

On Mixture Model Complexity Estimation for Music Recommender Systems

Wietse Balkema, Ferdi van der Heijden and Bas Meijerink

Signal and Systems (EWI-SAS) Group, Department of Electrical Engineering, University of Twente, The Netherlands

Email: j.w.balkema@alumnus.utwente.nl

Abstract—Content-based music navigation systems are in need of robust music similarity measures. Current similarity measures model each song with the same model parameters. We propose methods to efficiently estimate the required number of model parameters of each individual song. First results of a study on relationships between a small set of basic audio features are presented. We conclude that there are only very small correlations between models on low- and on high-dimensional features.

When we compare a very simple clustering algorithm with an algorithm that estimates model parameters using the MDL criterium, we find a surprisingly strong correlation between the estimated number of mixture components.

I. INTRODUCTION

With the ever increasing availability of digital music, new music access strategies are needed. Perrot [1] found that college students were capable to classify a piece of music quite accurately in a 10-class genre taxonomy, while only listening to an excerpt of 250 ms. This is “fundamentally inexplicable with present models of music perception” [2] and justifies the statement that the audio surface contains a lot of information that can be used for music genre classification.

Content-based music recommendation seems to be the most promising solution for finding music in large music collections. These systems usually extract a set of high dimensional features from the audio signal and model these with a statistical model, where a Gaussian Mixture Model with a fixed number of gaussians (between 10 and 100) is most common.

The main problem of content based music recommendation engines is the lack of robustness of the recommendations. Aucouturier [3] finds that certain songs are always ranked very high in a nearest neighbor search. He suggests that these songs (named *hubs*) contain outlier frames, that have great impact on the song model. It is interesting to note that the ‘hubness’ of a song is found to be not an intrinsic property of the song, but rather a property of a given algorithm. For this reason, we present an alternative approach for modelling music.

Each song has its own characteristics: Instrumentation, vocals, rhythm, etc. These characteristics influence the spectrum and structure of the song and thus the data distribution of the extracted features. We hypothesize that the appearance of hubs can be reduced by analyzing a song’s feature complexity and adapting the number of components of a song’s model accordingly. To the best of our knowledge, this has not yet been applied in the field of Music Information Retrieval.

In section II, we will give a short overview of Music Information Retrieval (MIR) and some commonly used music descriptor features. Section III is on modelling these features and section IV deals with complexity estimation of music data. We present two methods to estimate the model complexity of high-dimensional features. In section V we present some results of both methods.

II. MUSICAL FEATURES

Music Information Retrieval is the science of extracting information from music for various purposes. In large music collections we want to minimize the number of required actions of a user to find the music he likes. Some ten years ago, Wold [4] identifies four methods how to access sounds:

- *Simile*: Find sounds that belong to a certain class.
- *Acoustical/perceptual features*: Find sounds that fulfill certain feature criteria.
- *Subjective features*: Find sounds using a personalized description scheme.
- *Onomatopoeia*: Query by humming.

These methods are still major areas of research in MIR.

Current music retrieval systems mostly rely on *timbre*-based features. Timbre is the collection of properties that distinguish the sound of a musical note, when this note is generated from different sources or instruments. Three timbre-based features are the *Zero Crossings Rate*, *Spectral Centroid* and *Mel-scale Frequency Cepstrum Coefficients*. These features are calculated over timeframes in which the audio signal is quasi-stationary. A common framelength is 20ms [5].

A. Zero Crossings Rate

The zero crossings rate is a time-domain based feature. It is a measure of the *noisiness* of an audio signal. The ZCR is defined as:

$$ZCR = \frac{1}{T} \sum_{n=1}^N |\text{sign}(x[n]) - \text{sign}(x[n-1])| \quad (1)$$

where T is the time in seconds, $x[n]$ the time domain signal of the audio signal and $N = T * \text{Samplerate}$. Figure 1 depicts two 5 second intervals of songs of two different genres. It is clear to see that the Beatles (Figure 1(a)) have a ‘cleaner’ sound than Greenday (Figure 1(b)).

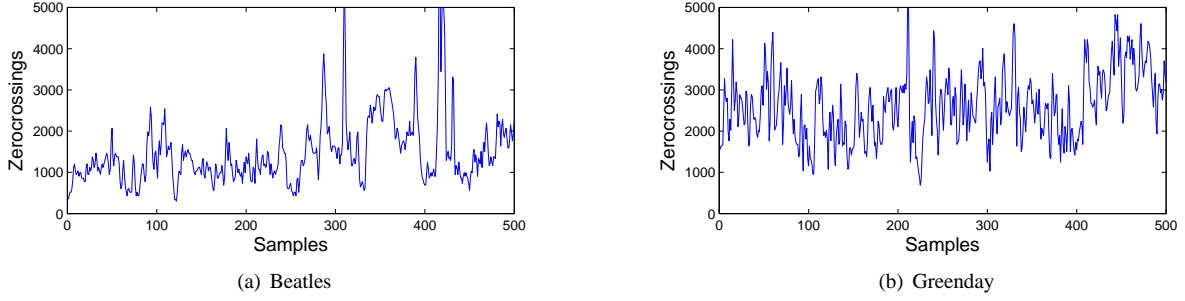


Fig. 1. Five second interval of Zero Crossings Rate for two songs

B. Spectral Centroid

The spectral centroid is defined as the center of gravity of the magnitude spectrum of the short-time Fourier transform. It is a measure of the *brightness* of the musical piece and is defined as:

$$SC = \frac{\sum_k kS(k)}{\sum_k S(k)} \quad (2)$$

where $S(k)$ is the power of the spectrum in the k^{th} frequency bin.

C. Mel-scale Frequency Cepstrum Coefficients

MFCC have first been used in speech recognition research and have proven to give a compact representation of the perceptually relevant frequency components in an audio signal. The MFCC is calculated as follows:

- 1) Convert the signal to frames
- 2) Take the discrete Fourier transform
- 3) Take the log of the amplitude spectrum
- 4) Apply the Mel-scaling and smoothing
- 5) Take the discrete cosine transform

The mel-scale is a nonlinear scale modelling perceived pitch. It can be approximated by:

$$\text{mel}(f) = 2595 \cdot \log_{10} \left(1 + \frac{f}{700} \right) \quad (3)$$

Aucouturier [5] systematically explored MFCC feature space and found the optimal number of components to use is 20.

III. MIXTURE MODELS

The timbre features mentioned above are calculated over $20ms$ windows, with a hopsize of $10ms$. For a three-minute song we thus have 18000 samples. For practical applications, this amount of data is too high. We therefore model the data with a mixture model.

A mixture model for a d -dimensional random variable \mathbf{x} is given by:

$$p(\mathbf{x}, \Theta) = \sum_{m=1}^k \alpha_m p_m(\mathbf{x}, \Theta_m) \quad (4)$$

where k is the number of components in the mixture and $\Theta = \{\alpha_1, \dots, \alpha_k, \Theta_1, \dots, \Theta_k\}$ are the model parameters. The mixture weights α_m are nonnegative and add up to one.

A. Gaussian Mixture Model

The most widespread mixture model type is the Gaussian Mixture Model. Each mixture component is a gaussian probability distribution. The parameters Θ of a gaussian are its mean μ and its covariance matrix Σ :

$$\mathcal{G}(\mathbf{x}, \mu, \Sigma_{\mathcal{X}}) = \frac{1}{(2\pi)^{N/2} |\Sigma_{\mathcal{X}}|^{1/2}} \cdot \exp \left(-\frac{1}{2} (\mathbf{x} - \mu)^T \Sigma_{\mathcal{X}}^{-1} (\mathbf{x} - \mu) \right) \quad (5)$$

The covariance matrix has to be positive definite.

B. Parameter Estimation

When the number of components in a mixture is known, the Expectation Maximization (EM) algorithm [6] provides an efficient method to estimate the parameters of the distribution of n datasamples $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$. The EM algorithm is an iterative procedure and is guaranteed to converge to a local maximum of the maximum (log-)likelihood estimate of the mixture parameters:

$$\hat{\Theta}_{\text{ML}} = \arg \max_{\Theta} (\log p(\mathcal{X} | \Theta)) \quad (6)$$

Each iteration consists of two steps:

- **E-step:** Assign each sample to the mixture component that is most likely to have generated the sample, based on the current estimate of the model parameters.
- **M-step:** Recompute the model parameters based on the current sample membership estimation.

These steps are repeated until convergence of the likelihood estimate.

IV. COMPLEXITY ESTIMATION

One major problem of the EM algorithm is that the number of mixture components in a mixture should be known in advance. When listening to various kinds of music, it is clear that there are broad variations in musical structure and sound. Even if this is recognized for music genre classification, where different feature sets are used for determining class likelihood (eg. [7]), no song-level feature model optimization is performed. We think that by modelling each song with an optimal number of components significantly improves classification

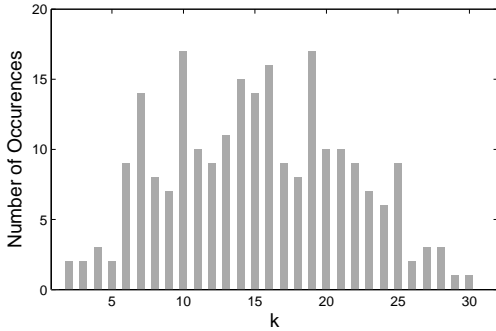


Fig. 2. Distribution of k_{opt} on a dataset of 234 songs

accuracy. For ‘complex’ songs, the number of components will be higher as for ‘simple’ songs.

We perform a study on feature complexity correlations between high and low dimensional features in order to be able to predict the optimal number of gaussians for a high dimensional feature by analyzing the complexity of a low dimensional feature. For this, we compare our predicted number of mixture components \hat{k} with a conventional ground truth k_{opt} . We use the algorithm presented by Figueiredo [8] to find k_{opt} for a small set of features.

A. Optimal Model Selection

Model selection algorithms try to find the number of components k , that minimize the cost function $\mathcal{C}(\hat{\Theta}(k), k)$:

$$\hat{k} = \arg \min_k \{\mathcal{C}(\hat{\Theta}(k), k)\}, \quad k = k_{\min}, \dots, k_{\max} \quad (7)$$

The cost function $\mathcal{C}(\hat{\Theta}(k), k)$ consists of two parts:

- A part expressing the goodness of fit of a model with k components. This function is a monotonically increasing function of k .
- A part penalizing models with higher k .

The method presented by Figueiredo uses a cost function that is based on the Minimum Description Length (MDL) criterion. This criterion is based on the idea that if you can describe some observed data with a short code, you have a good model of the source generating the data.

In Figure 3 we see the optimal number of mixture component as found by Figueiredo’s algorithm on a dataset of 234 songs. We see that k_{opt} varies between 3 and 30 around a center value of 15 components. Note that this value for k_{opt} is significantly less than the fixed value of $k = 50$ found by Aucouturier [5].

There are numerous algorithms that are also based on the MML criterion. Zivkovic [9] presents an algorithm that uses a coarse approximation of MML. It is much faster than Figueiredo, but also less robust.

B. Model Estimation

Although model selection algorithms like Figueiredo or Zivkovic presented provide reasonable speed, they are not

suitable to find the optimal number of model components for a dataset consisting of 20-dimensional features of 5000 songs because of computation time considerations. Therefore, we search for methods to efficiently estimate the minimal required model complexity of individual songs that are computationally less expensive than the algorithms mentioned above.

1) *Correlation between features*: We assume that the required model complexity is an intrinsic property of a song. Based on this assumption, we expect a relationship between the required number of components for modelling simple features such as the ZCR or SC and the required number for complex features such as MFCC.

2) *Correlation between algorithms*: When required model complexity is an intrinsic property of a song, different component estimation algorithms must also correlate. We have investigated correlations between the number of components k , found by very simple clustering algorithms and k_{opt} as found by Figueiredo. The most basic algorithm we use, is the “Basic Sequential Algorithmic Scheme” (BSAS [10]). As can be seen from the following pseudocode, BSAS only has two parameters: The threshold Θ for determining whether a new cluster has to be formed, and N_{maxClust} , the maximum number of clusters to be formed.

```

1:  $N_{\text{clust}} = 1$ 
2:  $C_1 = \{x_1\}$ 
3: for  $i = 2$  to  $N$  do
4:   find  $C_k: d(x_i, C_k) = \min_{\forall j} d(x_i, C_j)$ 
5:   if  $(d(x_i, C_k) > \Theta)$  and  $(N_{\text{clust}} < N_{\text{maxClust}})$  then
6:      $N_{\text{clust}} = N_{\text{clust}} + 1$ 
7:      $C_{N_{\text{clust}}} = \{x_i\}$ 
8:   else
9:      $C_k = C_k \cup \{x_i\}$ 
10:  end if
11: end for

```

V. RESULTS

We have selected 234 songs from 26 different genres and analyzed k_{opt} , the optimal number of components found by Figueiredo’s algorithm. Then, we compare k_{opt} with estimations of the number of components, obtained via the two methods that have been described in section IV: Correlation between features and correlation between algorithms.

A. Correlation between features

In Figure 3(a) we have plotted the relation between the number of mixture components as found by Figueiredo’s algorithm on the ZCR and SC, k_{ZCR} and k_{SC} . We find the Pearson’s correlation coefficient to be 0.39.

In Figure 3(b) we see the relation between k_{ZCR} and k_{MFCC} , the optimal number of mixture components of the complete 20-dimensional MFCC, as found by Figueiredo. The Pearson’s correlation coefficient is only 0.27.

The correlation coefficients we found are too low to meaningfully predict the number of mixture components of the complete 20-dimensional MFCC vector.

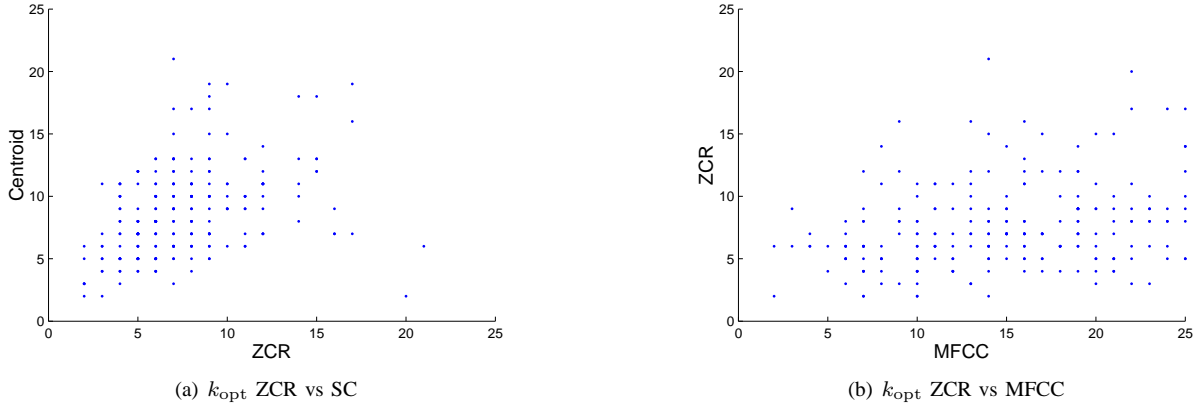


Fig. 3. Correlation of k_{opt} between features

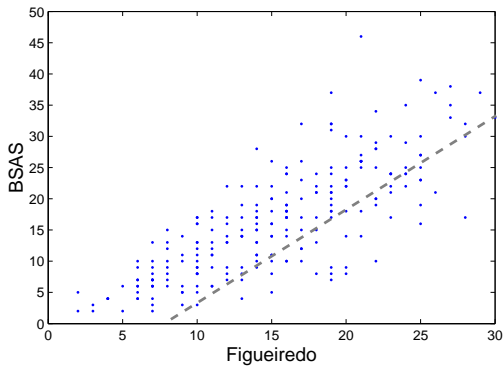


Fig. 4. k_{opt} on MFCC data, found by Figueiredo vs BSAS

B. Correlation between algorithms

We used BSAS on our dataset with an Euclidean distance measure, $\Theta = 4$ and $N_{\text{maxClust}} = 400$. BSAS tends to find a huge amount of very small clusters on outlier frames. When we discard clusters of less than 80 samples (out of 9000), we get the following relationship between k_{opt} as found by Figueiredo and BSAS: The Pearson’s correlation coefficient is 0.78, which is significantly higher than for the feature-based correlation approach. We can approximate the relationship between the number of components found by BSAS and found by Figueiredo as:

$$k_{\text{Figueiredo}} \cong (k_{\text{BSAS}} - 8) \cdot 1.5 \quad (8)$$

This expression overestimates the number of components for most cases. This is deliberate since the loss of information when modelling with less mixture components than required, gives a bigger loss of information than models with a too high number of components.

VI. CONCLUSION

We presented two methods to estimate the required model complexity of individual songs. We found only very weak correlations between different audio features. Especially between

low- and high-dimensional features, correlation is neglectable. Probably, the single dimension as used by the ZCR or SC contains far too little information to accurately predict the number of clusters in a higher dimensional feature space.

Further research on correlations between two- or three-dimensional features and high-dimensional features is needed to explore the possibilities of complexity estimation using simpler features.

The use of simple clustering algorithms, such as BSAS shows much better results. We can approximate the relationship between the number of mixture components as found by BSAS and the conventional ground truth as found by the algorithm of Figueiredo with a linear expression.

REFERENCES

- [1] D. Perrott and R. Gjerdingen, “Scanning the dial: An exploration of factors in the identification of musical style,” in *Proceedings of the 1999 Society for Music Perception and Cognition*, 1999.
- [2] E. D. Scheirer, “Music listening systems,” Ph.D. dissertation, MIT MediaLab Cambridge, 2000.
- [3] J. J. Aucouturier, “Ten experiments on the modelling of polyphonic timbre,” Ph.D. dissertation, Laboratoire d’Informatique de Paris 6, 8 rue du Capitaine Scott, 75015 Paris, France, June 2006.
- [4] E. Wold, T. Blum, D. Keislar, and J. Wheaton, “Content-based classification, search, and retrieval of audio,” *IEEE MultiMedia*, vol. 3, no. 3, September 1996.
- [5] J. J. Aucouturier and F. Pachet, “Improving timbre similarity: How high’s the sky?” *Journal of Negative Results in Speech and Audio Sciences*, January 2004.
- [6] A. Dempster, D. Rubin, and N. Laird, “Maximum likelihood from incomplete data via the em algorithm,” *J.Royal Statistical Soc., Series B (Methodological)*, vol. 1, no. 39, pp. 1–38, 1977.
- [7] F. Mörchen, I. Mierswa, and A. Ultsch, “Understandable models of music collections based on exhaustive feature generation with temporal statistics,” in *Proceedings of the Twelfth CM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, August 2006.
- [8] M. A. T. Figueiredo and A. K. Jain, “Unsupervised learning of finite mixture models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 381–396, March 2002.
- [9] Z. Zivkovic and F. van der Heijden, “Recursive unsupervised learning of finite mixture models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 651–656, 2004.
- [10] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 2nd ed. Elsevier Academic Press, 2003.