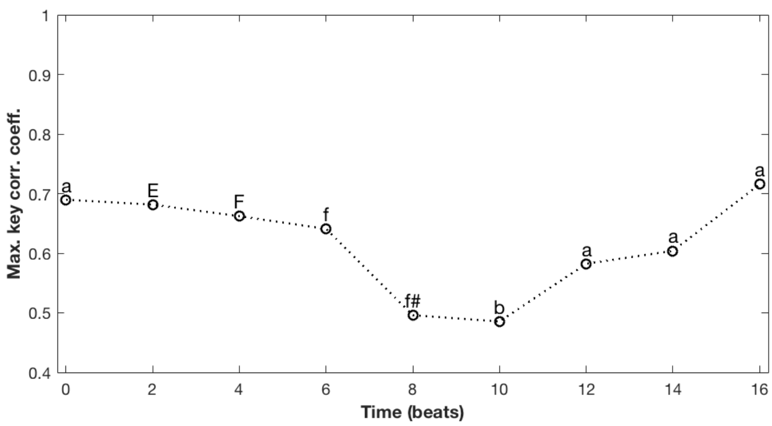


FIGURE 4.2: Maximum key correlation coefficients across time of song *Sarabande*

Figure 4.3 displays the dynamic tonality model with the dispersion of the key center between the alternate local keys. The tonality movement is also consistent with the key correlation plot in Figure 4.2. We can see the tonal center begins from a minor and then moves to other regions such as F, d, a, b and finally moves back to a minor.

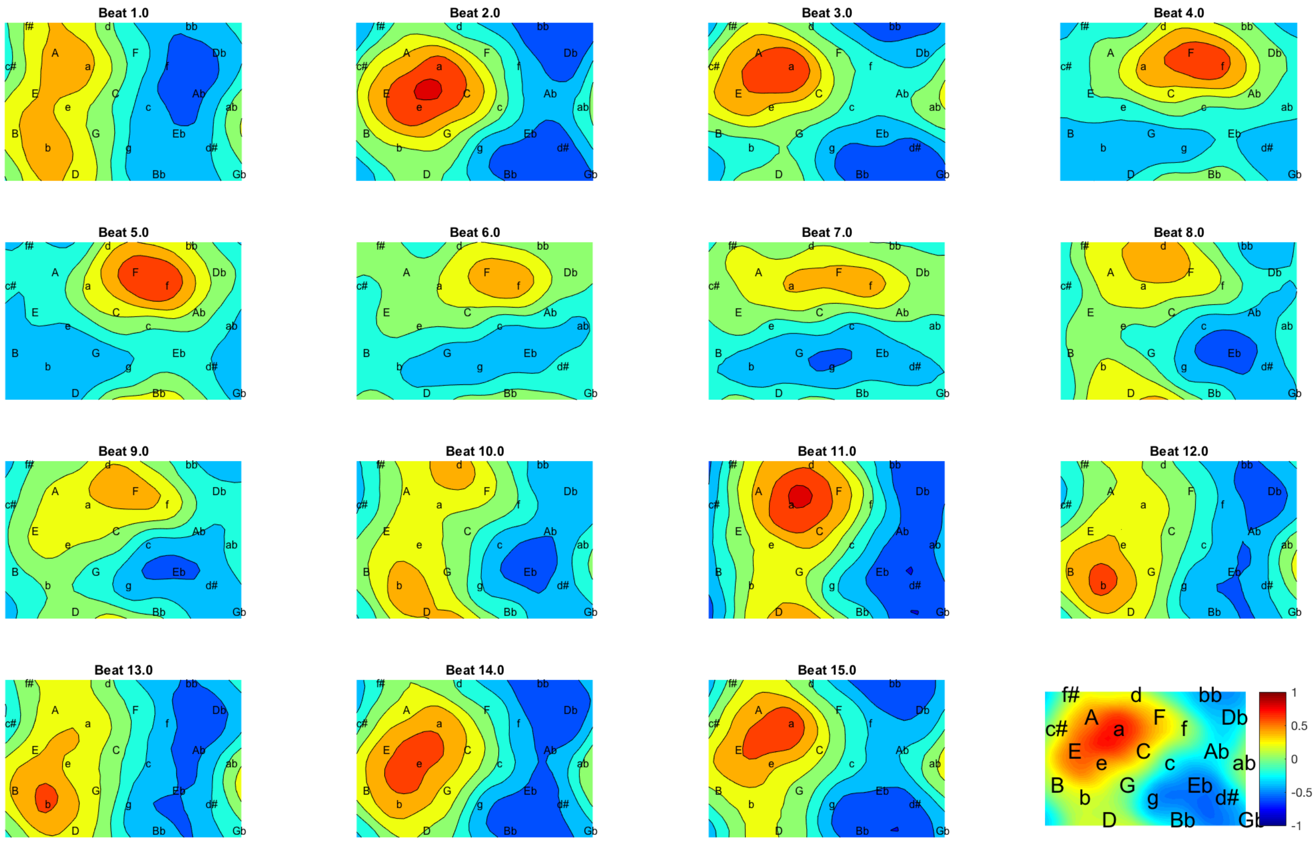


FIGURE 4.3: 16 beats of the tonality animation of song *Sarabande*

4.2 Other Music Genres

As I mentioned in the very beginning of this thesis, the initial motivation triggers me to do this topic is to answer the question *Why music from diverse culture can bring people so many different feelings?* So purely focusing on Jazz would not sufficiently help me to figure out the answer. Therefore I decide to add three different genres of music and compare them with Jazz respectively. I select songs from China, Japan and Arab due to their unique cultural characteristics.

Chinese Music

Chinese classical music reaches its peak around 224 to 262 A.D. It is based on the pentatonic scale, with heptatonic scale occasionally appear as the expansion. From Deva (1999), he mentioned that the exertion of timbre raises tone to a position of great importance. For example, Chinese musicians use use of portamento and vibrato which give a feeling of weeping or complaint.

Japanese Music

The traditional Japanese folk songs use pentatonic scale based on Western musical rules. In this pentatonic scale the subdominant and leading tone are ignored. This would lead to a musical scale with no half steps between note. According to Deva (1999), though Chinese music was exported to Japan, Japan did have a musical tradition before the advent of Chinese influences. The tradition existed in popular songs, indigenous Shinto religion (based on ancestor and nature worship), ritual and chant and possibly in court music and dances.

Arabic Music

Arabic music is originated from Cairo, Egypt, the center of Arab world. Morocco, Saudi Arabia and Lebanon are also well-known areas generate many Arabic songs. Maqam is the basis of Arabic songs. It appears like the mode, but actually not. It can determine the tonic note, dominant note, and ending note. Unlike the tradition of Western music, Arabic music contains microtones. Microtones are notes that lie between notes in the Western chromatic scale. While notes in the chromatic scale are separated by semitones, notes in Arabic music can be separated by quarter tones.

4.3 Input Data

As demonstrated in the previous Section 4.1 and Section 4.2, for Jazz part I mainly studied work from 7 Jazz musicians (Duke Ellington, Miles Davis, John Coltrane, Charlie Parker, Louis Armstrong, Bill Evans, Thelonious Monk), and for the comparison with other music genres we focus on Chinese, Japanese, and Arabic music. So I create two different albums based on the Measure-Note matrices I generated in previous Step 4.1.2. I use two different ways to demonstrate the album.

4.3.1 Note-Based Representation

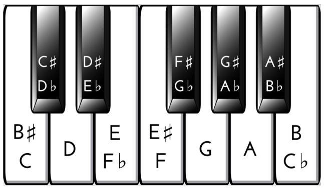


FIGURE 4.4: Music Key

Based on the 12 keys (5 black keys + 7 white keys) in the Figure 4.4, I make note-based representation according to the pitch class in Table A.1: forsaking the order of notes, we describe each measure in the song as a 12-dimension binary vector $\mathbf{X} = [x_1, x_2, ...x_{12}]$, where $x_i \in \{0, 1\}$ (Table 4.2, Appendix A.2)

TABLE 4.1: Pitch Class

Pitch Class	Tonal Counterparts	Solfege
1	C, $B\sharp$	do
2	$C\sharp$, $D\flat$	
3	D	re
4	$D\sharp$, $E\flat$	
5	E, $F\flat$	mi
6	F, $E\sharp$	fa
7	$F\sharp$, $G\flat$	
8	G	sol
9	$G\sharp$, $A\flat$	
10	A	la
11	$A\sharp$, $B\flat$	
12	B, $C\flat$	ti

TABLE 4.2: Notes collection from 4 Music Genres

Document	Pitch Class	Genre
China 1	0 0 0 0 1 0 1 0 0 0 0 1	China
China 2	0 0 0 0 1 0 1 0 0 0 0 0	China
China 3	0 0 0 0 0 0 1 0 0 0 0 1	China
⋮	⋮	⋮
China 7	0 1 0 0 1 0 1 0 0 0 0 1	China
China 8	0 0 0 0 1 0 1 0 0 0 0 1	China
⋮	⋮	⋮
Japan 1	1 0 1 1 0 0 1 0 0 0 0 0	Japan
Japan 2	1 0 0 0 0 0 0 1 0 0 0 0	Japan
⋮	⋮	⋮

- Document: song names, tantamount to document in text mining
- Pitch Class: binary vector whose element indicates if certain note is on, tantamount to word in text mining
- Genre: labeled contain Chinese songs, Japanese songs, Arabic songs, to compare with Jazz songs later
- The dimension of this data frame is 1469×3

Create the document term matrix (DTM) whose cells reflect the frequency of terms in each document. The rows of the DTM represent documents and columns represent term in the corpus. $A_{i,j}$ contains the number of times term j appeared in document i.

TABLE 4.3: Document Term Matrix

	Term			
Document	000000000000	000000000100	000000010100	...
Arab 5	15	6	20	...
Arab 7	0	5	5	...
China 6	1	12	0	...
China 7	13	0	1	...
Japan 4	8	4	1	...
Japan 5	0	0	0	...
USA 4	2	1	0	...
⋮	⋮	⋮	⋮	

4.3.2 Measure-Based Representation

TABLE 4.4: Notes collection from 7 musicians

Document	Notes	Musician
Charlie 1	B♭ O O O O O O O	Charlie
Charlie 1	B B♭ A A♭ G G G♭ F	Charlie
Charlie 1	E F G♭ B♭ G G A♭ O	Charlie
⋮	⋮	⋮
Charlie 7	E E E E G G C O	Charlie
Charlie 8	F♯ O O O O O O O	Charlie
⋮	⋮	⋮
Duke 1	C C C G G G G G	Duke
Duke 1	F F F A♭ A♭ A♭ B♭ B♭	Duke
⋮	⋮	⋮

- Document: song names, tantamount to document in text mining
- Notes: a series of notes in one measure, tantamount to word in text mining
- Musician: the composer, tantamount to the label for later analysis

- The dimension of this data frame is 5149×3

Create the document term matrix (DTM) whose cells reflect the frequency of terms in each document. The rows of the DTM represent documents and columns represent term in the corpus. $A_{i,j}$ contains the number of times term j appeared in document i. Dimension of DTM is 83×2960 with the last column as label: Duke, Miles, John, Charlie, Louis, Bill, Monk.

TABLE 4.5: Document Term Matrix

	Term			
Document	O O O O O O O O	B D B B D D E E	C A A♯ B D C A O	...
Miles 6	40	0	0	...
Louis 2	32	0	0	...
Sonny 3	26	0	0	...
Miles 2	25	0	0	...
Duke 4	0	9	0	...
Sonny 4	14	0	0	...
Charlie 9	0	0	8	...
⋮	⋮	⋮	⋮	

We can also talk a close look at the most frequent terms in the whole album: terms appear more than 20 times:

TABLE 4.6: Most Frequent Terms

Term
O O O O O O O O
C C C C C C C C
A A A A O O O O
B♭ B♭ B♭ B♭ B♭ B♭ B♭ B♭
B B B B B B B B
D D D D D D D D
G G G G G G G G
A A A A A A A A

4.4 Pattern Recognition

We take the topic proportion matrix as input and employ it on machine learning techniques for classification. We conduct the supervised analysis via 5 models with k-fold cross-validation:

- K Nearest Neighbors
- Multi-class Support Vector Machine
- Random Forest
- Neural Networks with PCA Analysis
- Penalized Discriminant Analysis

Algorithm 3 Supervised Analysis: 10-fold cross-validation with 3 times resampling

for $i \leftarrow 1:3$ **do**
  for $j \leftarrow 1:10$ **do**
    Split dataset $\mathcal{D} = \{\mathbf{z}_l, l = 1, 2, ..., n\}$ into k chunks so that $n = Km$
    Form subset $\mathcal{V}_j = \{\mathbf{z}_l \in \mathcal{D} : i \in [1 + (j-1) \times m, j \times m]\}$
    Extract train set $\mathcal{T}_j := \mathcal{D} \backslash \{\mathcal{V}_j\}$
    Build estimator $\hat{g}^{(\star)}(\cdot)$ using $\mathcal{T}_j$
    Compute predictions $\hat{g}^{(j)}(\mathbf{x}_l)$ for $\mathbf{z}_k \in \mathcal{V}_j$
    Calculate the error $\hat{\epsilon}_j = \frac{1}{m} \sum_{\mathbf{z}_l \in \mathcal{V}_j} l(y_l, \hat{g}^{(j)}(\mathbf{x}_l))$
  end for
  Compute $\texttt{CV}(\hat{g}) = \frac{1}{K} \sum_{j=1}^{K} \hat{\epsilon}_j$
  Find $\hat{g}^{(\star)}(\cdot) = \underset{j=1:J}{\texttt{argmin}}\{\texttt{CV}(\hat{g}(\cdot))\}$ with lowest prediction error
end for

4.4.1 K-Nearest Neighbors

kNN predicts the class of song via finding the k most similar songs, where the similarity is measured by Euclidean distance between two song vectors in this case. The class (label) here is the 7 musicians: Duke, Miles, John, Charlie, Louis, Bill, Monk.

Algorithm 4 k-Nearest Neighbors

for $i \leftarrow 1:n$ **do**
  Choose the value of k for $\mathcal{D} = \{(\mathbf{x}_1, Y_1), ..., (\mathbf{x}_i, Y_i), ..., (\mathbf{x}_n, Y_n), Y_i \in \{1, ..., g\}\}$
  Let $\mathbf{x}^\star$ be a new point. Compute $d_i^\star = d(\mathbf{x}^\star, \mathbf{x}_i)$
end for
Rank all the distance $d_i^\star$ in order: $d_{(1)}^\star \leq d_{(2)}^\star \leq ... \leq d_{(k)}^\star \leq ... \leq d_{(n)}^\star$
Form $\mathcal{V}_k(\mathbf{x}^\star) = \{\mathbf{x}_i : d(\mathbf{x}^\star, \mathbf{x}_i) \leq d_{(k)}^\star\}$
Predict response $\hat{Y}_{kNN}^\star =$ Most frequent label in $\mathcal{V}_k(\mathbf{x}^\star) = \underset{j \in \{1,...,g\}}{\texttt{argmax}}\{p_j^{(k)}(\mathbf{x}^\star)\}$
where $p_j^{(k)}(\mathbf{x}^\star) = \frac{1}{k} \sum_{\mathbf{x}_i \in \mathcal{V}_k(\mathbf{x}^\star)} \mathbf{I}(Y_i = j)$

4.4.2 Support Vector Machine

The task of Support Vector Machine (SVM) is to find the *optimal hyperplane that separates the observations in such a way that the margin is as large as possible.* That is to say, the distance between the nearest sample patterns (support vectors) should be as large as possible. SVM is originally designed as binary classifier, so in this case there are more than two classes, we use multi-class SVM. Specifically, we transform single multi-class task into multiple binary classification task. We train K binary SVMs and maximize the margins from each class to the remaining ones. We choose linear kernel (Eq.4.1) due to its excellent performance on high dimensional data that are very sparse in text mining.

$$\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) =< \mathbf{x}_i, \mathbf{x}_j >= \mathbf{x}_i^\top \mathbf{x}_j \tag{4.1}$$

Algorithm 5 Multi-class Support Vector Machine

for $k \leftarrow 1 : K$ **do**
  Given $\mathcal{D} = \{(\mathbf{x}_1, Y_{1k}), ..., (\mathbf{x}_i, Y_{ik}), ..., (\mathbf{x}_n, Y_{nk}), Y_{ik} \in \{+1, -1\}\}$
  Find function $h(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b$ that achieves
  $\max_{\mathbf{w},b} \left[\min_{y_{ik}=+1} \left(\frac{|\mathbf{w}^\top \mathbf{x}_i + b|}{||\mathbf{w}||} \right) + \min_{y_{ik}=-1} \left(\frac{|\mathbf{w}^\top \mathbf{x}_i + b|}{||\mathbf{w}||} \right) \right] = \max_{\mathbf{w},b} \frac{2}{||\mathbf{w}||} = \min_{\mathbf{w},b} \frac{1}{2} ||\mathbf{w}||^2$
  subject to $Y_{ik}(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \forall i = 1, 2, ..., n$
end for
Get $\underset{k=1,...,K}{\texttt{argmax}} f_k(\mathbf{x}) = \underset{k=1,...,K}{\texttt{argmax}} (\mathbf{w}_k^\top \mathbf{x} + b_k)$

4.4.3 Random Forest

Random Forest (RF) as an ensemble learning method that optimal the performance of single tree. Compared with tree bagging, the only difference in random forest is that then select each tree candidate with random subset of features, called *"feature bagging"*, for correction of overfitting issue of trees. If some features weigh more strongly than other features, these features will be selected in many of B trees among the whole forest.

Algorithm 6 Random Forest

for $b \leftarrow 1 : B$ **do**
  Draw with replacement from $\mathcal{D}$ a sample $\mathcal{D}^{(b)} = \{\mathbf{z}_1^{(b)}, ..., \mathbf{z}_n^{(b)}\}$
  Draw subset $\{i_1^{(b)}, ..., i_d^{(b)}\}$ of d variables without replacement from $\{1, 2, ..., p\}$
  Prune unselected variables from the sample $\mathcal{D}^{(b)}$ to ensure $\mathcal{D}_{sub}^{(b)}$ is d dimension
  Build tree (base learner) $\hat{g}_{(b)}$ based on $\mathcal{D}_{sub}^{(b)}$
end for
Output the result based on the mode of classes $\hat{g}^{RF}(\mathbf{x}) = \underset{j \in \{1,...,B\}}{\texttt{argmax}} \{p_j^{(b)}(\mathbf{x})\}$
where $p_j^{(k)}(\mathbf{x}^\star) = \frac{1}{B} \sum \mathbf{I}(\hat{g}_{(b)}(\mathbf{x}) = j)$

4.4.4 Neural Network with PCA Analysis

Principal Components Analysis (PCA) as one of the most common dimension reduction methods can help improve the result of classification. Neural Network with

Principal Component Analysis method proposed by Ripley (2007) is to run principal component analysis on the data first and then use the component in the neural network model. Each predictor has more than one values as the variance of each predictor is used in PCA analysis, and the predictor only has one value would be removed before the analysis. New data for prediction are also transformed with PCA analysis before feed to the networks.

Algorithm 7 Neural Network with PCA Analysis

Given data $\mathcal{D} = \{\mathbf{x}_1, ..., \mathbf{x}_n\}, \mathbf{x}_i \in \mathbb{R}^m$, finding $\hat{\Sigma}$ as estimates
for $i \leftarrow 1 : p$ **do**
  Obtain eigenvalues $\hat{\lambda}_i$ and eigenvectors $\hat{e}_i$ from $\hat{\Sigma}$
  Obtain principal components $y_i = \hat{e}_j^\top X$
end for
Get p-dimensional input vector $\mathbf{y} = (y_1, y_2, ..., y_p)^\top$ after PCA analysis
for $j \leftarrow 1 : q$ **do**
  Compute linear combination $h_j(\mathbf{y}) = \beta_{0j} + \beta_j^\top \mathbf{y}$ for each node in hidden layer
  Pass $h_j(\mathbf{y})$ through nonlinear activation function $z_j = \psi(\beta_{0j} + \sum_{l=1}^{p} \beta_{lj} y_l)$
end for
Combine z_j with coefficients to get $\eta(\mathbf{y}) = \gamma_0 + \sum_{j=1}^{q} \gamma_j \psi(\beta_{0j} + \sum_{l=1}^{p} \beta_{lj} y_l)$
Pass $\eta(\mathbf{y})$ with another activation function to output layer $\mu_k(\mathbf{y}) = \phi_k(\eta(\mathbf{y}))$

4.4.5 Penalized Discriminant Analysis

Linear Discriminant Analysis (LDA) is common tool for classification and dimension reduction. However, LDA can be too flexible in the choice of β with highly correlated predictor variables. Hastie, Buja, and Tibshirani (1995) came up with Penalized Discriminant Analysis (PDA) to avoid the overfitting performance resulting from LDA. Basically a penalty term is added to the covariance matrix $\Sigma'_W = \Sigma_W + \Omega$.

Algorithm 8 Penalized Discriminant Analysis

for $i \leftarrow 1 : n$ **do**
  Given data $\mathcal{D} = \{(\mathbf{x}_1, Y_1), ..., (\mathbf{x}_n, Y_n)\}, \mathbf{x}_i \in \mathbb{R}^q$
  Compute within-class covariance matrix $\hat{\Sigma}_w = \sum_{i=1}^{n} (\mathbf{x}_i - \mu_{y_i})(\mathbf{x}_i - \mu_{y_i})^\top + \Omega$
  Compute between-class covariance matrix $\hat{\Sigma}_b = \sum_{j=1}^{m} n_j (\mathbf{x}_j - \mu_{y_j})(\mathbf{x}_j - \mu_{y_j})^\top$
end for
Maximize the ratio of two matrices: $\hat{\mathbf{w}} = \underset{\mathbf{w}}{\texttt{argmax}} \frac{\mathbf{w}^\top \hat{\Sigma}_b \mathbf{w}}{\mathbf{w}^\top \hat{\Sigma}_w \mathbf{w}}$

4.4.6 Model Evaluation

Note-Based Model

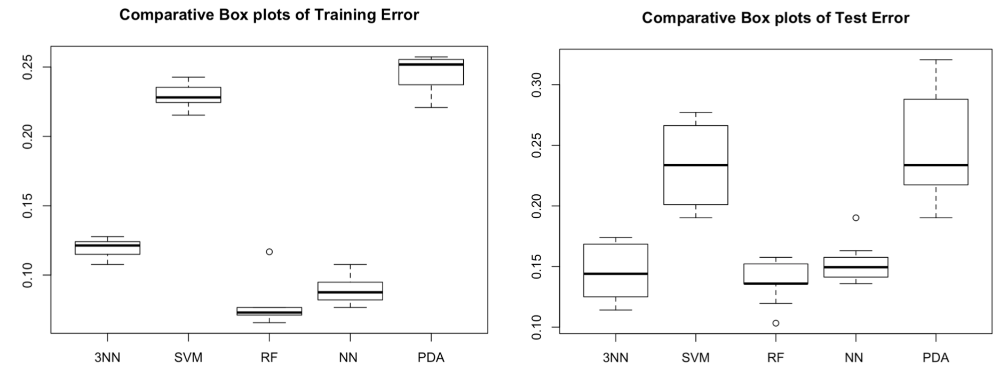


FIGURE 4.5: Pattern Recognition on Jazz and Chinese Music

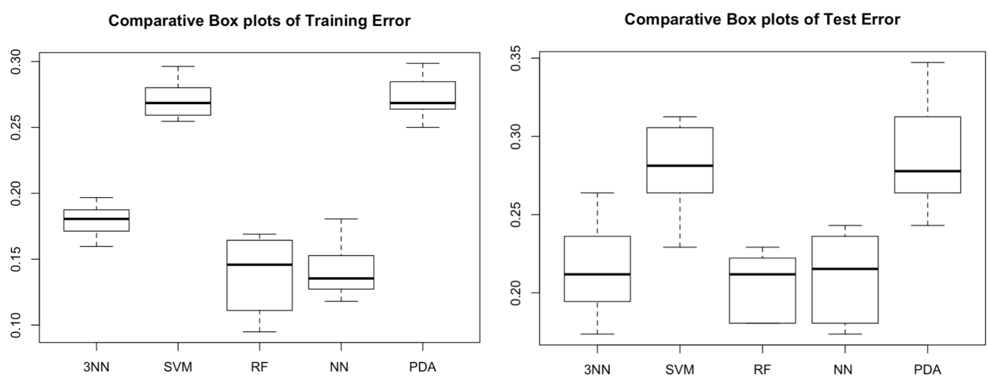


FIGURE 4.6: Pattern Recognition on Jazz and Japanese Music

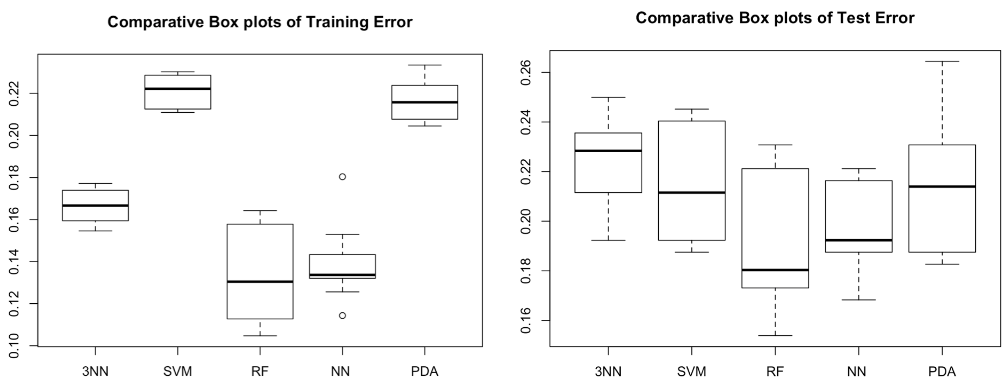


FIGURE 4.7: Pattern Recognition on Jazz and Arabic Music

Measure-Based Model

TABLE 4.7: Confusion Matrix: K Nearest Neighbors

	Reference						
Prediction	Charlie	Duke	John	Louis	Miles	Monk	Sonny
Charlie	7	1	6	0	2	3	1
Duke	0	0	0	0	0	0	0
John	0	0	4	0	0	0	1
Louis	0	0	3	6	2	0	1
Miles	0	10	7	8	5	7	1
Monk	0	0	0	0	0	0	0
Sonny	0	1	0	1	0	0	2

TABLE 4.8: Confusion Matrix: Support Vector Machine

	Reference						
Prediction	Charlie	Duke	John	Louis	Miles	Monk	Sonny
Charlie	11	0	0	0	0	0	0
Duke	0	12	0	0	0	0	0
John	0	0	20	0	0	0	0
Louis	0	0	0	15	0	0	0
Miles	0	0	0	0	9	1	0
Monk	0	0	0	0	0	9	0
Sonny	0	0	0	0	0	0	6

TABLE 4.9: Confusion Matrix: Random Forest

	Reference						
Prediction	Charlie	Duke	John	Louis	Miles	Monk	Sonny
Charlie	11	0	0	0	0	0	0
Duke	0	12	0	0	0	0	0
John	0	0	20	0	0	0	0
Louis	0	0	0	15	0	0	0
Miles	0	0	0	0	8	0	0
Monk	0	0	0	0	1	10	0
Sonny	0	0	0	0	0	0	6

TABLE 4.10: Confusion Matrix: Neural Networks

	Reference						
Prediction	Charlie	Duke	John	Louis	Miles	Monk	Sonny
Charlie	11	0	0	0	0	0	1
Duke	0	12	0	0	0	1	1
John	0	0	20	0	0	0	0
Louis	0	0	0	15	0	0	1
Miles	0	0	0	0	9	1	0
Monk	0	0	0	0	0	9	0
Sonny	0	0	0	0	0	0	3

TABLE 4.11: Confusion Matrix: Penalized Discriminant Analysis

	Reference						
Prediction	Charlie	Duke	John	Louis	Miles	Monk	Sonny
Charlie	11	0	0	0	0	0	0
Duke	0	12	0	0	0	0	0
John	0	0	20	0	0	0	0
Louis	0	0	0	15	0	0	0
Miles	0	0	0	0	9	1	0
Monk	0	0	0	0	0	9	0
Sonny	0	0	0	0	0	0	6

TABLE 4.12: Model Accuracy Comparison

Model	Accuracy
K Nearest Neighbors	28.92%
Support Vector Machine	98.80%
Random Forest	98.80%
Neural Network	95.18%
Discriminant Analysis	97.78%

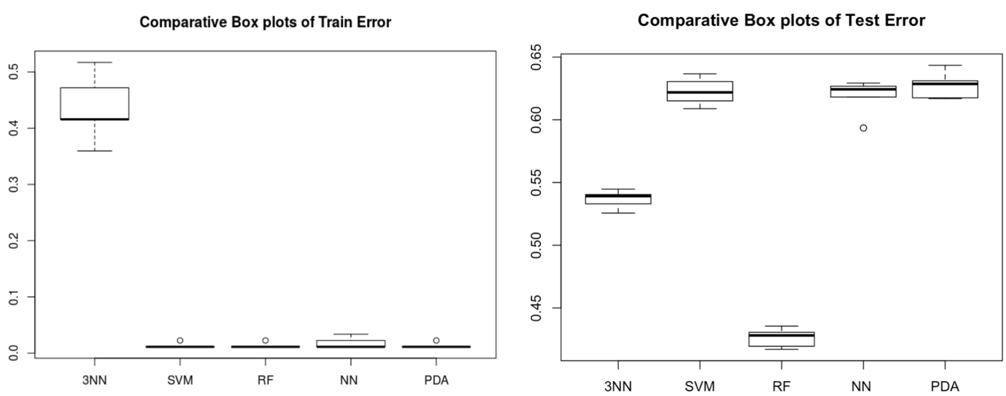


FIGURE 4.8: Pattern Recognition on Different Jazz Musicians

4.4.7 Comments and Conclusion

For note-based model we can see that the five supervised machine learning techniques could all classify different music genre with error rate no more than 35%. In addition, the performance of random forest, k nearest neighbors, and neural networks with PCA analysis are much better than the other two methods. Among the three comparisons (Jazz vs. Chinese music, Jazz vs. Japanese music, Jazz vs. Arabic music), the comparison of Jazz vs. Chinese would give better result than the other two, with random forest reaching lower than 0.1 error rate. For recognition between Jazz and Chinese songs, random forest is the best one with lowest error rate and variance. For recognition between Jazz and Japanese songs, k nearest neighbors, neural network and random forest have comparatively low error rate, but k nearest neighbors' performance has smaller variance. For comparison between Jazz and Arabic songs, neural network and random forest have comparatively low error rate, while they all have large variance.

For measure-based model, we can see that from the confusion matrix of training set, the model accuracy rate is very high for all techniques expect k nearest neighbors. However, but for the test set all the model fails to provide very good result with lowest error rate as 0.4 from random forest. It is obvious that this scenario has the challenging of overfitting issue. Further investigation is necessary if we want to use this representation.

4.5 Latent Dirichlet Allocation Model

4.5.1 Perplexity

In topic modeling, the number of topics is crucial for the to achieve its optimal performance. Perplexity is one way to measure how well is predictive ability of a probability model. Having the optimal topic number is always helpful in the sense to reach the best result with minimum computational time. Perplexity of a corpus $\mathcal{D}$ of M documents is computed as below Equation (4.2).

$$P(\mathcal{D}) = \exp\left(\frac{-\sum_{d=0}^{M-1} \log p(w_d;\lambda)}{\sum_{d=0}^{M-1} N_d}\right) \tag{4.2}$$

Apart from the above common way, there are many other methods to find the optimal topics. The existing `ldatuning` package stores 4 methods to calculate all metrics for selecting the perfect number of topics for LDA model all at once.

TABLE 4.13: Perplexity of Different Matrices

Topics Number	Griffiths2004	CaoJuan2009	Arun2010	Deveaud2014
2	-7454.086	0.11290217	13.856421	1.8604276
4	-6821.928	0.07120480	8.508257	1.7877936
6	-6516.431	0.06146701	5.613616	1.7126743
8	-6322.309	0.05740186	3.728195	1.6422201
10	-6184.650	0.05336498	2.404497	1.5998098
16	-6112.754	0.06507096	1.328469	1.3594688
20	-6101.264	0.07099931	1.512142	1.2242214
26	-6129.508	0.09352393	1.856783	1.0760613
30	-6121.120	0.10582645	2.545512	0.9585189
36	-6177.121	0.12330036	4.078891	0.8530592
40	-6183.168	0.14128330	5.226102	0.7767756
46	-6224.206	0.15072742	5.372056	0.7119278
50	-6253.992	0.16448002	6.637710	0.6719547
60	-6352.595	0.20606817	7.769699	0.5844223
72	-6325.653	0.25947947	9.892807	0.4742397
80	-6393.940	0.26968788	10.187645	0.4463054

Table 4.13 shows 4 different evaluating matrices. The extrema in each scenario illustrates the optimal number of topics.

- minimum
  - Arun2010 (Arun et al., 2010)
  - CaoJuan2009 (Cao et al., 2009)
- Maximum
  - Deveaud2014 (Deveaud, SanJuan, and Bellot, 2014)
  - Griffiths2004 (Griffiths and Steyvers, 2004)

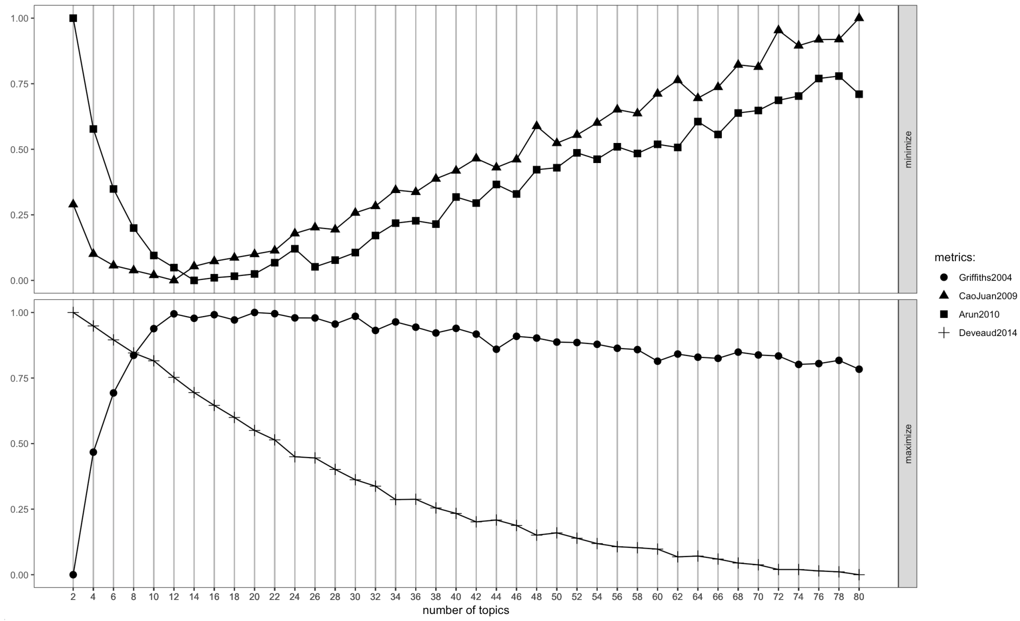


FIGURE 4.9: Evaluating LDA Models

From perplexity we can come to the conclusion that the optimal number of topics is around 8∼12. In this scenario Metric *Deveaud2014* is not as informative as the other three.

4.5.2 Discussion

Figure 4.10 shows the top 10 tokens in the topics from two scenarios.

For Measure-Based Scenario, we can see some topics purely natural keys:
e.g. Topic 1: $[E, O, O, O, O, O, O, O]$, Topic 5: $[B, D, B, B, D, D, E, E]$.
While some topics are very complicated with many sharps and flats in the notes:
e.g. Topic 3: $[B\flat, A, F, A\flat, B\flat, B\flat, O, O]$, Topic 6: $[F, G, F, E, E\flat, B\flat, C\sharp, D]$.

For Note-Based Scenario, each token is a 12-dimension vector indicating which of the pitch are "on" in certain measure. Some of the topics contains many active notes:
e.g. In Topic 2, some tokens have at most 7 active pitches.
While some topics are very silent with only few active notes:
e.g. In Topic 4 most pitches are mute, tokens have at most 3 active pitches.

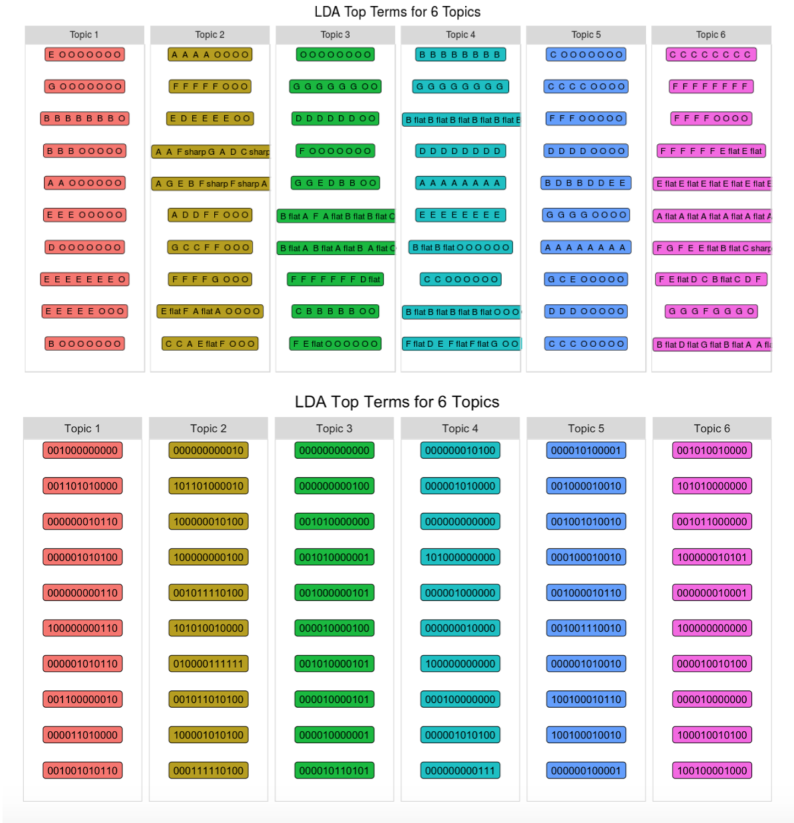


FIGURE 4.10: Top 10 Tokens in Selected Topic in Two Scenarios

Figure 4.11 shows the per-topic per-word probability of Measure-Based Scenario. We can see some topics appear very complicated with most of terms with flat or sharp notes (Topic 3, Topic 4). Some topics are very simple (Topic 8). Some topics contain too many terms with the same probability (Topic 2, Topic 4).

FIGURE 4.11: Topic Terms Distribution from Measure-Based Scenario

Figure 4.12 shows the per-topic per-word probability of Note-Based Scenario. Topic 4 and Topic 2 have certain distinctive terms while terms in Topic 9 have fairly similar probability. Further investigation involved musician is needed to better interpret the result.

FIGURE 4.12: Topic Terms Distribution from Note-Based Scenario

Lastly I draw chord diagram to see some potential relationship between topics learned from topic models and the targeted subjects.
In Figure 4.13, we can see:

- American songs (Jazz music in this case) are particularly dominant in Topic 9, which has most probable term $[1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1]$. It can also be interpreted as pitch class set: $\{C, E, G, A, B\}$, 
- Arabic songs contribute mostly to Topic 3, which has various terms equally distributed (see Figure 4.12).
- Most of Chinese songs attributes to Topic 4 and Topic 5 which contain most probable G major or E minor scale $\{E, F\sharp, B\}$
- Japanese songs seem to have similar contribution to every topic.

In Figure 4.14, we can see:

- Musician John Coltrane, Sonny Rollins and Louis Armstrong has some certain preference towards certain topics.
- Other musicians do not show clear bias to a specific topic.

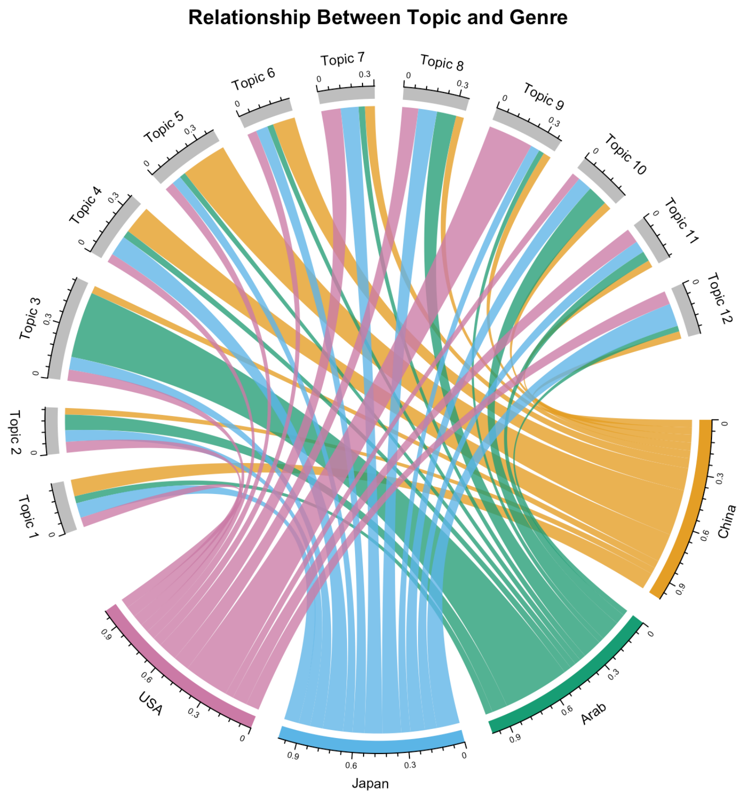


FIGURE 4.13: Chord Diagram for Music Genres

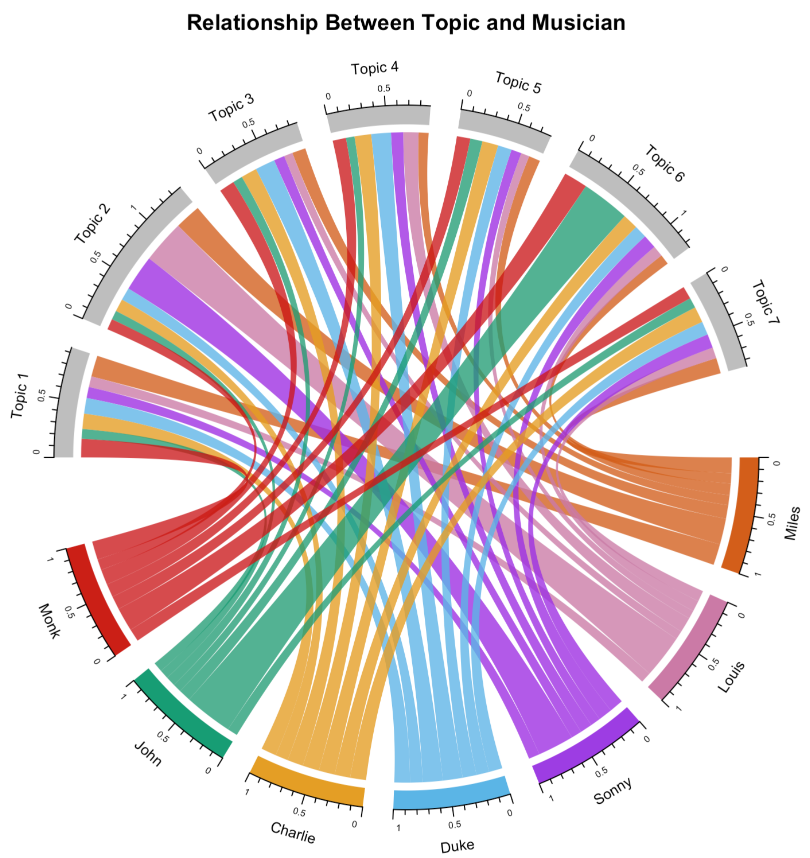


FIGURE 4.14: Chord Diagram for Jazz Music

Chapter 5

Conclusion

5.1 Summary

In this thesis I create two different representations in Chapter 4.3 for symbolic music and transform the music notes in music sheet into matrices for statistical analysis and data mining. Specifically, each song can be regarded as a text body consisting of different musical words. One way to represent these musical words is to segment the song into several parts based on the duration of each measure. Then the words in each song turn out to be a series of notes in one measure. Another way to represent music words is to restructure the notes in each segment based on the fixed 12-dimension pitch class. Both representations have been employed in pattern recognition and topic modeling techniques respectively, to detect music genres based on the collected songs, and figure out the potential connections between musicians and latent topics.

The predictive performance in pattern recognition for note-based representation turns out to be very good with 88% accuracy rate in the optimal scenario. In Chapter 4.5.2 I explore several aspects among music genres and musicians to see the hidden associations between different elements. Some genres contain very strong characteristics which make them very easy to detect. Jazz musicians John Coltrane, Sonny Rollins and Louis Armstrong show their particular preference towards certain topics. All these features are employed in the model to help better understand the world of music.

Furthermore, various model comparisons have been demonstrated in Chapter 3. I've compared latent Dirichlet allocation models between text mining and music mining. Within music field, compared symbolic music topic model and audio music topic model. In Chapter 2, I also include several relevant projects I've done during this two-year graduation training and compare LSA model with pLSA model, general LDA model with supervised LDA model, digit recognition with speech recognition in application.

5.2 Future Work

Music mining is a giant research field, and what I've done is merely a tip of the iceberg. Look back to the initial motivation that triggers me to embark on this research work: *Why does music from diverse culture have so powerful inherent capacity to bring people so many different feelings and emotions?* I have to say, to ultimately find out how to replace human intelligence with statistical algorithms for melody interpretation is still remain to be discovered.

Several potential studies I would love to continue exploring in the foreseeable future:

- Facilitate audio music and symbolic music transformation via machine learning technique.
- Deepen the understanding of musical lexicon and grammatical structure and create the dictionary in a mathematical way.
- How to derive representations for smooth recognition of Jazz by statistical learning methods?
- Apart from notes, can we embed other inherent musical structure such as cadence, tempo to better interpret the musical words?
- Explore the improvisation key learning (how many keys do the giants of jazz tended to play in, and what are those keys).
- Musical harmonies and its connection with elements of mood.

Appendix A

Selected Code

A.1 R Code for Extracting Notes from Music Sheet

Song *Hot House* from Charlie Parker and Dizzie Gillespie in `mxl` format:

FIGURE A.1: Piano Sheet for song *Hot House*

Transfer `mxl` file to `xml` file (partial code only for the second measure 

):

```
<measure number="1" width="252.10">
  <harmony print-frame="no">
    <root>
      <root-step>G</root-step>
      </root>
    <kind text="m7b5">half-diminished</kind>
    </harmony>
  <note default-x="12.00" default-y="-20.00">
    <pitch>
      <step>B</step>
      <alter>-1</alter>
      <octave>4</octave>
      </pitch>
    <duration>6</duration>
    <tie type="stop"/>
    <voice>1</voice>
    <type>eighth</type>
    <stem>down</stem>
    <beam number="1">begin</beam>
    <notations>
      <tied type="stop"/>
      </notations>
    </note>
  <note default-x="41.81" default-y="-20.00">
    <pitch>
      <step>B</step>
      <alter>-1</alter>
      <octave>4</octave>
      </pitch>
    <duration>6</duration>
    <voice>1</voice>
    <type>eighth</type>
    <accidental>flat</accidental>
    <stem>down</stem>
    <beam number="1">end</beam>
    </note>
  <note default-x="71.63" default-y="-25.00">
    <pitch>
      <step>A</step>
      <octave>4</octave>
      </pitch>
    <duration>6</duration>
    <voice>1</voice>
    <type>eighth</type>
    <stem>up</stem>
    <beam number="1">begin</beam>
    </note>
  <note default-x="101.44" default-y="-25.00">
    <pitch>
      <step>A</step>
      <alter>-1</alter>
```

```
      <octave>4</octave>
      </pitch>
    <duration>6</duration>
    <voice>1</voice>
    <type>eighth</type>
    <accidental>flat</accidental>
    <stem>up</stem>
    <beam number="1">end</beam>
    </note>
  <note default-x="131.25" default-y="-30.00">
    <pitch>
      <step>G</step>
      <octave>4</octave>
      </pitch>
    <duration>6</duration>
    <voice>1</voice>
    <type>eighth</type>
    <stem>up</stem>
    <beam number="1">begin</beam>
    </note>
  <note default-x="161.06" default-y="-30.00">
    <pitch>
      <step>G</step>
      <octave>4</octave>
      </pitch>
    <duration>6</duration>
    <voice>1</voice>
    <type>eighth</type>
    <stem>up</stem>
    <beam number="1">end</beam>
    </note>
  <note default-x="190.88" default-y="-30.00">
    <pitch>
      <step>G</step>
      <alter>-1</alter>
      <octave>4</octave>
      </pitch>
    <duration>6</duration>
    <voice>1</voice>
    <type>eighth</type>
    <accidental>flat</accidental>
    <stem>up</stem>
    <beam number="1">begin</beam>
    </note>
  <note default-x="220.69" default-y="-35.00">
    <pitch>
      <step>F</step>
      <octave>4</octave>
      </pitch>
    <duration>6</duration>
    <voice>1</voice>
```

```
        <type>eighth</type>
        <stem>up</stem>
        <beam number="1">end</beam>
        </note>
      </measure>
    <measure number="2" width="243.00">
```

R code for creating Measure-Note matrix based on the extracted notes:

	V1	V2	V3	V4	V5	V6	V7	V8
1	B flat	O	O	O	O	O	O	O
2	B	B flat	A	A flat	G	G	G flat	F
3	E	F	G flat	B flat	G	G	A flat	O
4	A	A flat	G	G flat	F	F	E	E flat
5	D	E flat	E	G sharp	E	E	O	O
6	G	A flat	B flat	G	B	O	O	O
7	B flat	A flat	G	E	E flat	C	O	O
8	G flat	F	F	F	F	F	E	O
9	C	C	O	O	O	O	O	O
10	G	A flat	B flat	C	B	F sharp	G	O
11	F sharp	D	O	O	O	O	O	O
12	F	G	A flat	B flat	A	E	F	O
13	E	C	O	O	O	O	O	O
14	G	G	F	G	B flat	A flat	G	F
15	B	B	A	B	D	C	B flat	A flat
16	G	G	D	B	C	C	O	O
17	G	G	F	O	O	O	O	O
18	C	D	E flat	F	F sharp	B flat	O	O
19	D	B	G flat	D	D	B	D	O
20	A	F sharp	G	A	B flat	C	O	O
21	F	D	G	F	F	O	O	O
22	G flat	B flat	F	D	A	G	O	O
23	F	F	D	B flat	O	O	O	O
24	F	A	E	D flat	A flat	F	O	O
25	E	E	D flat	A flat	B flat	O	O	O
26	B	B flat	A	A flat	G	G	G flat	F
27	E	F	G flat	B flat	G	G	A flat	O
28	A	A flat	G	G flat	F	F	E	E flat
29	D	E flat	E	G sharp	E	E	O	O
30	G	A flat	B flat	G	B	O	O	O
31	B flat	A flat	G	E	E flat	C	O	O
32	G flat	F	F	F	F	F	E	O
33	C	O	O	O	O	O	O	O

```
library(stringr)
library(XML)
source('MusicFunction.R')

doc       <- xmlParse(file = "example.xml")
xml_data  <- xmlToList(doc)
part      <- xml_data[["part"]]
measure   <- part[names(part) == "measure"]
## key signatures finding
attr      <- measure[[1]][names(measure[[1]]) == "attributes"]
key       <- attr$attributes$key$fifths
## store notes per measure every iteration ##
mat         <- list()
for (i in 1: length(measure)) {
  note    <- measure[[i]][names(measure[[i]]) == "note"]
  not     <- list()       ## list of notes per measure ##
  for (j in 1:length(note)) {
    step <- note[[j]][["pitch"]][["step"]]
```

```
    dura <- as.numeric(note[[j]][["duration"]])
    if( length(dura)==0  ){next}
    acc  <- note[[j]][["accidental"]]
    not[[j]] <- rep(paste(step,acc, sep = ' ' ), dura)
print(dura)
  }
  mat <- append(mat,list(unlist(not)))
}
mat   <- lapply(mat, 'length<-' , max(lengths(mat)))
## create measure-note matrix ##
mat   <- matrix(unlist(mat), nrow = length(measure), byrow = T)
## replace rest part NA to "O"
mat[is.na(mat)] <- "O"
## Use Key Signature Function to get the complete version
mat   <- trans(key,mat)
mat   <- matrix(str_replace_all(mat,"natural",""), nrow = length(measure))
write.csv(file = '~/exampl e.csv', x = mat)
```

A.2 Specific R Function

Key Signature Function

```
trans <- function(key, n){  # Key Signature Function
  #############
  ### Flats ###
  #############
  ## F major/D minor ##
  n[n=="B "] = ifelse(key == "-1", "B flat", "B ")
  ## B-flat major/G minor ##
  n[n=="B "] = ifelse(key == "-2", "B flat", "B ")
  n[n=="E "] = ifelse(key == "-2", "E flat", "E ")
  ## E-flat major/C minor ##
  n[n=="B "] = ifelse(key == "-3", "B flat", "B ")
  n[n=="E "] = ifelse(key == "-3", "E flat", "E ")
  n[n=="A "] = ifelse(key == "-3", "A flat", "A ")
  ## A-flat major/F minor ##
  n[n=="B "] = ifelse(key == "-4", "B flat", "B ")
  n[n=="E "] = ifelse(key == "-4", "E flat", "E ")
  n[n=="A "] = ifelse(key == "-4", "A flat", "A ")
  n[n=="D "] = ifelse(key == "-4", "D flat", "D ")
  ## D-flat major/B-flat minor ##
  n[n=="B "] = ifelse(key == "-5", "B flat", "B ")
  n[n=="E "] = ifelse(key == "-5", "E flat", "E ")
  n[n=="A "] = ifelse(key == "-5", "A flat", "A ")
  n[n=="D "] = ifelse(key == "-5", "D flat", "D ")
  n[n=="G "] = ifelse(key == "-5", "G flat", "G ")
  ## G-flat major/E-flat minor ##
  n[n=="B "] = ifelse(key == "-6", "B flat", "B ")
  n[n=="E "] = ifelse(key == "-6", "E flat", "E ")
  n[n=="A "] = ifelse(key == "-6", "A flat", "A ")
  n[n=="D "] = ifelse(key == "-6", "D flat", "D ")
```

```
  n[n=="G "] = ifelse(key == "-6", "G flat", "G ")
  n[n=="C "] = ifelse(key == "-6", "C flat", "C ")
  ## C-flat major/A-flat minor ##
  n[n=="B "] = ifelse(key == "-7", "B flat", "B ")
  n[n=="E "] = ifelse(key == "-7", "E flat", "E ")
  n[n=="A "] = ifelse(key == "-7", "A flat", "A ")
  n[n=="D "] = ifelse(key == "-7", "D flat", "D ")
  n[n=="G "] = ifelse(key == "-7", "G flat", "G ")
  n[n=="C "] = ifelse(key == "-7", "C flat", "C ")
  n[n=="F "] = ifelse(key == "-7", "F flat", "F ")
  ##############
  ### Sharps ###
  ##############
  ## G major/E minor ##
  n[n=="F "] = ifelse(key == "1", "F flat", "F ")
  ## D major/B minor ##
  n[n=="F "] = ifelse(key == "2", "F flat", "F ")
  n[n=="C "] = ifelse(key == "2", "C flat", "C ")
  ## A major/F-sharp minor ##
  n[n=="F "] = ifelse(key == "3", "F flat", "F ")
  n[n=="C "] = ifelse(key == "3", "C flat", "C ")
  n[n=="G "] = ifelse(key == "3", "G flat", "G ")
  ## E major/C-sharp minor ##
  n[n=="F "] = ifelse(key == "4", "F flat", "F ")
  n[n=="C "] = ifelse(key == "4", "C flat", "C ")
  n[n=="G "] = ifelse(key == "4", "G flat", "G ")
  n[n=="D "] = ifelse(key == "4", "D flat", "D ")
  ## B major/G-sharp minor ##
  n[n=="F "] = ifelse(key == "5", "F flat", "F ")
  n[n=="C "] = ifelse(key == "5", "C flat", "C ")
  n[n=="G "] = ifelse(key == "5", "G flat", "G ")
  n[n=="D "] = ifelse(key == "5", "D flat", "D ")
  n[n=="A "] = ifelse(key == "5", "A flat", "A ")
  ## F-sharp major/D-sharp minor ##
  n[n=="F "] = ifelse(key == "6", "F flat", "F ")
  n[n=="C "] = ifelse(key == "6", "C flat", "C ")
  n[n=="G "] = ifelse(key == "6", "G flat", "G ")
  n[n=="D "] = ifelse(key == "6", "D flat", "D ")
  n[n=="A "] = ifelse(key == "6", "A flat", "A ")
  n[n=="E "] = ifelse(key == "6", "E flat", "E ")
  ## C-sharp major/A-sharp minor ##
  n[n=="F "] = ifelse(key == "7", "F flat", "F ")
  n[n=="C "] = ifelse(key == "7", "C flat", "C ")
  n[n=="G "] = ifelse(key == "7", "G flat", "G ")
  n[n=="D "] = ifelse(key == "7", "D flat", "D ")
  n[n=="A "] = ifelse(key == "7", "A flat", "A ")
  n[n=="E "] = ifelse(key == "7", "E flat", "E ")
  n[n=="B "] = ifelse(key == "7", "B flat", "B ")
  return(n)
}
```

Further explanation for Key-Signature Function above:

Because key-signature in music piece  reflects in the `xml` file looking like this,

```
<key>
  <fifths>-1</fifths>
</key>
```

in the form of number. And then I came to know that in music there is an amazing circle called "Circle of Fifth" that can relate the number `<fifths>-1</fifths>` with the key I want. Based on this circle I wrote a function to transfer the number into flat/shape and then got the complete version of the Measure-Note Matrix.

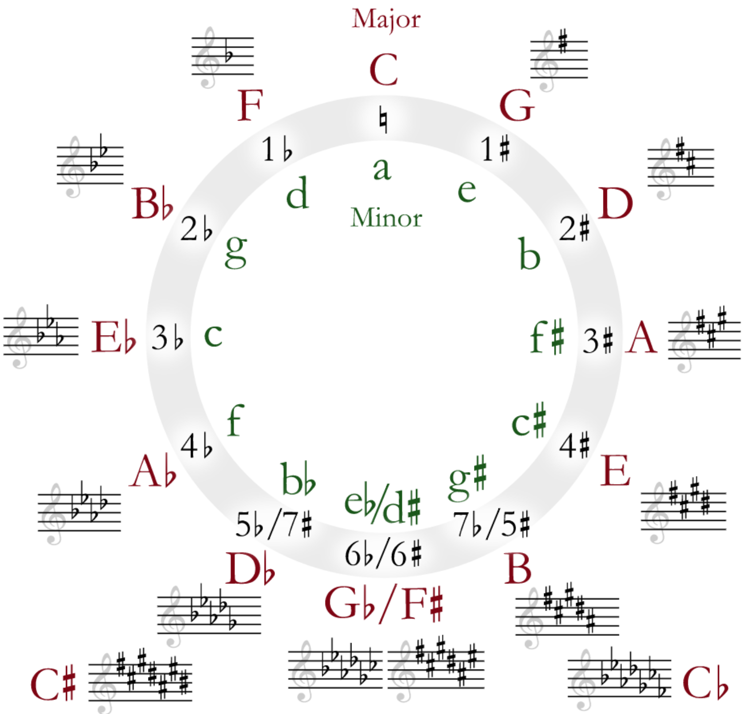


FIGURE A.2: Circle of Fifth

Key Index Function

```
ind <- function(x){
ind = ifelse(x == "C ", 1,
      ifelse(x == "B sharp", 1,
       ifelse(x == "C sharp", 2,
       ifelse(x == "D flat", 2,
        ifelse(x == "D ", 3,
         ifelse(x == "E flat", 4,
         ifelse(x == "D sharp", 4,
          ifelse(x == "E ", 5,
          ifelse(x == "F flat", 5,
           ifelse(x == "F ", 6,
           ifelse(x == "E sharp", 6,
```

```
            ifelse(x == "F sharp", 7,
            ifelse(x == "G flat", 7,
             ifelse(x == "G ", 8,
              ifelse(x == "A flat", 9,
              ifelse(x == "G sharp", 9,
               ifelse(x == "A ", 10,
                ifelse(x == "B flat", 11,
                ifelse(x == "A sharp", 11,
                 ifelse(x == "B ", 12,
                 ifelse(x == "C flat", 12, 0)))))))))))))))))))))
return(ind)
}
```

TABLE A.1: Pitch Class

Pitch Class	Tonal Counterparts	Solfege
1	$C, B\sharp$	do
2	$C\sharp, D\flat$	
3	D	re
4	$D\sharp, E\flat$	
5	$E, F\flat$	mi
6	$F, E\sharp$	fa
7	$F\sharp, G\flat$	
8	G	sol
9	$G\sharp, A\flat$	
10	A	la
11	$A\sharp, B\flat$	
12	$B, C\flat$	ti

A.3 MATLAB Code for Tonality Animation

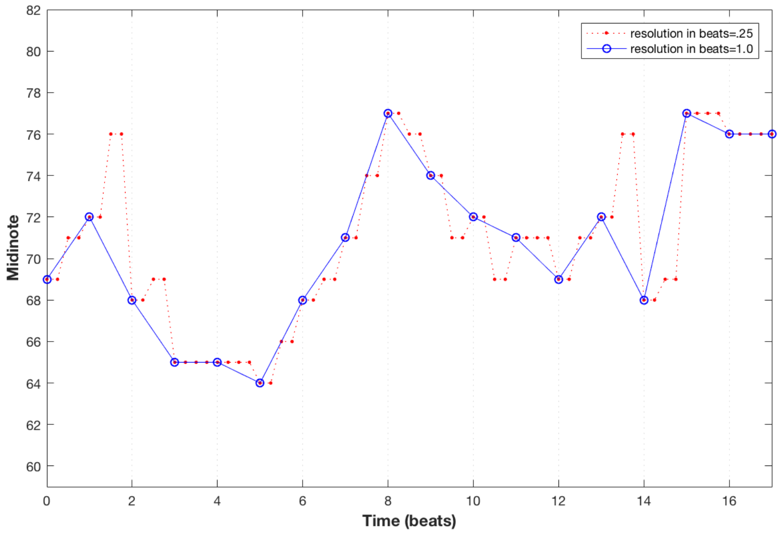


FIGURE A.3: Melodic contour of song *Sarabande*

Figure A.3 depicts two melodic contours with different degrees of resolution. The larger the resolution, the more coarse the contour.

```
%%  Reference  %%
Toiviainen, P., & Eerola, T. (2016). MIDI Toolbox 1.1.
URL: https://github.com/miditoolbox/1.1

nmat = readmidi('Sarabande.mid');
prelude = onsetwindow(nmat, 0, 32, 'beat');
keysomanim(prelude,1,2,'beat','strip');

plotmelcontour(prelude,0.25,'abs',':r.'); hold on
plotmelcontour(prelude,1,'abs','-bo'); hold off
legend(['resolution in beats=.25';
'resolution in beats=1.0']);
```

Appendix B

Theorem

B.1 Inequalities

Theorem B.1.1. *$X \in \mathcal{R}$, let $f(x)$ and $g(x)$ be monotone nondecreasing functions. Then*

$$\mathbb{E}\{f(X)g(X)\} \geq \mathbb{E}\{f(X)\}\mathbb{E}\{g(X)\}$$

If $f(x)$ is monotone increasing and $g(x)$ is monotone decreasing, then

$$\mathbb{E}\{f(X)g(X)\} \leq \mathbb{E}\{f(X)\}\mathbb{E}\{g(X)\}$$

Proof. For the first inequality:

$$\begin{aligned}
&\mathbb{E}\{f(X)g(X)\} - \mathbb{E}\{f(X)\}\mathbb{E}\{g(X)\} \\
&= \int f(x)g(x)\mu(dx) - \int f(y)\mu(dy) \int g(x)\mu(dx) \\
&= \int \left(\int [f(x) - f(y)]g(x)\mu(dx) \right) \mu(dy) \\
&= \int \left(\int h(x,y)g(x)\mu(dx) \right) \mu(dy) \quad \textit{(where } h(x,y) = f(x) - f(y)\textit{)} \\
&= \int_{\mathcal{R}^2} h(x,y)g(x)\mu^2(dxdy) \quad \textit{(from Fubini's theorem)} \\
&= \int_{x>y} h(x,y)g(x)\mu^2(dxdy) + \int_{x<y} h(x,y)g(x)\mu^2(dxdy) \\
&= \int_{x>y} h(x,y)g(x)\mu^2(dxdy) + \int_{x} \left(\int_{y>x} h(x,y)g(x)\mu(dy) \right) \mu(dx) \\
&= \int_{x>y} h(x,y)g(x)\mu^2(dxdy) + \int_{y} \left(\int_{x>y} h(y,x)g(y)\mu(dx) \right) \mu(dy) \\
&= \int_{y} \left(\int_{x>y} [h(x,y)g(x) + h(y,x)g(y)]\mu(dx) \right) \mu(dy) \\
&= \int_{y} \left(\int_{x>y} h(x,y)[g(x) - g(y)]\mu(dx) \right) \mu(dy) \quad \textit{(} h(x,y) \geq 0 \textit{ and } g(x) - g(y) \geq 0 \textit{)} \\
&\geq 0
\end{aligned}$$

For the second inequality:

$$
\begin{aligned}
&\mathbb{E}\{f(X)g(X)\} - \mathbb{E}\{f(X)\}\mathbb{E}\{g(X)\} \\
&\quad = \int f(x)g(x)\mu(dx) - \int f(y)\mu(dy) \int g(x)\mu(dx) \\
&\quad = \int \left(\int [f(x) - f(y)]g(x)\mu(dx) \right) \mu(dy) \\
&\quad = \int \left(\int h(x,y)g(x)\mu(dx) \right) \mu(dy) \quad \textit{(where } h(x,y) = f(x) - f(y)\textit{)} \\
&\quad = \int_{\mathcal{R}^2} h(x,y)g(x)\mu^2(dxdy) \quad \textit{(from Fubini's theorem)} \\
&\quad = \int_{x>y} h(x,y)g(x)\mu^2(dxdy) + \int_{x<y} h(x,y)g(x)\mu^2(dxdy) \\
&\quad = \int_{x>y} h(x,y)g(x)\mu^2(dxdy) + \int_x \left(\int_{y>x} h(x,y)g(x)\mu(dy) \right) \mu(dx) \\
&\quad = \int_{x>y} h(x,y)g(x)\mu^2(dxdy) + \int_y \left(\int_{x>y} h(y,x)g(y)\mu(dx) \right) \mu(dy) \\
&\quad = \int_y \left(\int_{x>y} [h(x,y)g(x) + h(y,x)g(y)]\mu(dx) \right) \mu(dy) \\
&\quad = \int_y \left(\int_{x>y} h(x,y)[g(x) - g(y)]\mu(dx) \right) \mu(dy) \quad \textit{(} h(x,y) \geq 0 \textit{ and } g(x) - g(y) \leq 0 \textit{)} \\
&\quad \leq 0
\end{aligned}
$$

$\square$

Bibliography

Arun, Rajkumar et al. (2010). "On finding the natural number of topics with latent dirichlet allocation: Some observations". In: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, pp. 391–402.

Blei, David M (2012). "Probabilistic topic models". In: *Communications of the ACM* 55.4, pp. 77–84.

Blei, David M, Andrew Y Ng, and Michael I Jordan (2003). "Latent dirichlet allocation". In: *Journal of machine Learning research* 3.Jan, pp. 993–1022.

Cao, Juan et al. (2009). "A density-based method for adaptive LDA model selection". In: *Neurocomputing* 72.7-9, pp. 1775–1781.

Deerwester, Scott et al. (1990). "Indexing by latent semantic analysis". In: *Journal of the American society for information science* 41.6, pp. 391–407.

Deva, Dharma (1999). "Underlying socio-cultural aspects and aesthetic principles that determine musical theory and practice in the musical traditions of China and Japan". In: *Renaissance Artists and Writers Association*.

Deveaud, Romain, Eric SanJuan, and Patrice Bellot (2014). "Accurate and effective latent concept modeling for ad hoc information retrieval". In: *Document numérique* 17.1, pp. 61–84.

Devroye, Luc, László Györfi, and Gábor Lugosi (2013). *A probabilistic theory of pattern recognition*. Vol. 31. Springer Science & Business Media.

Eerola, Tuomas and Petri Toiviainen (2004). "MIDI toolbox: MATLAB tools for music research". In:

Gouno, Evans (2018). personal communication.

Griffiths, Thomas L and Mark Steyvers (2004). "Finding scientific topics". In: *Proceedings of the National academy of Sciences* 101.suppl 1, pp. 5228–5235.

Hastie, Trevor, Andreas Buja, and Robert Tibshirani (1995). "Penalized discriminant analysis". In: *The Annals of Statistics*, pp. 73–102.

Hofmann, Thomas (1999). "Probabilistic latent semantic analysis". In: *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., pp. 289–296.

Hu, Diane and Lawrence K Saul (2009a). "A Probabilistic Topic Model for Unsupervised Learning of Musical Key-Profiles." In: *ISMIR*. Citeseer, pp. 441–446.

Hu, Diane J (2009). "Latent dirichlet allocation for text, images, and music". In: *University of California, San Diego. Retrieved April* 26, p. 2013.

Hu, Diane J and Lawrence K Saul (2009b). "A probabilistic topic model for music analysis". In: *Proc. of NIPS*. Vol. 9. Citeseer.

Kemp, Rebecca Ann Finnangan (2018). personal communication.

Kruger, Jonathan (2018). personal communication.

Krumhansl, Carol L. and Edward J. Kessler (1982). "Tracing the dynamic changes in perceived tonal organization in a spatial representation of musical keys." In: *Psychological Review* 89.4, pp. 334–368. DOI: 10.1037//0033-295x.89.4.334.

Krumhansl, Carol L and Mark Schmuckler (1990). "A key-finding algorithm based on tonal hierarchies". In: *Cognitive Foundations of Musical Pitch*, pp. 77–110.

Le Cun, Yann et al. (1990). "Handwritten zip code recognition with multilayer networks". In: *[1990] Proceedings. 10th International Conference on Pattern Recognition*. Vol. 2. IEEE, pp. 35–40.

Longuet-Higgins, H Christopher and Mark J Steedman (1971). "On interpreting bach". In: *Machine intelligence* 6, pp. 221–241.

Mcauliffe, Jon D and David M Blei (2008). "Supervised topic models". In: *Advances in neural information processing systems*, pp. 121–128.

Ripley, Brian D (2007). *Pattern recognition and neural networks*. Cambridge university press.

Silge, Julia (2018). *The game is afoot! Topic modeling of Sherlock Holmes stories*.

Spiliopoulou, Athina (2013). "Probabilistic models for melodic sequences". In:

Temperley, David (2002). "A Bayesian approach to key-finding". In: *Music and artificial intelligence*. Springer, pp. 195–206.

Temperley, David et al. (2007). *Music and probability*. Mit Press.

Toiviainen, P. and T. Eerola (2016). *MIDI toolbox 1.1*. https://github.com/miditoolbox/.

Toiviainen, Petri (2005). "Visualization of tonal content with self-organizing maps and self-similarity matrices". In: *Computers in Entertainment (CIE)* 3.4, pp. 1–10.

Willimek, Bernd and Daniela Willimek (2013). *Music and Emotions-Research on the Theory of Musical Equilibration (die Strebetendenz-Theorie)*. Bernd Willimek.

Wu, Qiuyi and Ernest Fokoue (2018). "Naive Dictionary On Musical Corpora: From Knowledge Representation To Pattern Recognition". In: *arXiv preprint arXiv:1811.12802*.