Exploratory Datamining in Music

Bjørn Sand Jensen

Kongens Lyngby 2006
IMM-THESIS-2006-49

Errata

General clarification: Notation of various distances may seem a bit confusing, and here is the rationale behind the use (a few corrections below, though). D is used as the global distance, "generalized" from a metric. d is used in case where the distance can be both global and local distance. In chapter 5 this means that the divergence based ground distance is denoted with d. In the general properties of a metric d is used also with the intend to distinguish between various concepts of distances.

Priority: 0 crucial for meaning, 1 important, 2 minor mistake/error, 3 barley worth correcting

Pri.	**Page, Line (- from bottom)**	
	3, 1	Cepstrum -> Ceptral
2	17	Figure, 2.9 The axis is wrong. Should be from 0-10 sec not 1-10 sec.
1	34	Eq. 4.27 parentheses are wrong, θ should be included in the conditional probabilities (i.e. two misplaced right-parentheses)
2	37	Eq. 4.34 missing (t) on the latter **x**
1	37, -7	$p(\mathbf{x},\theta)$ should be either $\log\prod_N p(\mathbf{x}_n \mid \theta)$ or simply L as defined earlier.
2	37,-4	"Tipping: Locally weighted covariance…" should formally be "Tipping: Locally weighted inverse covariance…"
3	38, -16	"…where K->1 where…" should be "…where K->1, …"
	40	Eq. 4.50, clarification. The notation d**s** originates from the original paper from Rattray, but is not a vector as indicated here but a scalar valued incremental distance on the manifold *S* (typo). Not relevant for other metrics.
1	40	4.53 The ∇x should be removed. (typo)
1	42	4.64 $G_*(x_i, x_j)$ should be $\mathbf{G}*$ in terms of earlier use (Tipping). Formally it can also be written as $\mathbf{G}*(\mathbf{x}_i,\mathbf{x}_j)$ (as in original Rattray paper) specifying a constant metric along the path from $\mathbf{x}_i$ to $\mathbf{x}_j$
2	44	Eq. 4.78 J(**x**) should be **J**(**x**)
	49	p(c|**x**) should be p(y|**x**)
0/1	49	T-point equation is wrong. Replace with $D_T(\mathbf{x}_i,\mathbf{x}_j)=\sum_{t=1}^{T} D\left(\mathbf{x}_i+\frac{t-1}{T}\mathbf{v},\mathbf{x}_i+\frac{t}{T}\mathbf{v}\right)$ $\mathbf{v}=\mathbf{x}_j-\mathbf{x}_i$
0/1	50	Graph approximation equation. The equation may seem confusing due to the use of the variable M which has previously been used for the dimension of the space and the use of a small d for the inter-point distance. Replace M by N, and d with D for clarity, i.e.

		$D_{floyd}(\mathbf{x}_i, \mathbf{x}_j) = \min_{N, \mathbf{X} \in \{\mathbf{x}_1', \mathbf{x}_2', \ldots, \mathbf{x}_N'\}} D(\mathbf{x}_i, \mathbf{x}'_i) + \sum_{n=1}^{N} D(\mathbf{x}'_n, \mathbf{x}'_{n+1}) + D(\mathbf{x}'_N, \mathbf{x}_j)$ Where N is the number of points and the original distance $D(\cdot,\cdot)$ can of course be calculated using any approximation available. Note: The graph distances can be found be the use of other algorithms than Floyd, however it is used throughout this thesis.
3	50,-2	"how" should be removed
3	52, -5	"…the d**xF**(**x**)d**x"** should be "…of d**xF**(**x**)d**x"**
3	61	Figure 4.20 caption. p(y|x) should be p(y|**x**)
1	76	"hieratical" -> "hierarchical"
3	81	Subfigure captions. The three digit numbers in the captions should be ignored. Simply used as verification of correct figure insertion.
2	82	"…for CLR, 0.67 for EMD and 0.62 for the…" should be for CLR, 0.67 for EMD-KL and 0.62 for the…"

Abstract

This thesis deals with methods and techniques for music exploration, mainly focussing on the task of music retrieval. This task has become an important part of the modern music society in which music is distributed effectively via for example the Internet. This calls for automatic music retrieval and general machine learning in order to provide organization and navigation abilities.
This Master's Thesis investigates and compares traditional similarity measures for audio retrieval based on density models, namely the Kullback-Leibler divergence, Earth Mover Distance, Cross-Likelihood Ratio and some variations of these are examined. The methods are evaluated on a custom data set, represented by Mel-Frequency Cepstral Coefficients and a pitch estimation. In terms of optimal model complexity and structure, a maximum retrieval rate of ∼74-75% is obtained by the Cross-Likelihood Ratio in song retrieval, and ∼66% in clip retrieval.
An alternative method for music exploration and similarity is introduced based on a local perspective, adaptive metrics and the objective to retain the topology of the original feature space for explorative tasks. The method is defined on the basis of Information Geometry and Riemannian metrics. Three metrics (or distance functions) are investigated, namely an unsupervised locally weighted covariance based metric, an unsupervised log-likelihood based metric and finally a supervised metric formulated in terms of the Fisher Information Matrix. The Fisher Information Matrix is reformulated to capture the change in conditional probability of pre-defined auxiliary information given a distance vector in feature space. The metrics are mainly evaluated in simple clustering applications and finally applied to the music similarity task, providing initial results using such adaptive metrics. The results obtained (max ∼69%) for the supervised metric are in general superior to or comparable with the traditional similarity measures on the clip level depending on the model complexity.

Keywords: Music Similarity & Retrieval, Audio Features, Clustering, Classification, Learning Metric, Information Geometry, Fisher Information Matrix, Supervised Gaussian Mixture Model.

Resumé (Danish)

Dette eksamensprojekt omhandler metoder og teknikker til musikanalyse, med hovedfokus på musiksøgning. En sådan opgave er blevet en vigtig del af det moderne musiksamfund, hvor musik distribueres effektivt via for eksempel Internettet. Det kræver kræver automatisk søgning og såkaldt datamining for organiserings- og navigeringsformål.
Eksamensprojektet undersøger og sammenligner traditionelle similaritetsmål for audiosøgning baseret på sandsynlighedsmodeller, og Kullback-Leibler Divergens, Earth Mover Distance, Cross-Likelihood Ratio og enkelte variationer af disse. Metoderne er evalueret på et specialdesignet datasæt, beskrevet ved Mel-Frekvens Cepstral Koefficienter og et pitch estimat. Ved optimal model kompleksitet og struktur opnås en maksimal søgningsrate på ∼74-75% for Cross-Likelihood Ratio ved søgning på sange og ∼66% for søgning på klip.
En alternativ metode til musiksøgning og datamining introduceres, baseret på et lokalt perspektiv og adaptive metrikker, med det formål at bevare topologien af det originale featurerum for explorative formål. Metoden er defineret på baggrund af Informations Geometri og Riemannian metrikker. Tre metrikker er defineret, en unsupervised vægtet kovarians matrix baseret metrik, en unsupervised log-likelihood baseret metrik, og endelig en supervised metrik formuleret på basis af Fishers Informations Matrix. Fishers Informations Matrix er omformuleret til at afspejle ændringer i den konditionelle sandsynlighed for pre-defineret auxiliary information givet en afstandsvektor i feature-rummet. Metrikkerne er hovedsagligt evalueret i simple cluster-applikationer og endeligt anvendt i musiksøgning, hvilket giver initiale resultater ved brug af sådanne adaptive metrikker i musik. Resultaterne ved brug af en supervised metrik (maksimalt ∼69%) er generelt bedre eller som minimum sammenlignelige med de traditionelle similaritetsmål ved søgning på musikklip afhængig af modelkompleksitet.

Preface

This Master Thesis is submitted as partial fulfilment for the Master of Science degree in Engineering at the Technical University of Denmark (DTU), Kongens Lyngby, Denmark. The work leading to this Master Thesis has been conducted in the Department of Informatics and Mathematical Modelling (IMM), DTU.

The author of the thesis is Bjørn Sand Jensen (s001416).

Main supervisor is Professor Lars Kai Hansen, Department of Informatics and Mathematical Modelling, DTU. Co-supervisor is post.doc. Tue Lehn-Schiøler (PhD), IMM.

Bjørn Sand Jensen
April 26, 2006

Contents

Chapter 1

Introduction

The Sound of the Information Society

The amount of data collected in today's knowledge based society is tremendous. The data spans from food recipes, brain scans to music and even complete books. The digitalization of information is the main reason, since the information is compressed in a very convenient and often distributed way.

In the good old days information was kept in paper books - novels, financial accounts etc. - which naturally implied a limit to the degree of details in the information, since every entry in for example an financial account, would have to be entered manually. It also meant that the amount of information available was limited and therefore the task of getting an overview of the data presented, would be a relatively easy task (of course with some exceptions). With the digitization and an creating of the computer age, has the amount of detailed data become enormous, and every little detail about, for example, a financial transaction is saved for later potential retrieval.

Datamining

The availability of information or data is, of course, to some degree a very appealing thought. However, what happens when you cannot find structure and overview in the data? This could be due to some very complex structure in a small amount of data - but it could also be because of the huge amount of data presented. This basically means that the information is more or less useless in the complete form. The intuitive solution would be to split the data up into smaller chunks and analyze it; however doing so might mean destroying some important structural information in the data.

...in a music database

Music plays an important role in the everyday life for many people, and with the digitalization, music has a prime example of huge data collections and is basically available anytime and everywhere. This has lead to music collections - not on the shelf in form of vinyl records and CD's - but on the hard drive and on the internet, to grow beyond what previously was physical possible.

It has become impossible for humans to keep track of music and the relations between songs and pieces, and this fact naturally calls for datamining and machine learning techniques to assist in the navigation within the music world. The objective of the thesis is first of all to explore methods of performing such datamining in music databases.

Traditionally, datamining comes with a rather large toolbox often involving methods for tasks such as classification, regression and clustering, but one common thing is the problem of representing the data at hand. In case of a music database this data can be many things; the music itself, metadata such as the title of the songs or even statistics of how many people have listened to a track.

This thesis will be limited to the music itself which will be represented in terms of suitable low-level features (like the cepstral coefficients). This essentially means two different tasks at hand: a feature extraction including the database creation, and a datamining part. While the feature extraction is primarily based on traditional signal processing, the datamining is a part of the area known as machine learning.

This involves statistical modelling and - in popular terms - some sort of artificial intelligence. The purpose is to discover patterns and hidden links between the data available. Although the pattern discovery might seem trivial for humans when dealing with certain (often limited amounts of) music, machine learning techniques has yet to get the final breakthrough in the machine/computer world when dealing with music. One main reason is that music - and in particular music perception - is a quite complex subject, e.g. just think of the potential difficult task of classifying a given song into one single genre. This problem is often referred to as a lack of ground-truth, which implies that there may not exits a real way of performing certain tasks, such as genre classification in which a hard genre taxonomy of music is assumed to exist.

Project description

The purpose of this thesis is to take an other approach than the traditional hard classification way to audio exploration, and focus on a more *explorative* approach. The focus will be on individual songs or even clips using a custom data set in order to evaluate the methods applied on a more solid ground-truth than e.g. an overall genre level.

The fuzzy term *explorative* used in the title of this project can be quite broad, and here it is linked to an intrinsic problem in music datamining: *when do two songs sound alike*? The human brain is for some reason "designed" to pick up on such similarities between individual tracks - or at least be trained to do so. This ability to give some sort of evaluation of the similarity is in essence what this thesis is all about. Machine learning and datamining techniques applied so far are often based on a density estimation in the so-called timbre space

(see chapter 2 and 5) of each individual track. Various methods have then been suggested in order to compare these density models, ranging from divergence based measures (e.g. Kullback-Leibler divergence) or estimation of the cross-likelihood ratio based on sampling (see further discussion in chapter 5).

In this thesis, these ideas will be examined, both in terms of model complexity and training, which has been noted to be a general issue with these methods in previous evaluations. Furthermore will an alternative direction in music exploration be explored based on a distance in a geometric space, hence similar to the well-known K-Nearest-Neighbour family. However, a density model will still be maintained to account for complex data relations, but now in a global sense. Both an unsupervised approach and a supervised approach is investigated in order to evaluate the effect of manually guiding the extraction of the distance between e.g. two clips.

The new distance/similarity functions - also called metrics - are based on the concept of Riemannian geometry in which such (local) metric can be generalized to the entire feature space, providing a distance or similarity measure quite different from the well-know Euclidian or Mahalanobis distance. The properties of these metrics will be evaluated through various artificial examples and a real-world data set, in order to show the various benefits and disadvantages of such an approach, including some approximations to their true formulation.

A special data set is constructed for the evaluation of the various techniques, described in chapter 2. Although custom, the purpose is not to do a subjective experiment, and only simple relationships between the tracks are considered based on associations such as artist.

Potential Applications

In relation to a music database the similarity function can be exploited in some very simple, such as K-Nearest-Neighbour methods, and can provide an adaptive metric for various tasks in music exploration and analysis.

It is the aim that the results - good or bad - can be used for the development and research into a music search and exploration application. The current thesis deals, as already mentioned, with the task of finding similar subjects in feature space, and will therefore contribute to a kind of browser function where an user can ask the million dollar question: *give me something that sounds the same!*.

Roadmap

This report, describing the work carried out in the project period, is organized in the following way.

Chapter 2 An introduction to the basic properties of music and the considerations made about features. Furthermore the algorithms for features extraction will shortly be described including a perceptual multipitch estimation algorithm for extracting the two predominant fundamental frequencies (pitch), and the extraction of Mel-Frequency Cepstrum Coefficients (MFCC).

Chapter 3 A short description of the custom data set constructed for the evaluation of similarity, mainly on clip level, including a visualization of the data.

Chapter 4 A methodology chapter describing the learning algorithms considered, including a description of the Expectation-Maximization algorithm for both unsupervised and supervised purposes, and a discussion of the practical approaches taken for overcoming overfitting in the music data set.

The formulation and derivation of metric based learning, formulated on the theory of Riemannian geometry. A relatively detailed insight into the properties and approximations is provided, including experiments on various data sets, mainly performed through K-means clustering.

Chapter 5 A chapter describing the similarity measures used. The traditional techniques are described in detail, including description of simple Kullback-Leibler based methods, Cross-Likelihood Ratio and the Earth Mover Distance.

Various considerations concerning the use of the metric learning principle in music in described, and a simple suggestion of how to apply the geometric metrics in practice is described.

Chapter 6 Providing results on the custom data set for the distribution based methods for comparison. Includes a number of variations on the Earth Mover Distance compared to previous reported results in music retrieval, including suggestions for using a BIC-based model selection on a song level.

Providing initial, limited results using geometric measures based on both unsupervised and supervised assumptions in audio set, through evaluation of the retrieval abilities of the metrics using a rough vector quantization approach.

Chapter 7 Summery, Conclusion and a suggestions for improvements and further work.

Chapter 2

Music - Basic Properties

This chapter reviews some of the basic properties of music in order to motivate the choice of features, and provide motivation for the task of similarity estimation and exploratory datamining in music, based on the local meaning of the features.

Music is physically speaking changes in sound pressure, which is detected by the ear and perceptual system for further processing further on in the auditory pathway. However, in the mathematical sense the music can be described conveniently by a one-dimensional time varying signal like shown in 2.1

In order to analyze the actual musical contents, the spectrum is often extracted using the Fast Fourier Transform to show the contents in the frequency domain. In order to extend this with temporal information, the spectrogram shows the changes in frequency content over time.

The spectrogram shows all the details in frequency and time domain resulting from various instruments, like a noisy guitar, singing voices etc., and each music piece or song, of course, has its own signature in such a spectrogram. The spectrogram does provide a more or less complete description of the music, including information not relevant to the actual task of comparing e.g. different songs, and does furthermore only contain purely physical or even mathematical attributes, hence not describing how the sounds are perceived and processed by the listener.

In order to provide more practical and perceptual description in form of so-called features, assumptions are often made about the perception of sounds - a fairly short review of relevant properties of the human perceptual understanding of music is included for completeness and motivation.

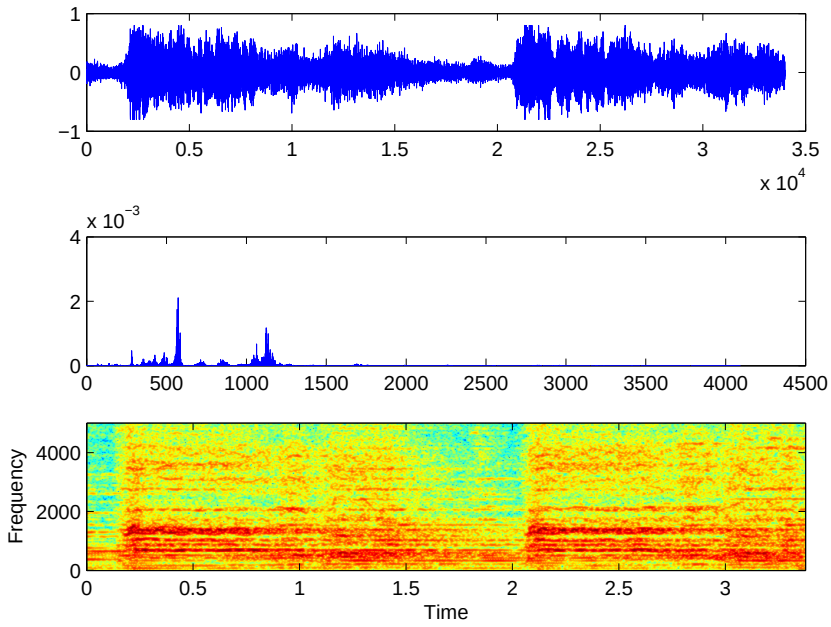


Figure 2.1: A music signal and analysis options. Top shows the raw time domain signal. The plot shows the spectrum, as magnitude vs. frequency (Hz). The bottom plots shows the spectrogram.

2.1 Music Perception

In the human, subjective understanding of audio three different concepts are traditionally found fundamentally important: pitch, loudness and timbre. The three concepts originates from the perception of tones, i.e. not complete polyphonic music, and all of these have undergone extensive research (overview in e.g. [9]). One more, perhaps, underestimated attribute of audio in this context is temporal and structural information, like beat, rhythm, and melody - which is omitted in this thesis, though.

2.1.1 Pitch

In spectral analysis, a fundamental frequency is often refereed to as the lowest (frequency wise) component of harmonically related spectral components. In case of a musical signal, this fundamental frequency will in some cases be referred to as the pitch. There is, however, one catch: the human pitch perception is not as simple as initially implied by the definition of a so-called fundamental frequency.

While fundamental frequency is a deterministic, physical attribute of a audio clip - often extracted from the spectrum - pitch is a psychological phenomena, which is an extremely complex perceptive and cognitive process. For example humans can perceive pitch, a so-called virtual pitch, even though the fundamental component is not physically present [9, 22]. If for example listening to the notes C_0, C_1, G_1, E_2, G_2 added one by one, a removal of C_0 will not have any noticeable influence on the perceived pitch. And the same goes for C_1 and to a lesser extend G_1.

Various theories and models describing human pitch perception has been suggested, but not one which can account for all reported experiments. Often a compromise will have to be made in the model applied and the assumptions made, of which one such model will be mentioned later, when considering a automatic pitch extraction algorithm.

An interesting concept in pitch theory is / at least in the western music - the composition of music based on the octave system, in which an so-called octave is divided into twelve tones/semitones like depicted in figure 2.2.

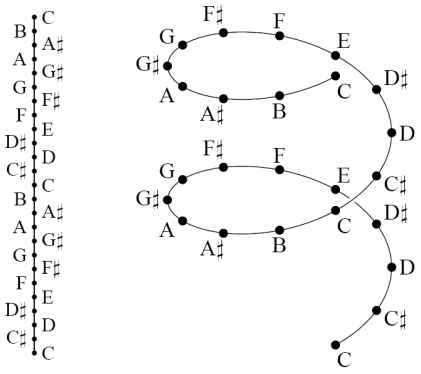


Figure 2.2: The concept of pitch as a scale (logarithmic) and as a helix, which illustrates the notion of pitch "height". From [9]

Whether this geometric system of pitch structure is orthogonal, i.e. a simple translation of the musical piece up an octave gives the same perceptual result is not conclusive [9, p. 375], which can be proven with some fairly clever paradoxes (see e.g. [9, p 376]). In machine learning such a translation could involve reducing a potential pitch description to a pitch class, which is often referred to as tonality.

Critical Bandwidth Analysis

Humans have an (possible learned) ability to recognize the first 6-7 harmonics of a fundamental tone (single sine), however music and almost all other sounds are complex mixture of different tones. In order to understand the pitch/frequency analysis part of the human system, Fletcher as one of the first, did a number of test, in which a pure tone was mixed with a band-limited white noise signal [22]. The pure tone amplitude was decreased until the listener could not hear the tone. The noise-bandwidth was then decreased. The conclusion was that a decrease in bandwidth (and thereby noise power) did not influence the perception until a critical width. The experiment was then repeated for a number of frequencies and the conclusion was that the critical bandwidth increased logarithmically with the increase of the pure tone frequency (center frequency).

This observation has been extended and researched rigorously, and can also be explained by the use of two pure tones played simultaneously (see e.g. [13, p. 74-79]). If these tones are closely spaced in frequency (between 0.05-1/CB), i.e. within the critical bandwidth the tones will be perceived as been rough combination of tones (also described as dissonance) and when very close (<0.05/CB) a kind of beat is perceived (consonance). However, if these tones are separated by more than the critical bandwidth the result is perceived as two

separate tones and gives as smoother sound (consonance).

This effectively means that a perceptual filter is imposed on the signal, described by the width of the critical frequency, which is fairly accurate when considering single pure tones. Several computational models of this filtering-like operation has been constructed, and the actual shape of the filters will ultimately depend on the application in which they are used. In this thesis two variation of such filters will be applied, although for different purposes.

2.1.2 Loudness

When comparing two musical pieces the perceived loudness may have an profound influence, however, the sensation of sounds are often dependent on the specific environment in which the perceived sound is experienced.

An fundamental result, is the fact that the sensation (of pressure, loudness etc.) increase logarithmical as the stimulus is increased. This is a well know experimental result, which has been proven by several results [9, p. 99] - although often based on the idea of applying a single tone as stimuli.

The absolute loudness perceived is very difficult to incorporate, since music is experienced in a unlimited number of psychical situations, from a concert hall to elevator muzak. Therefore this kind of absolute loudness description is rather impossible to include in the specific context. There are however some psychological features, which could potentially be used, namely the so-called sonogram, based on the sone scale. It gives a measure of how loudness is perceived based on the energy of the signal.

In this thesis loudness will not be considered directly, however since the loudness is very much depended on the energy in the signal, a energy measure will be included based on the feature extraction of the timbre, described in the next section. Such a measure is definitely not a perceptual motivated feature, but simply describes the overall energy of the signal (on a short-time basis though).

2.1.3 Timbre

Timbre is a somewhat fuzzy concept and is often defined by what it is not:

> "Timbre is that attribute of auditory sensation in terms of which a listener can judge that two sounds similarly presented and having the same loudness and pitch are dissimilar" (American National Standards Institute, 1960).

Timbre is also said to be the quality of the sound, and can in terms of the definition be seen as the discriminating factor between two instruments playing a tone with the same pitch and loudness, i.e. it identifies the source of the sound. This initially sounds ideal if we want to be able to find a similarity between music, however the construction of a timbre feature is perhaps not as simple as first implied.

Timbre description has undergone extensive research, not at least in the production of electronic sounds, since the quality of the electronic/digital reproduction of instruments

depends heavily on the timbre similarity between the true tone and artificial created version. This has lead to different ways of analyzing and viewing timbre, which relates directly to the feature extraction and in some sense datamining part, which will be evident later. Furthermore timbre description has been the natural basis for music similarity applications, which will be reviewed in chapter 5.

A spectral view: Timbre is often viewed as the spectral difference between instruments (with same pitch and loudness), and does in some sense give the *feeling* of the music or instrument based on the frequency contents.

Spectral analysis of timbre is often the predominant analysis technique used when considering the timbre attribute, but this approach has, however, also been shown to lack some properties. One of the assumptions made, is in regards to the periodicity of the musical tone/sound, relying only on the relative amplitude of frequency components in the spectrum analysis, thereby ignoring the temporal development of the tones. However, a musical tone is often thought of as consisting of the attach/onset, steady state and the decay, and it has been shown that the attach of an instrument contributes greatly to the human perception of the resulting sound (see e.g. [9, 13] and hence contributing to the timbre concept. However, these aspects are not included in a basic spectral viewpoint.

Another critical point is the ability to recognize instruments even though the recording has been altered (filtered) by e.g. a rooms acoustics. This illustrates that the spectrum may not be the sole contributor to the timbre, leaving a gap to be filled in order to fully understand the workings behind timbre.

Multidimensional scaling: The work performed by e.g. Grey (1977) [11] on multidimensional scaling (MDS) applies a very subjective approach to timbre analysis and similarity in order to understand the factors contributing to the perception. Based on various experiments in which pitch, loudness and duration was constant various sounds were presented and listeners were asked to describe the similarity. Grey then used so-called multi-dimensional scaling with three dimensions in order to illustrate the difference between sounds. In this case the similarity described by the listeners was interpretable against three physical attributes: spectral energy distribution, transient synchrony spectral fluctuation and low-amplitude, high frequency energy. Such a subjective evaluation is probably the only true indication of similarity, but does lack a generalization ability in the sense that humans often perceive sound and music differently.

This thesis deals with a complex mixture of tones, instruments and human singing and improvisation in a machine learning application, where the analysis of each sound is sought performed automatically. In such a setting, is the example of multidimensional scaling by a subjective evaluation not an realistic option, which implies that the timbre description in this thesis will be based on spectral properties as described in the following paragraph. However, the principle behind multidimensional scaling of sounds are very much relevant, since it is in essence what we are trying to do automatically by the use of a similarity function defined in the feature space.

2.2 Features

In this thesis the representation of the musical signal will be based on the observations described in the section above concerning auditory perception. By doing so, we often throw some information away present in the original signal, and it is obviously crucial that the most important information is retained in some manner.

Only a few sets of short-time features will be included, however these include a description of the pitch and timbre of the music. In the setting of finding similarity, this is believed to be a workable starting point. It is hereby indicated that features based on temporal information such as, beat, rhythm and overall structure as been left out in this project.

Some of the simplest features are the purely statistical ones, which is based on the various low order statistical moments, like the mean and variance. It often includes the well-known zero-crossing rate (ZCR), root-mean-square (RMS) level, Spectral centroid, Bandwidth, Spectral roll-off frequency, Band energy ratio, Delta spectrum magnitude etc. While such statistical features may provide exceptional information for a pure classification application, they have been omitted in this project, mainly due to the focus on similarity measures based on perceptually motivated features.

2.2.1 Windowing

The signals considered here, are all audio signals, however audio signals are only considered stable for a short period of time, which supports a short-time window for the extraction.

In order to calculate the features with this stability property in mind, the signal is divided into overlapping frames of 20 ms. However, this truncation of the signal, does not comply with the periodical assumption made by the fourier transform. In order to limit this truncation effect a filter with attenuating side-lobes is applied and in this thesis a Hamming window is used.

2.2.2 Pitch

Pitch is, as described, a fundamental property of music and perception, which have motivated the selection of this feature to be included.

Most pitch estimators have been developed for speech signal in which a single speaker is present (see e.g. M. Slaney [11]). Speech is often considered having one fundamental frequency - music on the other hand is mixture of instruments potentially playing different chords on different instruments etc.

Generally pitch is not easily extracted automatically in complex sounds with several instruments, harmonics and pitches, but recently Klapuri [16] has suggested a pitch estimator directly aimed at music applications, in which results of estimating two pitches (using a 92.8 ms window) vary from approx. 2-8% percent for a true two pitched signal.

Correlation Based Method

In music signals the most predominant pitch estimations method is based on the autocorrelation principle (see e.g. [11]), in which the outputs of a filterbank are autocorrelated, as illustrated in figure 2.3

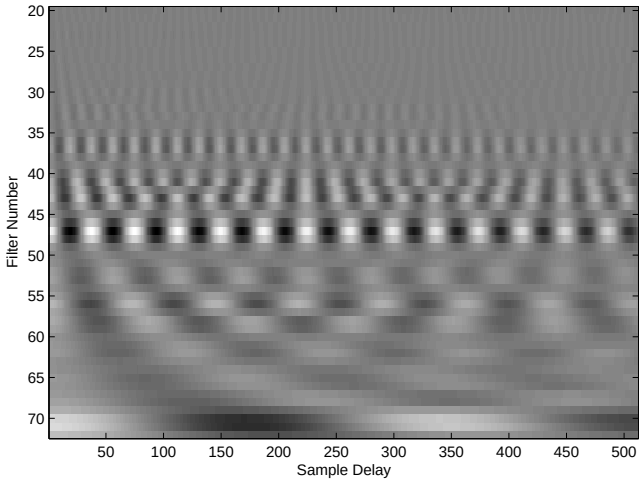


Figure 2.3: Autocorrelation of individual subbands of Bachs' Clavier Concerto in F minor. Illustrated with the same auditory filter used in the method by Klapuri

While the autocorrelation provides information of all periodicals, recent techniques by Klapuri is able to extract estimations of the individual pitches. Based on initial trials and the reported results in [16], this method has been adopted for the description of pitch in the similarity experiments.

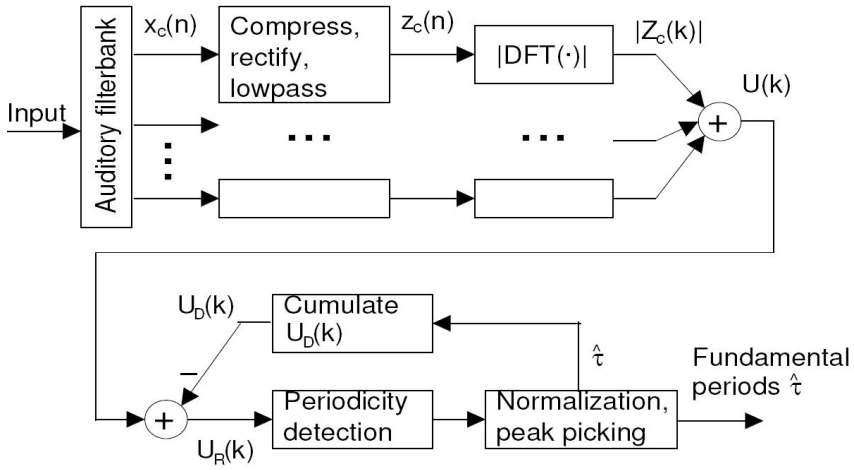


Figure 2.4: Multipitch estimation method overview. From [16, p. 292]

Auditory Model applied

The pitch model behind the extraction is somewhat more accurate than the one usually applied in e.g. the extraction of the Mel-Frequency Cepstral Coefficients (MFCC) described later. However, due to the objective of this thesis and the obviously important modelling of

pitch perception, when directly estimating it; parameters such as filters, inner-ear compression etc. have not been changed compared to the original suggestion in [16].

The first step in the pitch extraction is a filter-bank based on the principles of critical bandwidth described in section 2.1.1. The filters suggested are based on the *gammatone* filter (see e.g. [9, 22, 16]). Due to the nature of pitch, i.e. the fact that higher order components gets more and more spurious, and the more important fact that the human phase-locking seems to break down at about 5 kHz ([9]), the highest filter frequency has a center frequency at 5.2 kHz.

The filterbank provides the possibility to perform subband analysis, which is done by noting the auditory functioning in which the mechanical vibrations in the basilar membrane are transformed into a neural transducing. This is modeled by a compression, half-wave rectifying and low-pass filtering (for details see [16]).

Periodical analysis The output of the auditory model is transformed into the frequency domain in order to perform the needed periodic analysis, where the real difference between Klapuri's estimator and other's work is found. The chosen method applies an iterative approach developed through several experiments and papers. An overview is illustrated in 2.4. The basic idea is to locate the harmonics of the currently, predominant pitch and simply cancel this estimate in the correlation.

The periodicity analysis is furthermore custom designed for the purpose of finding the harmonic shapes through the use of the short-time inverse DFT and a specially shaped filter function - however given the purpose of this thesis the details has been left out (see [16] for further details).

The performance of the pitch estimator has only been carried out empirically on a small test signals similar to the one illustrated in figure 2.5, and a number of smaller audio signals. A real audio example is shown in figure 2.9. In short we rely on the quite promising results reported in [16] to hold for this purpose, however the exact performance of i.e. an individual window is not considered crucial in this work, since we are mainly interested in the distribution of the estimation , which may very well change from e.g. song to song despite the absolute value of the pitch. This is illustrated in chapter 3 plotting the pitch distributions of the genres.

2.2.3 Cepstrum analysis

The core idea in so-called cepstrum analysis applied to music is a smoothed spectral representation. However the cepstrum analysis has first and foremost been a primary tool in speech processing, in which a model is assumed consisting of slow varying part of the speech due to the vocal tract, $v(n)$, and a fast varying part due to the excitation signal, $e(n)$ of of the vocal tract, i.e. leading to a convolution in the time domain.

$$x\left(n\right) = e\left(n\right) * v\left(n\right) \tag{2.1}$$

The motivation for the use of the cepstrum in speech analysis is a desire to separate these signals, which is done by a number of operations. First the power spectrum is found formally using the discrete fourier transform (DFT)

$$|X(\omega)| = |E(\omega)|\,|V(\omega)| \tag{2.2}$$

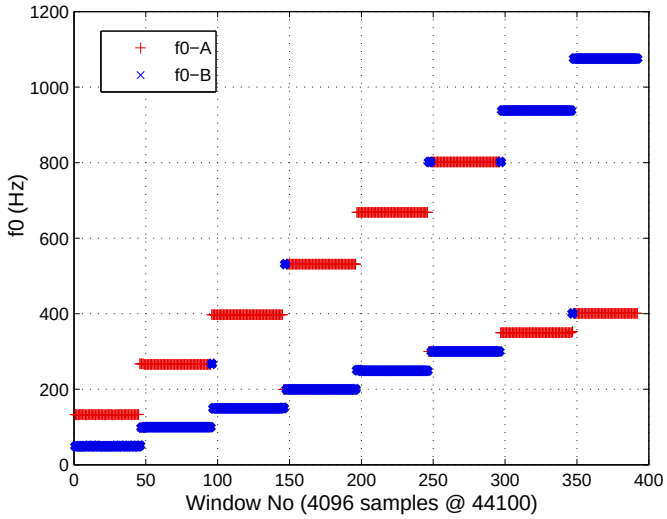


Figure 2.5: Simple test of the multi-pitch estimator for two harmonics (50 and 133 Hz) including their 4 overtones. The frequency is increased by adding the original fundamental in each step.

Then taking the logarithm to the power spectrum yields an additive result

$$\log\left(|X(\omega)|\right) = \log\left(|E(\omega)|\right) + \log\left(|V(\omega)|\right) \tag{2.3}$$

The so-called cepstral coefficients are then found using the inverse DFT

$$c(n) = \frac{1}{2n} \int_{-\pi}^{\pi} \log|X(\omega)|\, e^{j\omega n} \tag{2.4}$$

The principle of the cepstrum approach is illustrated in figure 2.6 and it is seen that it is possible to separate the excitation signal and the vocal contribution by a filtering in the cepstrum domain.

Despite cepstrum analysis being formulated in terms of speech signals, it is highly applicant to musical signals, in which the smoothed spectral representation can be used in the similarity estimation considered in this project.

Mel-scale: Making perceptual features

Cepstrum analysis provides a smoothed spectral representation, but it does not really provide any features in which the auditory models are included directly.

A popular approach to this task is the use of the critical band filters previously mentioned, which in terms of cepstrum analysis, was done originally by the use of the so-called mel scale (1 mel = 1000 Hz). This will emulate the single tone pitch perception by transforming the power spectrum of the signal into the mel-scale (or sometimes Bark-scale). I.e. a number of filters N are defined with a center frequency according to some definition of the critical

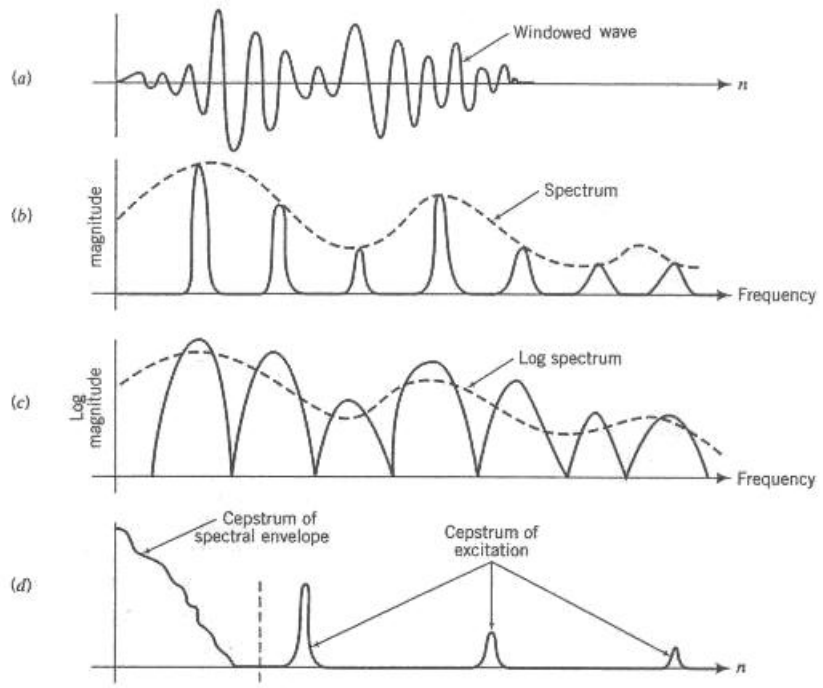


Figure 2.6: The principle of Cepstrum analysis. From [22] (adapted slightly).

bandwidth in a given frequency region. The energy of the signal around this center frequency is then included when filtering the spectrum.

The frequency transformation is done by a filter bank, however, there is no real consensuses on the optimal definition of these filters. Various filter banks have be proposed in the music retrieval and similarity estimation community (see e.g. [5, 3], but the overall structure is the same: in the low frequency band a equally set of spaced relative narrow filters is placed. From about 1 kHz, a set of logarithmical spaced filters is introduced in order to include a rough description of the pitch (pure tone) perception.

The filters in this project are constructed using linearly-spaced filters below 1 kHz (133.33Hz between center frequencies,) followed by log-spaced filters (separated by a factor of 1.0711703 in frequency[1]) as defined by Malcom Slaney (see e.g. [28]). The total log-energy in each band is furthermore kept constant, providing a logarithmic decreasing in filter magnitude. An example of this structure is illustrated in figure 2.7.[2]

The overall MFCC extraction is illustrated in figure 2.8

[1]The initially weird factor comes from the goal of going from 1kHz to 6.4 kHz in 27 steps [28]

[2]The Mel-Frequency Cepstral coefficients are calculated using the toolbox provided by Dan Ellis, containing a wide variety of various filter proposals

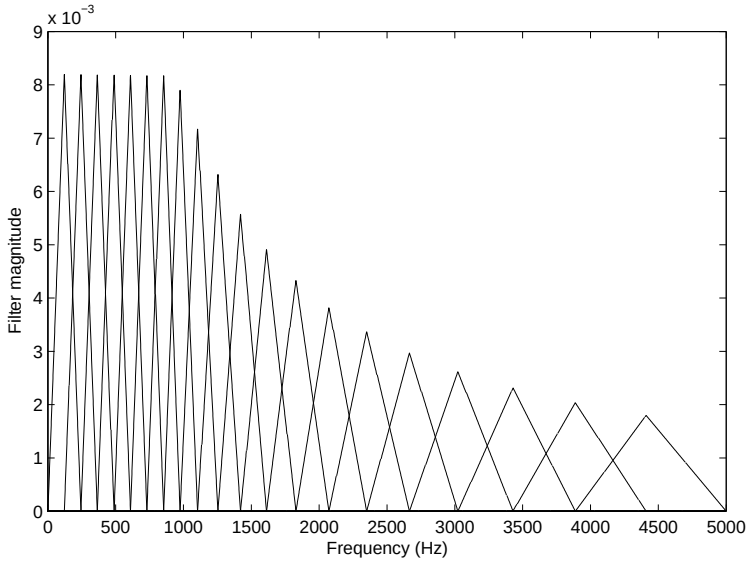


Figure 2.7: Filterbank for mel frequency transformation of the input signal. For illustration purposes a 20 filter example is shown given a sampling rate of 10 kHz. The real data set considered is sampled at 44.1 kHz and 40 filters will be used, in order to provide reasonable resolution of the filters

Dynamic features

An quite important extension of the basic short time MFCC's is the inclusion of dynamic information in the form of the delta coefficients given by

$$\Delta c_i(n) = \frac{\sum_{k=-N}^{N} k c_i(n+k)}{\sum_{k=-N}^{N} k^2} \tag{2.5}$$

Which is essentially a correlation between a straight line and the different coefficients. Although mentioned due to their importance, the delta coefficients will not be applied in this project, since the main objective is a basic comparison of methods.

2.2.4 Temporal features & Feature Integration on a short time basis

Only short time features will be investigated in this thesis in term of the similarity objective, however temporal features such as tempo, beat and rhythm may very well be important properties when considering the similarity of songs.

In stead of extracting individual descriptors of temporal information, a concept known as feature integration can be applied to the short-time features described above to provide temporal information of these. An interesting representation does also fall into this group namely the Auto Regressive representation (AR) which originates from time-series analysis. AR models is a stochastic model fitted to the given signal, i.e. the AR model can be applied directly to the given signal (maybe windowed) or applied to other windowed features like the well known MFCC's (see e.g. [20]) in order to account for the dynamic long-term behavior. But such an integration has also been left out.

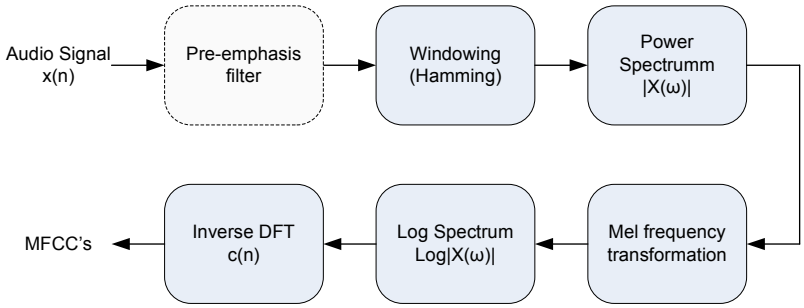


Figure 2.8: MFCC feature extraction. The pre-emphasis filter is usually used to emphasize high.frequency contents, however this option has not be applied in this project.

2.3 Summary & Choice of features

This section included a short review of some of the properties of music, which serves as motivation for the overall task of finding similarity in music. A few important properties, namely timbre and pitch, where singled out as the two properties to examined in this thesis. Based on this choice, a feature set consisting of the 8 first MFFC's - including the 0th as a measure of shot-time log-energy - and the two dominant fundamentals. The feature set has been limited for the purpose of showing the properties of the measures and provide some further insight - not into the very best obtainable - but into the difference between techniques for music similarity.

The MFCC has been shown to provided a reliable retrieval rate in other similarity projects (see e.g. [3, 2, 5]) focusing mainly on timbre similarity. In this thesis, a description of the pitch was suggested based on a multi-pitch detector in order to extend the similarity examination from timbre space with another perceptual motivated feature. The pitch detector was chosen based on promising results provided in [16], although no extensive testing was performed. The inclusion of such an feature should be seen as "just" another - possible great - feature, as the motivation is mainly based to the investigation of similarity functions. The pitch has an intrinsic property of being discrete in nature (see e.g. figure 2.9), which will later prove quite challenging for the classic similarity methods presented in chapter 5.

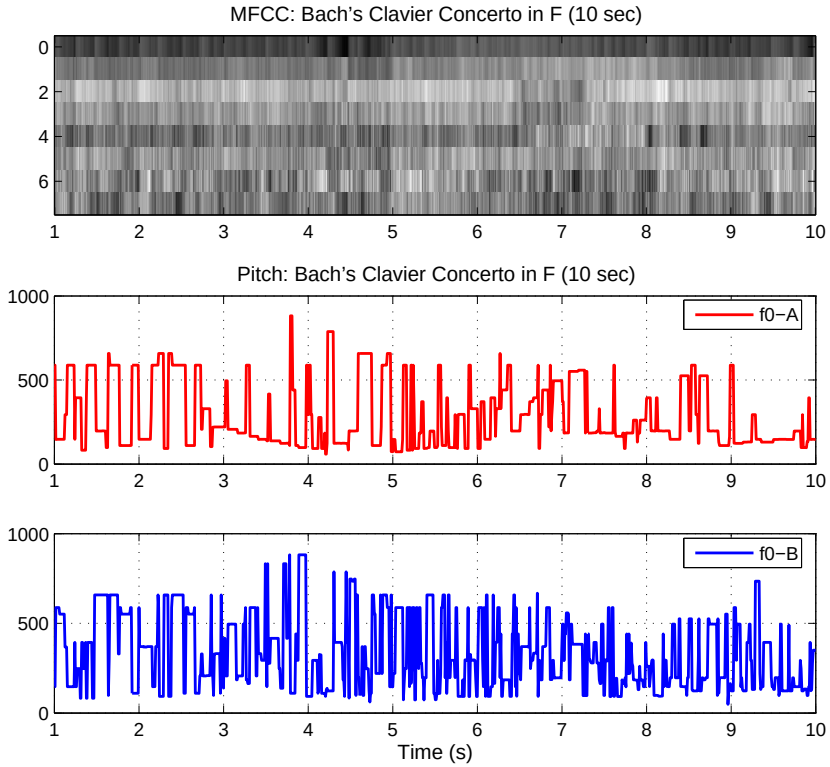


Figure 2.9: Illustration of the MFFC (including the 0th coefficient, as a log-energy measure) and Pitch (two fundamentals) feature set used in the experiments. Notice the discrete nature of the pitch. This is potentially a problem for the density model used to model the distribution

Chapter 3

Music Dataset

This chapter describes the custom data set used in the analysis of the various techniques described later. The raw data is obtained from mp3-encoded music files sampled at 44.1 kHz, and is, after feature extraction, represented by the feature combination described above (i.e. 8 MFCCs, incl. the 0th and two fundamentals).

The data set used in this thesis is inspired by a smaller data set used currently in the Intelligent Sound Project at The Technical University of Denmark, and is based on the ability to defined a ground-truth, i.e. define what is similar.

The data set is constructed on the main assumption that the hierarchy consisting of, Genre $->$ Artist $->$ Track $->$ Clip, is obeyed, and i.e. no artist can produce music in another genre. While this is obviously not true in general, the data set has been created with this in mind. A 1000 clips (of 10 sec.) data set with 10 clips per song and 2 songs per artist, i.e. 100 tracks, is constructed. The small data set is in contrast to the actual task of mining in often large databases, however through a proper training and selection of models it will give some hint of the generalization abilities of the techniques and first and foremost provide an solid base for showing the properties of the various similarity measures and techniques applied.

The tracks are represented by a 100 second continuous interval, divided into 10 clips of 10 seconds[1]. The features will be calculated using a 20 ms window with 10 ms overlap for the MFCC extraction and a 92.8 ms (4096 samples) window for pitch estimation (based on the results in [16]) .

The data set consist of five genres, and while the ground truth of genre classification is not obvious, this data is considered adequate in terms of describing the characteristics of each particular group.

[1]The actual splitting of the tracks is performed to account for any required overlap in the feature extraction part of the system

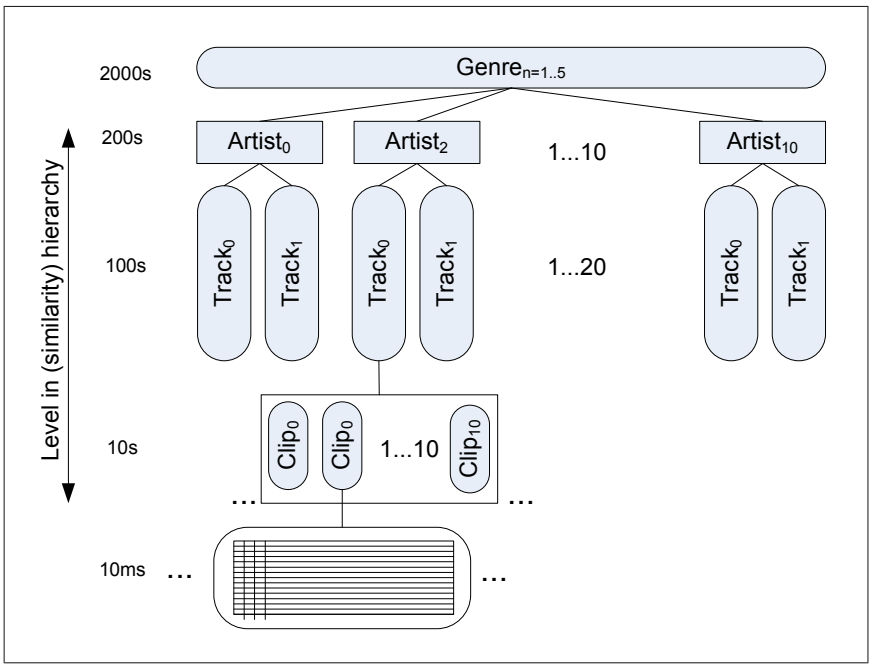


Figure 3.1: The hierarchy of the data set, including the included track time per item (i.e. 10 sec./track

The genres are shortly describe for completeness:

Classical: Classical music covers a large time span in music history, and one main feature is the lack of vocal (not considering opera), which does give a good separation in timbre space (see e.g. figure H.2). Further is classical music often described by the use of a limited number of classical instruments, leading to an assumption that the pitch is fairy stable, and hence may offer a distinct distribution of the fundamental values.

Pop: Turn on the Radio and with a very high probability you will listen to a so-called pop track. Pop is a abbreviation of "popular music", and every human raised in the western world do have an very good idea of what pop music is - but it does to some degree vary on the century.

Over the years, and especially by the late 1980's and 1990's the concept of pop music have gotten its own meaning, which is very hard to describe in words, but in terms of variation, the pop genre contains a large variation of instrumentation, vocal and other general properties. Such a variation may introduce problem sin defining the ground truth in this area of machine learning which quite difficult to obtain - perhaps impossible for the so-called pop music. Despite this negative observation, a pop set has been adopted (and adapted) which does seem to describe the current state of pop music, obviously obtained trough half a decade.

The data set consist of 20 tracks from the 1980's, the 1990's and 2000's and seem to describe the paradox of pop music: it can contain everything from semi-hard rock like Coldplay and U2 to Robbie Williams and Madonna.

In terms of the features used, this variation in both style and instrumentation leads to an interesting genre, which will be used to show the abilities of certain traditional and new similarity techniques described in chapter 4 and 5.

HardRock (Heavy metal): Rock music is a very wide concept, but due to the extend of this thesis a special subgenre of Rock as been included, namely Heavy Metal. However while some subgenres are hard to define by common man, heavy metal is often quite distinct in form of its noise high energy sound, which in terms of features this means a high level of energy in a wise area of bands, although this effect is limited due to the simplification of using only 8 MFFCs.

Electronic (Trance) : The history of the electronic music is not as defined as e.g. the classical music, since it is a fairly new genre. However some of the sub-genres do share some common grounds. Electronic music have a large number of sub-genres like, dance,trance, hip-hop and even new age and in the present music culture it does affect the so-called pop music in some manner.

Electronic music is obviously a wide concept, and in order to keep things relatively simple, this thesis will only include one of the more distinct subgenres and perhaps the one that differs the most from pop-dance music, namely: Trance.

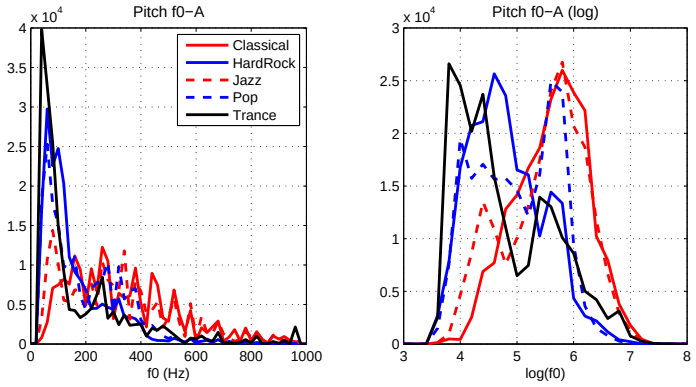
Trance is characterized by its use of a very clear beat in terms of a deep bass, however one of the more interesting attributes is the way it is composed. E.g. a majority of trace artist do not rely on a singers abilities to carry the track, but composes the track like a classical piece, where the use of tempo change, instrumentation and loudness carries a great weight, which in terms of feature is often seen as a semi-disconnected distributions.

Jazz: While pop music is often composed by following certain rules of harmony and melody, jazz music has the a very distinct use of improvisation, which makes it both quite interesting and difficult to handel in a machine learning environment. However, one special attribute of jazz is the use of distinct and often limited acoustical instruments like the saxophone seldom used in e.g. pop music, which often provides a distinct signature in the MFCC distributions.

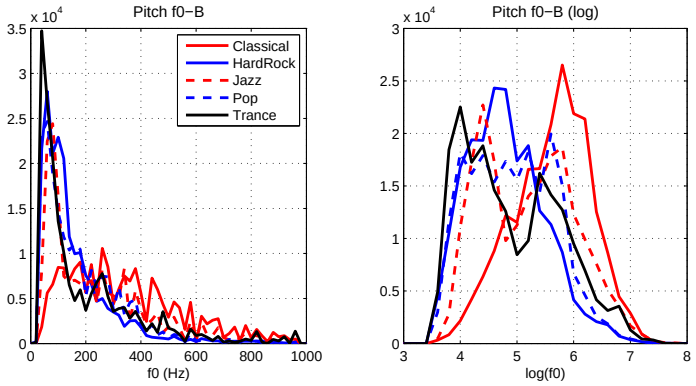
While the pieces and songs (referred to as tracks) by no means represent the complete musical scene, they do however cover some of the more dominant ones, which can always be found in larger sets.

3.1 Selected Feature Plots

During the creation of the data set the feature values were examined in a empirical fashion based on a visualization of the feature space. A few informative plots are included in figure H.2 in order to show the distribution of the features in terms of the individual genres described above. The genre distributions does obviously only provide information of genre separation, and is not optimal in the sense what we later consider the data set on the track and clip level to be. However, a detailed plot has a tendency to become non-informative. Due to a deeper insight into the POP genre a PCA plot is shown in appendix H.



(a)



(b)

Figure 3.2: Histogram for the pitch feature(s). The histograms shows, as expected, a quite skewed distribution towards the lower pitch range which in general is not a desirable property when using gaussian distributions to model the data, which in this thesis is songs, not genres. Although the mixture structure does improve this fact. Therefore the logarithm is applied to obtain the final features used. However, despite the smooth histogram shown, is the pitch a relatively discrete feature as previously noted.

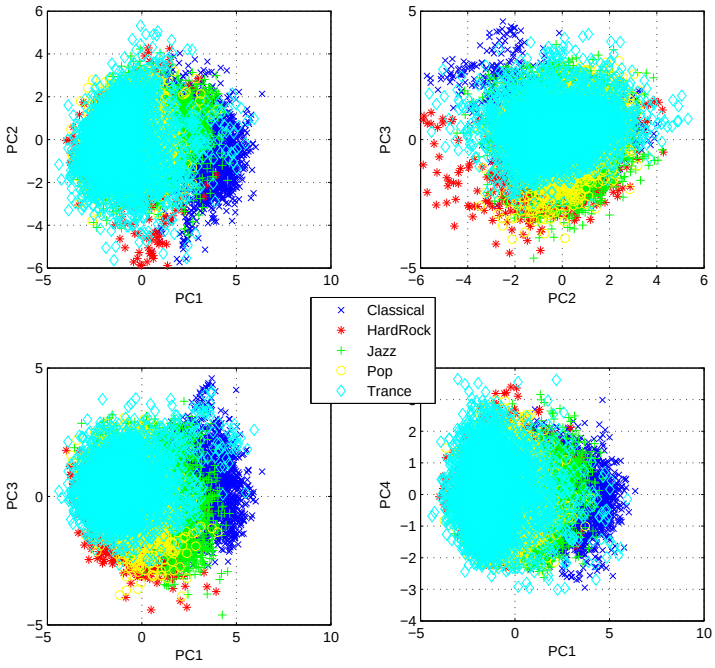


Figure 3.3: PCA projection of the genres using the MFCC and Pitch set. It is noticeable that the classical genre provides good separation from the remanding four genres, which are only partly separated. The separation of genres is of cause desirable when finding similar items across genres, however the genre plots does not indicate the with-in genre separation. Such a detailed plot is included in appendix H for the POP genre which will be examined in-dept through experiments. The projection is performed on a normalized data set, and a re-scaling may provide a better insight than the rather dense plot shown here.

Chapter 4

Learning in Music Databases

The concept of datamining and machine learning does to a large degree rely on the ability to learn how data relates to each other or how the data was generated. As for instance this thesis is motivated by the exploration of how data of one song, in terms of perceptual motivated features relates to the features of an other song.

The learning concept is the main focus of this chapter and is often, quite reasonably, referred to as machine learning. A huge number of techniques exist within this field, however, this thesis is not a review of machine learning and only describes the parts relevant for this project.

4.1 Learning by clustering

A very common tool in machine learning is so-called clustering, in which groups of data are identified based on some similarity measure. There are a great number of more or less custom clustering algorithms, however, in this thesis the focus is one a very basic clustering algorithm namely the well-known K-means algorithm, in which a number of clusters, K, is user defined. The basic K-means algorithm is a simple, but often applied clustering algorithm, which has been used in a huge number of applications. It is also referred to as a hard clustering algorithm compared to the EM (for GMM) later described, since it assigns each sample to one cluster, and one cluster only, creating so-called Voronoi regions, which are non-overlapping partitions in the feature space. The overall objective is to minimize the following cost-function

$$E = \sum_{k=1}^{K} \sum_{x_i \in \mathcal{S}_i} D\left(\mathbf{x}_i, \mu_k\right)^2$$

Where $D(\cdot, \cdot)$ is the distance function or metric, providing the distance from the cluster centroids μ_k to the data points and for all disjoint sets $\mathcal{S}_j$ of the entire feature space and

for all clusters K (user defined). This optimization is obtained through a simple iterative procedure.

The distance, D, is often defined to be the Euclidian distance. While this is an effective measure in high-spherical situations with good separation between clusters, such a simple distance measure is often too simple to account for the structure of the data. A large number of other distances has been considered of which some are listed in table4.1.

By predefining the number of clusters, K, we effectively assume a given structure of the data and the exploration idea might be somewhat fuzzy. In order to overcome this problem, hierarchical clustering is often used. One approach to this is agglomerative hierarchical clustering in which a large number of clusters are first fitted. These clusters are then combined/merged based on some similarity function. Despite the nice explorative idea in hierarchical clustering this will not be considered directly in this thesis which is aimed at a more basic retrieval type of exploration - which in essence is a distance only between two items. However, based on such a retrieval, a hierarchy can obviously be constructed, but the aim is first and foremost to construct the basic distance function between data points (or representations of these).

Minkowski	$\left(\sum \lvert x_{mi} - x_{mj}\rvert^p\right)^{1/p}$
Manhattan	$\sum \lvert x_{mi} - x_{mj}\rvert^2$
Euclidian	$\sqrt{\sum \left(x_{mi} - x_{mj}\right)^2}$
Cosine	$\cos(x_i, x_j) = \frac{\sum x_{mi} \cdot x_{mj}}{\sqrt{\sum (x_{mi})^2 \sum (x_{mj})^2}}$
Mahalanobis	$\sqrt{(x_i - x_j)^T \mathbf{C}^{-1} (x_i - x_j)}$

Table 4.1: Distance function or metrics. The summation is over the dimensions M. $\mathbf{C}$ is covariance matrix.

The Mahalanobis distance listed in the table is actually fairly closely related to the generic formulation of a distance function or metric, in which all directions and linear combinations of these, are weights to the Euclidian distance. This can be expressed as:

$$D(\mathbf{x}_i, \mathbf{x}_j)^2 = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{F} (\mathbf{x}_i - \mathbf{x}_j)$$

Where the matrix $\mathbf{F}$ defines the weighting of the direction, or features. The $\mathbf{F}$ matrix is here formulated as a constant matrix which in regards to the basic metrics is true, (i.e. the inverse covariance in the Mahalanobis formulation), however as we shall se later a general distance function can be expressed by a local $\mathbf{F}$, i.e. $\mathbf{F(x)}$, which can then be generalized to the entire space which will be describe in-depth later in this chapter. One major objective in this thesis is to investigate such a local distance function with the purpose of doing explorative retrieval in music based on the local properties of the feature space. The distance functions defined later has intrinsic relations to clustering applications, and hence will be evaluated in such a setting - of course compared to basic distance functions represented by the Euclidian and the Mahalanobis.

In this thesis the K-means will furthermore be used to initialize a considerably more complex algorithm, the EM-algorithm, described in section 4.2.1.

4.2 Density Modeling using Gaussian Mixture Models

The K-means clustering described above is often an effective clustering algorithm, but we are often interested in describing the way data was actually generated, i.e. form a model that explains the data $\mathcal{X}$ from a generative and probabilistic viewpoint, where $\mathcal{X}$ is the set of datapoints, i.e. $\mathcal{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$, where N is the number of points.

This can be done in various ways, but one widely used approach is to use a density model, i.e. a probabilistic model, describing the data by a distribution denoted as $p(x|\theta)$, where $\theta = \{\theta_1, ..., \theta_M\}$ are the parameters of the model.

Probably, the simplest option is to describe the data by a single, possibly multi-variate, Gaussian probability distribution given by

$$p(x|\theta) = \frac{1}{\sqrt{(2\pi)^M \det \mathbf{C}}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu)^T \mathbf{C}^{-1}(\mathbf{x}-\mu)\right\} \tag{4.1}$$

Where θ is given by the paraments μ as the mean vector and $\mathbf{C}$ as the MxM positive definite, covariance matrix $\mathbf{C} = E\left\{(\mathbf{x}-\mu)(\mathbf{x}-\mu)^T\right\}$. A single multivariate gaussian is however, often too simple for modelling complex data, and a more flexible mixture model is often preferred. In this thesis the focus will be on the well-known Gaussian Mixture Model of the form

$$p(\mathbf{x}) = \sum_{k=1}^{K} P(k) p(\mathbf{x}|\theta_k) \tag{4.2}$$

Where θ_k denotes the parameters of component k, although this parametrization will in the remaining text be denoted simply by $p(\mathbf{x}|k)$. K is the number of mixtures or components. Furthermore $\sum_{k=1}^{K} P(k) = 1$ and $0 \leq P(k) \leq 1$. $p(x)$ is of course conditioned on the combined set of θ_k's.

The pdf, $p(\mathbf{x}|k)$, can in principle be any distribution, however the most common is to use the Gaussian probability distribution in 4.1, which of course indicates the assumption that the data is generated from a number of Gaussian Distributions, which might not always be accurate. However given the central limit theorem, stating that the mean of N random variables tends to by distributed by a Gaussian distribution, for $N \to \infty$, we can hope that the data in some respect obeys by this generally stated theorem.

The Gaussian Mixture Model (GMM) is extremely flexible in the sense that the number of components K is user-defined, i.e. one can in theory model each data point by its own pdf, which will result in the likelihood $\mathcal{L}(\theta)$ to go to infinity, however as discussed above this is general not desirable since new data will most likely not be described well by such a model.

The number of components in the model is just one issue; another is the structure of the covariance matrix, which has a large influence on the complexity of the overall model, and it will later be demonstrated that certain traditional music similarity methods are very dependent on the correct choice of covariance model (on the data set described in chapter 2). Consider the following choices for each individual components and the number of parameters to be estimated This leads to a variety of options, and the best choice often depends on the data to be fitted and the noise in this respect. However, since the full covariance, does encapsulate the special case, a full covariance structure has been the main focus in this

Full	$K-1+K\left(M(M+1)/2+M\right.$
Diagonal	$K-1+KM+M$
Spherical	$K-1+K+M$

Table 4.2: Model complexity as a function of the number of components K and the dimension M. The case of a common covariance/variance for all models has been left out.

thesis for reason regarding the formulation of an distance measure, in which the shapes defined by the MxM covariance matrix $\mathbf{C}$ play an important role.

4.2.1 Maximum Likelihood learning - The EM algorithm

While the general formulation of the model was described in 4.2, the actual learning of the parameters is of course another main issue. In case of GMM's the far most predominantly option is to use the so called Expectation-Maximization (EM) algorithm, first suggested in 1977 by Dempster et al [8].

When considering parameter estimation, a common idea is the maximum likelihood principle, which is formulated in terms of the likelihood of the parameters given the data $\mathcal{X}$. The probability of $\mathcal{X}$ can, if assumed being independently drawn, be written as the product of individual probabilities of the data point $\mathbf{x}_n \in \mathcal{X}$, and applying the logarithm (natural).

$$\mathcal{L}(\theta) = \log \prod_{n=1}^{N} p(\mathbf{x}_n|\theta) \tag{4.3}$$

$$= \sum\nolimits_{n=1}^{N} \log\left(p(\mathbf{x}_n|\theta)\right) \tag{4.4}$$

Which is referred to as the log-likelihood of the data given the the model. When using the Gaussian Mixture defined in 4.2 we get

$$\mathcal{L}(\theta) = \sum\nolimits_{n=1}^{N} \log\left\{\sum_{k=1}^{K} P(k)p(\mathbf{x}|k)\right\} \tag{4.5}$$

$$\tag{4.6}$$

This leads to the optimization problem defined by

$$\theta^* = \arg\max_{\theta^*} \mathcal{L}(\theta) \tag{4.7}$$

Often, the optimization is formulated as minimizing the negative log-likelihood, which gives the same result due to the monotonic function. The solution to the maximization problem is often formulated as a bound optimization problem using so-called hidden variables S, so the likelihood can then be written based on the influence of these hidden variables and the visible variables, i.e.

$$\mathcal{L}(\theta) = \log p(\mathbf{S}, \mathcal{X}|\theta) \tag{4.8}$$

$$= \int p(\mathbf{S}, \mathcal{X}|\theta)\, d\mathbf{S} \tag{4.9}$$

Introducing a set of distributional function $q(\cdot)$ over the hidden variables $q(\mathbf{S})$, the log-likelihood can be rewritten so a lower bound is introduced on the log-likelihood and making

E-Step:

$$p(k|\mathbf{x}_n) = \frac{p\left(\mathbf{x}_n|k\right) P\left(k\right)}{\sum_{n=1}^{N} p\left(\mathbf{x}_n|k\right) P\left(k\right)}$$

M-Step:

$$P\left(k\right) = \frac{1}{N} \sum_{n=1}^{N} p\left(k|x_n\right)$$

$$\mu_k = \frac{\sum_{n=1}^{N} p\left(k|\mathbf{x}_n\right) \mathbf{x}_n}{\sum_{n=1}^{N} p\left(k|\mathbf{x}_n\right)}$$

$$\mathbf{C}_k = \frac{\sum_{n=1}^{N} \left(\mathbf{x}_n - \mu_k\right) \left(\mathbf{x}_n - \mu_k\right)^T P(k|y_n, \mathbf{x}_n)}{\sum_{n=1}^{N} P\left(k|y_n, \mathbf{x}_n\right)}$$

Figure 4.1: EM algorithm for Gaussian Mixture Model. Notice that the estimated posterior probability computed in the E-step is reused in the M-step, without explicit notation

an indirect optimization possible.

$$\begin{aligned} \mathcal{L}\left(\theta\right) &= \log\left(\int q\left(\mathbf{S}\right) \frac{p\left(\mathbf{S}, \mathcal{X}|\theta\right)}{q\left(\mathbf{S}\right)} d\mathbf{S}\right) && (4.10) \\ &\geq \int q\left(\mathbf{S}\right) \log \frac{p\left(\mathbf{S}, \mathcal{X}|\theta\right)}{q\left(\mathbf{S}\right)} d\mathbf{S} && (4.11) \\ &= \mathcal{F}\left(q\left(\mathbf{S}\right), \theta\right) && (4.12) \end{aligned}$$

where the inequality is introduced based on Jensen's inequality and the coactivity of the function. Noting than an optimization of $\mathcal{F}$ will also lead to a bounded optimization of the log-likelihood $\mathcal{L}$, the EM algorithm is formulated as an iterative method, by first optimizing the distribution over hidden variables q($\mathbf{S}$) and then subsequently modifying the parameters to reflect this change, i.e.

$$q\left(s_k\right)^{(i+1)} = \arg\max_{q(\mathbf{s}_k)} \mathcal{F}\left(q\left(\mathbf{S}\right), \theta^i\right) \quad (4.13)$$

$$\theta^{(i+1)} = \arg\max \mathcal{F}\left(q\left(\mathbf{S}\right)^{(i+1)}, \theta\right) \quad (4.14)$$

$$(4.15)$$

It can be shown that these updates guarantee convergence towards higher log-likelihood in each combined iteration.

The Gaussian Mixture model is an excellent example of application of the EM algorithm, and can be derived using the lower bound formulation above. The expectation steps estimates the posteriors given the current parameters and the maximization step re-estimates the parameters given the new posterior estimates. The actual updates equations are found by differentiating the complete log-likelihood of S and X in regards to the individual parameters and equating to zero, however this technical derivation has been left out. The algorithm for the Gaussian Mixture case is outlined in 4.1

In terms of the likelihood, the EM algorithm is guaranteed to converge in each iteration, see e.g.[8], however despite its wide use there are disadvantages. One of the more serious ones

is the fact that the EM approach is not immune to local minima which can be seen from the formal proof in Dempster [8], which implies a non-deterministic model if the initialization of the parameters are based on random assignments.

Another disadvantage is the tendency to overfit which originates in the structure of the model and the log-likelihood based cost-function[1]. The extreme overfitting example is when a component describes a single point, i.e. $\mathcal{L}(\mathbf{x}_n) \rightarrow \infty$, and this is not uncommon especially using a full covariance model if attention is not drawn to solving this issue.

Overcoming overfitting and initialization issues

Various steps will be taken to approach the overfitting behavior of the EM approach as mentioned previously in a practical sense. Due to the amount of models to be fitted (see chapter 6), a manual inspection of all models is not in general possible and a robust form of training is of course needed.

A common overfitting problem or indication of overfitting when using the full covariance model is a "collapse" of σ parameters, i.e. $\sigma_{i,j} -> 0$, where $1 < i, j \leq M$, leading to an extremely spiked posterior probability. This issue is resolved with a reconstruction of the covariance matrix back to e.g. the initial matrix when a collapse is observed (some threshold value is reached). A quite similar approach used is a regularization of the covariance matrix in order to avoid the collapse in the first place, i.e. $\mathbf{C}_{i+1} = \mathbf{C}_{i+1} + \alpha\mathbf{\Sigma}$ where Σ can be both a constant matrix, often the identity matrix, or the current covariance matrix $\mathbf{C}_i$.

Another practical more or less ad-hoc approach based on the generalization aspect, is early stopping. As the name suggests, does this involve stopping the EM algorithm when a certain criterium is satisfied, here based on an estimation of the generalization error. This estimation is based on a split of the training data $\mathcal{X}$ into two disjoint sets so $\mathcal{X} \equiv \mathcal{S}_A \cup \mathcal{S}_B$. The choice of $\mathcal{S}_A$ and $\mathcal{S}_B$ is described in section in connection with the models fitted.

The EM algorithm is guaranteed to have convergence towards lower negative log-likelihood, i.e. a (local) maximum likelihood solution, however this formulation does not guarantee convergence of the parameters, and a manual verification might be beneficial.

Another way used to ensure a relatively robust EM training is by proper initialization, which in this thesis will be performed by the K-means algorithm, initialized by a given random set of parameters based on the overall mean and variance of the complete data set. In this setting we ensure fairly stable convergence of the subsequent EM-training. An initial study showed a general improvement in both convergence speed and the consistency of the models returned, although they can still converge to suboptimal parameters, which for the experiments represented in chapter 6, will be handled by the multiple training of a model, and post-selecting the best model, based on a criterion presented in section 4.4.

[1] A Bayesian approach has been formulated which will either include Monte-Carlo sampling methods or a variational approach in which the lower-bound defined by Jensens inequality in the EM-derivation is addressed though variational methods.

4.3 Supervised Gaussian Mixture Model

In machine learning a classic paradox is the distinction between unsupervised learning - as considered above - and supervised learning in which human supervision is performed indirectly, often in the form of labeled data, i.e. a data point is defined by its vectorial data and class/label $\mathbf{x_i}, \mathbf{y_i}$, where y is a value from the set of labels, i.e. $y_i \in \{y_i, y_2, .., y = Y\}$corresponding to a predefined class.

The use of supervised learning does seem attractive in some cases, since the objective is now based, not solely on the data, but also on the defined property of these data. However, this can also lead to a very bad generalization since the supervised data available, may not provide the sufficient, or even correct, information of the underlying problem in order to create a general model for this underlying process.

The supervised formulation considered here is based on [17] is a fairly natural extension of the Gaussian Mixture Model and the EM-algorithm also considered in [21]. The model is formulated in terms of the joint posterior probability of data $\mathbf{x}$ and class labels $\mathbf{y}$, i.e [2]

$$p(\mathbf{x}, y) = \sum_{k=1}^{K} p(\mathbf{x}|k) P(k) p(y|k) \tag{4.16}$$

Where $p(\mathbf{x}|k)$ still refers to the gaussian component parameterized by θ_k. The class probability $p(y|k)$ is included in order to account for the labeled data. Furthermore the parameters of the individual Gaussians are now given by $\theta_k = \{P(y|k), \mu_k, C_k, P(k)\}$ and we restrict ourselves to cases where $\sum_{y=1}^{Y} P(y|k) = 1$.

The supervised model will effectively be used to model, not the joint probability distribution of y and $\mathbf{x}$, but the conditional distribution $p(y|\mathbf{x})$, which can be found from the joint distribution though Bayes theorem.

$$\begin{aligned}
p(y|\mathbf{x}) &= \frac{p(y, \mathbf{x})}{p(\mathbf{x})} && (4.17)\\
&= \sum_{k=1}^{K} p(y|k) p(k|\mathbf{x}) && (4.18)\\
&= \sum_{k=1}^{K} p(y|k) \frac{p(\mathbf{x}|k) P(k)}{p(\mathbf{x})} && (4.19)\\
&= \sum_{k=1}^{K} p(y|k) \frac{p(\mathbf{x}|k) P(k)}{\sum_{k'=1}^{K} p(\mathbf{x}|k') P(k')} && (4.20)\\
&= \frac{\sum_{k=1}^{K} p(y|k) p(\mathbf{x}|k) P(k)}{\sum_{k=1}^{K} p(\mathbf{x}|k) P(k)} && (4.21)
\end{aligned}$$

The training algorithm is based on the standard EM-algorithm for the unsupervised case. While [21] considers both unlabeled and labeled data, only the labeled data is considered

[2] While one often denotes the class by c, y is maintained due to a later conceptual distinction between the defined classes c and relevant information defined in terms of the assigned labels y. Although they do effectively reflects the same information.

here and the log-likelihood becomes.

$$\begin{aligned} \mathcal{L} &= \log p(\mathcal{X}|\theta) \\ &= \sum_{n\in\mathcal{X}} \log \sum_{k=1}^{K} P(y_n|k)p(\mathbf{x}_n|k)P(k) \end{aligned} \tag{4.22}$$

The learning is again based on the EM algorithm with $p(y|\mathbf{x})$ being estimated simply as the ratio of posterior component probabilities assigned a given label to the overall probability of the label. The basic unsupervised EM-algorithm, does have an unattractive tendency

E - step

$$P(k|y_n,\mathbf{x}_n) = \frac{P(y_n|k)p(\mathbf{x}_n|k)P(k)}{\sum_{k=1}^{K} P(y_n|k)p(\mathbf{x}_n|k)P(k)} \ , \ \forall n \in \mathcal{X}$$

M-Step

$$\mu_k = \frac{\sum_n \mathbf{x}_n P\left(k|y_n,\mathbf{x}_n\right)}{\sum_n P\left(k|y_n,\mathbf{x}_n\right)} \ , \ \forall k$$

$$\mathbf{C}_k = \frac{\sum_n \mathbf{S}_{kn} P\left(k|y_n,\mathbf{x}_n\right)}{\sum_n P\left(k|y_n,\mathbf{x}_n\right)} \ , \ \forall k$$

with $\mathbf{S}_{kn} = (\mathbf{x}_n - \mu_k)(\mathbf{x}_n - \mu_k)^T$

$$P\left(k\right) = \frac{\sum_n P\left(k|y_n,\mathbf{x}_n\right)}{N} \ , \ \forall k$$

$$P\left(y|k\right) = \frac{\sum\limits_n \delta\left(y - y_n\right) P\left(k|y_n,\mathbf{x}_n\right)}{\sum\limits_n P\left(k|y_n,\mathbf{x}_n\right)} \ , \ \forall k$$

Figure 4.2: Supervised Gaussian Mixture Model: EM training using a purely supervised approach.

to overfit (see e.g. [6]). Using a supervised algorithm does not improve this fact, on the contrary. By forcing the learning of the labeled training set, we very much assume that novel data comes from the exact distribution learned.

In order to provide better generalization the generalized mixture model has been suggested, in which a splitting of the data set is performed so the mean and covariances are estimated on two independent data sets. This approach was examined using a custom implementation of the algorithm in [17], however it was noticed that the convergence did not comply with the expected decrease in training error. Therefore the approach to better generalization is again performed by the early stopping criterium based on a validation set, and together with covariance regularization this has provided the expected behavior of training error, and found more beneficial in the situations encountered, since the deterministic behavior of the training error can also be exploited for other stopping criteria such as relative decrees in negative log-likelihood.

In this thesis the supervised algorithm will be used in a metric learning formulation, in which the metric is based on the change in posteriors class probability, 4.21, which of course requires an estimation of $p\left(y|\mathbf{x}\right)$.

4.4 Bayesian Learning & Approximations

The maximum likelihood objective which was used in the formulation of the well-known EM algorithm, is a quite way of estimating the model parameters. However, the problem with overfitting is quite serious, and a few practical solutions was mentioned. The formulation of the density model does not include the model complexity as such, which obviously has an extreme influence on the ability to overfit, with one mixture per data point being the extreme case.

While the optimal way to estimate the parameters is through a Bayesian formulation, i.e. by enforcing a prior on the parameters, it might not be worth applying this principle. In a mixture model trained using EM, the Bayesian approach can be approximated by looking at the lower bound introduced on the error, i.e. Jensen's equality. This leads to the Variational Bayes EM algorithm.

While this variational Bayes approach might seem optimal, another more practical approximation will be used. While the K-means and other non-probabilistic algorithms are simple and often provides quite fast convergence. The probabilistic nature of the Gaussian Mixture Model, does provide a direct advantage [3] because a Bayesian method of model selection can be used, although only approximately in this case.

When using Bayesian techniques for model comparison, we often assume a flat prior, i.e. no preference for either model. In this context the Bayesian Information Criterion can be derived.

Consider two models H_1 and H_2 . Using Bayes theorem the posterior probability of the data being generated from hypothesis is written as

$$p(H_k|\mathcal{X}) = \frac{p\left(\mathcal{X}|H_k\right)p\left(H_k\right)}{p\left(\mathcal{X}|H_1\right)p\left(H_1\right) + p\left(\mathcal{X}|H_2\right)p\left(H_2\right)} \tag{4.23}$$

or

$$\frac{p\left(H_1|\mathcal{X}\right)}{p\left(H_2|\mathcal{X}\right)} = \frac{p\left(\mathcal{X}|H_1\right)p\left(H_1\right)}{p\left(\mathcal{X}|H_2\right)p\left(H_2\right)} \tag{4.24}$$

where the factor $p\left(\mathcal{X}|H_1\right)/p\left(\mathcal{X}|H_2\right)$ is named the *Bayes factor*. Furthermore the factor $p(H_1)/p(H_2)$ is seen to be a prior factor. So given the Bayes factor between H1 and H2,

$$B_{12} = \frac{p\left(\mathcal{X}|H_1\right)}{p\left(\mathcal{X}|H_2\right)} \tag{4.25}$$

we are interested in estimating the likelihood of the data, $\mathcal{X}$, given the two models. When dealing with the mixture models, parameterized by the components θ_k we generally need to estimate the marginal distribution given by the integral over the conditional probability multiplied with the prior distribution,

$$p(H_1|\mathcal{X}) = \int p(\mathcal{X}|H_k, \theta_k)p(\theta_k, H_k)d\theta_k \tag{4.26}$$

In general this is not a trivial task and various approximations are suggested (see e.g. [32]), of which the following is often applied. Consider ignoring the prior distributions directly

[3] The K-Means "model" can also be formulated as mixture model with a given likelihood, but the training of course remains "hard", i.e. based on individual instances

and defining the following,

$$S_{12} = \log p(\mathcal{X}|H_1), \hat{\theta}_1 - \log p(\mathcal{X}|H_2), \hat{\theta}_2 - \frac{1}{2}(M_1 - M_2)\log N \tag{4.27}$$

Where $\hat{\theta}$ is the maximum likelihood estimation of the true θ and M_k is the dimension of $\hat{\theta}_{\mathbf{k}}$. Furthermore N is the sample size. The Schwarz criterion is then defined when $N \to \infty$, to satisfy

$$\frac{S_{12} - \log B_{12}}{\log B_{12}} \to 0 \;\; for \; N \to \infty \tag{4.28}$$

For multi model comparison we split the expression into two $S_{12} = S_1 - S_2$ and the individual contribution from each model is then given by

$$S_k = \log p(\mathcal{X}|H_k, \hat{\theta}_k) - \frac{1}{2} M_k \log N \tag{4.29}$$

The model, k, with the largest S is then preferred. It is custom to define a variant called the Bayesian Information Criterion as twice the negative Schwarz criterion, i.e.

$$BIC \equiv -2\log p(\mathcal{X}|H_k, \hat{\theta}_k) + M_k \log N \tag{4.30}$$

Where the preferred model with the highest evidence is chosen as the one with lowest BIC value.

While BIC is based on a Bayesian approach, other evaluation measures exist which in turn is not. One of these is the Akaike Information Criterion. The deviation results in a criterion quite similar to BIC

$$AIC \equiv -2\log p(\mathcal{X}|H_k, \hat{\theta}_k) + 2M_k \tag{4.31}$$

The only difference between BIC and AIC is clearly the $\log N$ factor in the penalty term, which will make the AIC suggest a more complex model than BIC.

AIC and BIC can be used to support the choice of model complexity in terms of number of components in the mixture model, and potentially the parametrization of the covariance matrices. While BIC and AIC are obviously nice guides in model selection, the real test for a given model is its ability to generalize in terms of new data. However this kind of evaluation is often quite time consuming, and obviously BIC provides an easy estimation, which will later be used for two main purposes: First in discarding badly initialized models by multiple training on the same data, in order to avoid the computational demanding task of calculating music similarity on all models. Furthermore BIC will be tested as a complexity indicator for music in which variable sized models may prove beneficial for well-known music similarity techniques presented in chapter 5.

4.5 Learning Using Metrics

Machine learning and datamining can in a very rough taxonomy be divided into probabilistic methods or instance based methods as previously mentioned, where the gaussian density model obviously is a probabilistic modelling technique. In such a relatively high-level description of the data in a given feature space we tend to abandon the meaning of the features in a local sense, and for example describe the audio clip by a gaussian distribution with a given covariance and centroid. The covariance obviously describes the more important directions or features for the given audio clip assuming an individual gaussian is fitted to each clip individually. As we shall see later such a description can be used directly to compare the clips, however comparing distributions directly may not directly reveal in an intuitive sense what makes one song more similar (or close) than another. Further a comparison of individual distributions does not directly account for the influence of other data points (maybe audio clips).

Based on the observations above, an alternative learning approach is taken in this section, dealing with an formal geometric view of the feature space on a somewhat lower level than the density models. However, a global density model is still maintained to describe the overall nature of the feature space. A local perspective is then constructed by the formulation of a so-called metric, which depending on the formulation describes the local importance of each direction in feature space. If the original feature space is retained (i.e. not projected) we hereby obtain a local importance of the features in each location from the metric.

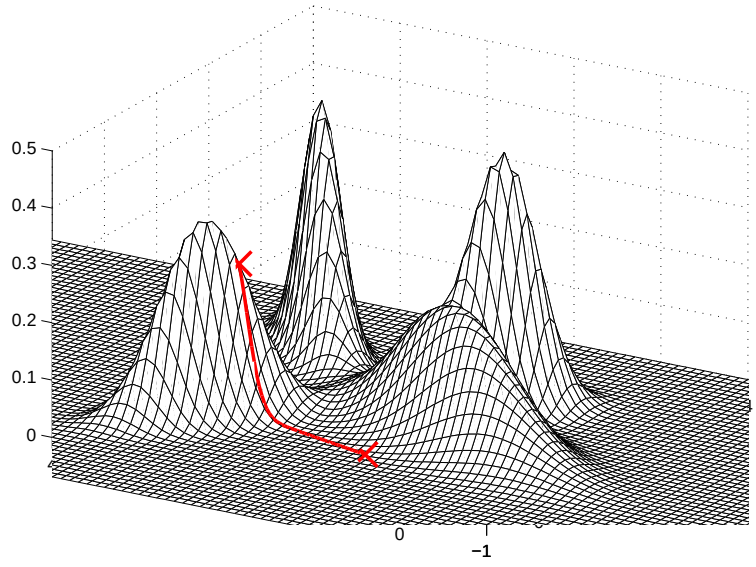


Figure 4.3: The "topology" of a density model. The log-likelihood is used as the description of the topology. The red line shows how a strath-line approximation leads to a non-Euclidian distance if integrating the change in log-likelihood along the path. This is effectively the Rattray metric.

The learning and datamining part can then be defined in terms of this local distance depending on the task at hand. As in many other machine learning applications, we are primarily focused on the distance between data points, like in a simple clustering algorithm. Given the metric (as a function of $\mathbf{x}$) we consider this along a curve between the two data points. This

is illustrated using a basic formulation of a metric (Rattray's metric) in figure 4.3, in which a density model provides a kind of height information of the feature space which effectively alters the inter-point distance, formally expressed by the MxM matrix $\mathbf{G}(\mathbf{x})$ (the metric at $\mathbf{x}$), where M is the dimension of the space. Figure 4.3 clearly shows the topology in this case defined as the change in log-likelihood given the probability model.

4.5.1 Metrics, Distances & Riemannian Geometry

In general we define a metric or distance function as a function d defined in a set $\mathcal{X}$, with four basic properties:

1. d is nonnegative and finite: $d(\mathbf{x}_i, \mathbf{x}_j) < \infty \, \forall \, \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$
2. $d(\mathbf{x}_i, \mathbf{x}_j) = 0$ if and only if $\mathbf{x}_i = \mathbf{x}_j$
3. d is symmetric: $d(\mathbf{x}_i, \mathbf{x}_j) = d(\mathbf{x}_j, \mathbf{x}_i)$.
4. The triangle inequality holds: $d(\mathbf{x}_i, \mathbf{x}_j) \leq d(\mathbf{x}_i, \mathbf{x}_k) + d(\mathbf{x}_k, \mathbf{x}_j) \, \forall \, \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in \mathcal{X}$.

A metric (or distance function) always define a topology in the space $\mathcal{X}$. The formal concept behind topology is rather complex (see i.e. [31, 1]), but the intuitive feeling of a map is quite applicant for many purposes since the main idea can be considered as an investigation of geometry (see e.g. for an in-depth treatment [10, 1]).

The properties listed above are both locally and global properties which must be obeyed also when generalizing the local metric to the global case, and does indeed hold for the very basic metrics defined in table 4.1. The following describes the theoretical steps in such a generalization.

The idea of defining a local metric and distance based on the topology or geometry of the space, is based upon the formal mathematical description of differential geometry, however a large part of the formal mathematical background is left out and only the absolute basic concepts will be included in order to formulate the idea of local metrics leading to global distances.

Given a certain level of abstraction and two points, a and b[4], in a Riemannian manifold (see e.g. [10, p.500]), S, connected by a curve, γ, we define the length of the curve based on an inner product, $g(u, v)$, between tangent vectors v and u to the curve γ at t. The length of a tangent vector is $\sqrt{g\,(v, v)}$ and the length of the curve parameterized by t with $\gamma(t = 0) = a$ and $\gamma(t = 1) = b$ is given by

$$\|\gamma\| = \int_0^1 \left\| \frac{d\gamma}{dt} \right\| dt = \int_0^1 \sqrt{g_{\gamma(t)} \left(\frac{d\gamma\,(t)}{dt}, \frac{d\gamma\,(t)}{dt} \right)} dt \tag{4.32}$$

The minimum length curve from a to b is defined as infinum of the curve lengths, i.e.

$$d\,(a, b) = \inf_{\{\gamma | \gamma(0)=a, \gamma(1)=b\}} |\gamma| \tag{4.33}$$

[4]Denoted a,b in order to specify that it is not necessarily a Euclidian space

which indicates that there is obviously more than one route between the points [a, b] on S, of which we are, as a starting point, interested in the minimum, denoted the distance, d.

In physics the length of the curve is often expressed in terms of the local coordinates (see e.g. [1, p. 6-12], but the abstraction above is quite sufficient for the purpose in this thesis.

Given an arbitrary local metric $\mathbf{F}(\mathbf{x})$, with the elements in $\mathbf{F}$ defined as an inner product between tangent vectors, in an assumed Euclidian space, we can we parameterize the curve γ from $\mathbf{x}_i$ to $\mathbf{x}_j$ as a straight line by $\mathbf{x} = \mathbf{x}_i + t(\mathbf{x}_j - \mathbf{x}_i)$ where $t \in [0, 1]$ in order to obtain an expression for the distance between the points. Based on the definition given in 4.32 of the distance between two points on the manifold, the global distance given a general metric $\mathbf{F}(\mathbf{x})$ becomes:

$$D(\mathbf{x}_i, \mathbf{x}_j) = \int_{t=0}^{1} \left[\nabla_t \mathbf{x}(t)^T \mathbf{F}(t) \nabla_t \mathbf{x}\right]^{1/2} \tag{4.34}$$

where $\nabla_t \mathbf{x}(t) = [\partial x_1/\partial t, \partial x_2/\partial t, ..., \partial x_d/\partial t]^T$. Even though appearing fairly simple, the integral in 4.34 is often analytically intractable, however there are different approaches to solving this problem of which some will be described and used for the demonstration of metrics in clustering examples and further on, in music retrieval.

The assumption of a Euclidian space is often a huge simplification given the true geometry of the space, but it is a convenient approximation to a true curve. A method for approximating the true minimum length curve, the geodesic, will be described later in this chapter.

The objective of using a local metric which is then generalized to the entire space, is based upon the idea that features does not mean the same at all points in space - i.e. we effectively weight the various directions differently in all points of the space, where the weighting is given by the topology of the space. This can be interpreted as a non-linear mapping of the original feature space, which could be performed using e.g. a neural network. However, the primary objective using a metric is the preservation of the topology, i.e. we maintain the meaning of the original features. This property is quite relevant in music based on meaningful perceptual features in which local metric can for example be used to describe the local meaning of the features in a point $\mathbf{x}$ as the relative relevance in a direction along the coordinate axis l, i.e.

$$r_l(\mathbf{x}) = \sqrt{\frac{\mathbf{e}_l^T \mathbf{J}(\mathbf{x}) \mathbf{e}_l}{\sum_m^M \mathbf{e}_m^T \mathbf{J}(\mathbf{x}) \mathbf{e}_m}} \tag{4.35}$$

This basic result can potentially be used to analyze the features on a low level, e.g. could indicate whether or not the pitch is important in the similarity estimation between two audio clips.

The formulation leading to the illustration in 4.3 was defined using the change in log-likelihood $p(\mathbf{x}|\theta)$ of a mixture model as the descriptor of the feature space. This, as well as two other formulations, will be considered in details in the following. For now they are described by

- Tipping: Locally weighted covariance based metric
- Rattray: Log-likelihood based metric
- Fisher/Kaski: Supervised metric based on the Fisher Information Matrix and change in so-called auxiliary information (class/labels)

The metrics are denoted by either $\mathbf{J}(\mathbf{x})$ for the metric based on Fishers Information Matrix (later described), or $\mathbf{G}(\mathbf{x})$ for the two metrics based on an unsupervised (heuristic) formulation. Furthermore, the general reference to the various metrics will be based on the originators of the three formulations (Tipping, Rattray and Fisher/Kaski[5]).

4.5.2 Tipping's Riemannian distance measure

The metric defined by Mike Tipping [29]is based on a pure clustering perspective in which, the foundation is the Mahalanobis distance, which is invariant to any non-singular scaling due to the covariance weighting (given a maximum likelihood estimate). However, the Mahalanobis distance, based on a global estimation and in terms of mixture models, is not applicant.

Therefore Tipping suggests using a heuristic metric, defined as a local weighting of the covariance matrices of a basic Gaussian Mixture Model. I.e. on the manifold, S, we define a metric based on a very general heuristic observation regarding the local contribution of the covariance weighted by the component posterior at a local point in data space:

$$\mathbf{G}(\mathbf{x}) = \sum_{k=1}^{K} p(k|x)\mathbf{C}_k{}^{-1} \tag{4.36}$$

where $\mathbf{C_k}$ is the covariance for each component. This means the metric is a weighted average of the inverse covariance matrices.

In general, this is still invariant to non-singular transformations and in the (perhaps local) limit where $K \to 1$ where we obtain the standard Mahalanobis distance. As with all other theory based on mixture models, there is the risk of poor local minima and a bad model will of course degrade the performance, for which reason certain measures was taken in the fitting process. While the local Tipping metric is rather intuitive, it does have to be generalized to the global feature space, which is done through the use of the straight line approximation described by 4.34. This is intractable in this case and an approximation will need to be formulated. In this thesis an analytical approximation will be investigated, which is suggested by Tipping, and further more will numerical integration methods be applied in subsequent sections.

Analytical Approximation

Tipping [29] suggests two major simplifications in order to obtain an analytical approximation of the Gaussian Mixture Model. First thing is to switch the order of the square root and the integral in 4.34. Furthermore the posterior conditional probability $p(k|\mathbf{x}) = p(\mathbf{x}|k)P(k)/p(\mathbf{x})$ is replaced by the approximation $p(\mathbf{x}|k)P(k)$. No evaluation of the simplification consequences are provided in [29], which will be performed through numerical evaluation in this thesis.

[5]The supervised metric is based on the Fisher Information Matrix, however the concrete application has been developed in several papers of which S. Kaski is the consistent author, hence the supervised metric will be refereed to as Fisher/Kaski or simply Kaski when the reference makes more sense in term of the formulation

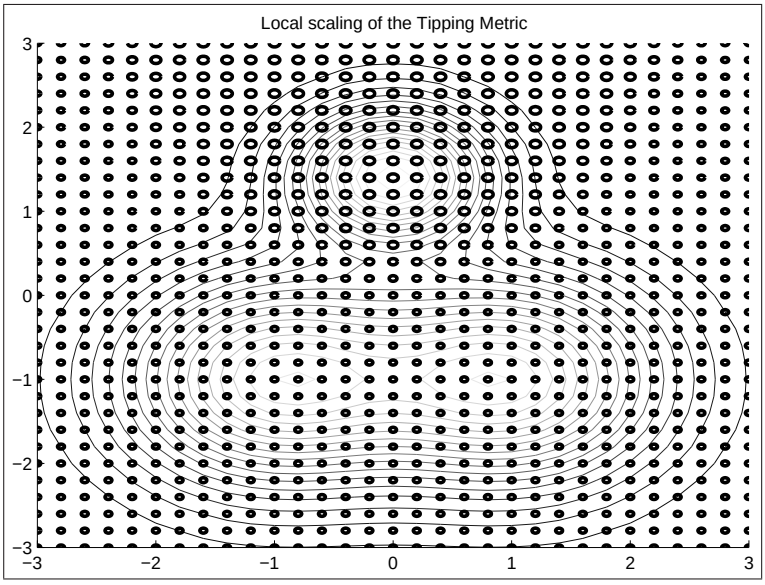


Figure 4.4: The local scaling in the Tipping Metric. Clock-wise from the top the three equally spaced distributions have the following parameters: P(n)=[1/3,2/3,2/3], $\sigma = [0.3, 0.7, 0.7]$

Since $\nabla_t \mathbf{x}(t) = (\mathbf{x}_i - \mathbf{x}_j)$, we get

$$D = \int_{t=0}^{1} \left[(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{G}(t) (\mathbf{x}_i - \mathbf{x}_j) \right]^{1/2} dt \tag{4.37}$$

$$= \int_{t=0}^{1} \left[(\mathbf{x}_i - \mathbf{x}_j)^T \left[\sum_{k=1}^{K} C_k^{-1} p(k|\mathbf{x}(t)) \right] (\mathbf{x}_i - \mathbf{x}_j) \right]^{1/2} dt \tag{4.38}$$

and applying the approximations yields

$$D^2 \approx \int_{t=0}^{1} (\mathbf{x}_i - \mathbf{x}_j)^T \left[\sum_{k=1}^{K} \mathbf{C}_k{}^{-1} p(k|\mathbf{x}(t)) \right] (\mathbf{x}_i - \mathbf{x}_j) dt \tag{4.39}$$

$$\approx \int_{t=0}^{1} (\mathbf{x}_i - \mathbf{x}_j)^T \left[\sum_{k=1}^{K} \mathbf{C}_k{}^{-1} \frac{p(\mathbf{x}(t)|k)P(k)}{\sum_K p(\mathbf{x}(t)|k)P(k)} \right] (\mathbf{x}_i - \mathbf{x}_j) dt \tag{4.40}$$

This integral has no closed form solution which makes it analytically intractable. The sum-factor still has dependence on t, so in order obtain a trackable expression, the posterior conditional probability of component k, $p(k|\mathbf{x})$, is approximated by $P(k)p(\mathbf{x}|k)$, i.e. neglecting the normalization in Bayes theorem.

$$D^2 \approx \int_{t=0}^{1} (\mathbf{x}_i - \mathbf{x}_j)^T \left[\sum_{k=1}^{K} \mathbf{C}_k{}^{-1} P(k) p(\mathbf{x}(t)|k) \right] (\mathbf{x}_i - \mathbf{x}_j) dt \tag{4.41}$$

The expression still has dependence on t in the sum, so the $\mathbf{G}(\mathbf{x})$ expression (brackets) is made constant by making a probabilistic weighted average of the individual covariance

matrices along the straight line approximation, based on the approximation of $p(k|\mathbf{x})$. This is formulated as

$$D^*(\mathbf{x}_i, \mathbf{x}_j)^2 = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{G}^* (\mathbf{x}_i - \mathbf{x}_j) \int_0^1 dt \tag{4.42}$$

$$= (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{G}^* (\mathbf{x}_i - \mathbf{x}_j) \tag{4.43}$$

$$\mathbf{G}^* = \frac{\sum_{k=1}^{K} {\mathbf{C}_k}^{-1} P(k) \int_{\mathbf{x}=\mathbf{x}_i}^{\mathbf{x}_j} p(\mathbf{x}|k) d\mathbf{x}}{\sum_{k=1}^{K} P(k) \int_{\mathbf{x}=\mathbf{x}_i}^{\mathbf{x}_j} p(\mathbf{x}|k) d\mathbf{x}} \tag{4.44}$$

Since $p(\mathbf{x}|k)$ is a single gaussian component of the mixture model, we can find a closed form solution to the approximation. The integral in 4.44 is the path integral from $\mathbf{x}_i$ to $\mathbf{x}_j$ along the **s**traight path. After writing the parameterized pdf as a quadratic form and utilizing the error function, the following expression is found,

$$\int_{\mathbf{x}=\mathbf{x}_i}^{\mathbf{x}_j} p(\mathbf{x}|k)\, dx = \sqrt{\frac{\pi b^2}{2}} \exp\{-Z/2\} \left[erf\left(\frac{1-a}{\sqrt{2b^2}}\right) - erf\left(\frac{-a}{\sqrt{2b^2}}\right)\right] \tag{4.45}$$

With erf being the error function defined as

$$erf(y) = \frac{2}{\sqrt{2}} \int_{t=0}^{y} e^{-t^2} dt \tag{4.46}$$

Furthermore a,b and Z are given by:

$$b^2 = \left(\mathbf{v^T C_k^{-1} v}\right)^{-1} \tag{4.47}$$

$$a = \mathbf{b^2 v}^T \mathbf{C_k^{-1} u} \tag{4.48}$$

$$Z = \mathbf{u^T C_k^{-1} u} - b^2 \left(\mathbf{v^T C_k^{-1} u}\right)^2 \tag{4.49}$$

It is very difficult to evaluate the consequences of this approximation in a general setting, and the final conclusion is almost entirely based on the results obtained through practical simulations in the end of this section.

4.5.3 Rattray's Riemannian distance metric

Tipping's metric as defined above has a somewhat heuristic formulation in terms of weighted covariance matrixes. Rattray [26] has suggested a different metric also aimed at clustering based on the assumption that a cluster is a homogeneous, connected region. The metric itself can then be defined from a viewpoint saying that it should reflect the change in log-likelihood of the data, so the distance is given by the incremental change in log-likelihood:

$$d\mathbf{s} = |\log p(\mathbf{x} + d\mathbf{x}) - \log p(\mathbf{x})| \tag{4.50}$$

By assuming the incremental distance $d\mathbf{x}$ is small enough we can write

$$\begin{aligned} d\mathbf{s} &\simeq \left| d\mathbf{x}^T \nabla_x \log p(\mathbf{x}) \right| && (4.51) \\ &= \sqrt{d\mathbf{x}^T \nabla_x \log p(\mathbf{x}) \left(\nabla_x \log p(\mathbf{x})\right)^T d\mathbf{x}} && (4.52) \\ &= \sqrt{d\mathbf{x}^T \nabla_x G(\mathbf{x}) d\mathbf{x}} && (4.53) \end{aligned}$$

With the Riemannian metric given as

$$\mathbf{G}(\mathbf{x}) = \nabla_{\mathbf{x}} \log p(\mathbf{x}) \left(\nabla_{\mathbf{x}} \log p(\mathbf{x})\right)^T \tag{4.54}$$

A metric based directly on the log-likelihood and the assumption that the data is grouped or clustered in high-density areas will inevitable be sensitive to situations in which these assumptions break down. This can happen when either the model is a poor reflection of the true data, e.g. in cases of overfitting or local minima will a point described by its own pdf have a long distance to all other points regardless of these being relatively close or not.

The metric can furthermore be seen (illustrated in figure 4.5) to reward high density areas where the local scaling is insignificant.

In order to improve the performance of the log-likelihood based metric, Rattray suggests using a shortest distance search algorithm to find the shortest distance between two points using other points, again supporting the high-density idea. This algorithm will be applied to all three metrics and described later.

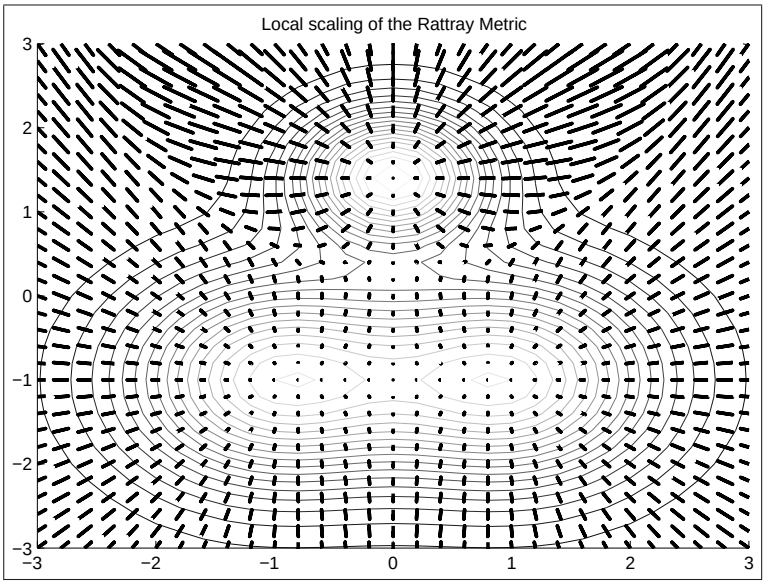


Figure 4.5: The local scaling in the Rattray Metric. Clock-wise from the top the three equally spaced distributions have the following parameters: P(n)=[1/3,2/3,2/3], $\sigma = [0.3, 0.7, 0.7]$

Modeling with Gaussian Mixtures Models

Although not formulated in terms of mixture models, this model is applied in [26], and outlined below. The metric can, using the GMM, be expanded:

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{x}} \log p(\mathbf{x}) &= \frac{1}{p(\mathbf{x})} \frac{\partial}{\partial \mathbf{x}} \sum_K P(k) p(\mathbf{x}|k) && (4.55)\\
&= \frac{1}{p(\mathbf{x})} \frac{\partial}{\partial \mathbf{x}} \sum_K P(k) \frac{1}{\sqrt{(2\pi)^d \det C^{-1}}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu_k)^T \mathbf{C}_k^{-1}(\mathbf{x}-\mu_k)\right\} && (4.56)\\
&= \frac{1}{p(\mathbf{x})} \sum_K -P(k)\mathbf{C}^{-1}(\mathbf{x}-\mu_k) \frac{1}{\sqrt{(2\pi)^d \det \mathbf{C}_k^{-1}}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu_k)^T \mathbf{C}_k^{-1}(\mathbf{x}-\mu_k)\right\} && (4.57)\\
&= \frac{1}{p(\mathbf{x})} \sum_K -P(k)p(x|k)\mathbf{C}_k^{-1}(\mathbf{x}-\mu_k) && (4.58)\\
&= \sum_K \frac{-P(k)p(\mathbf{x}|k)}{p(\mathbf{x})} \mathbf{C}_k^{-1}(\mathbf{x}-\mu_k) && (4.59)
\end{aligned}$$

Through Bayes theorem we find $p(k|\mathbf{x}) = P(k)p(\mathbf{x}|k)/p(\mathbf{x})$ and the expression can be simplified to yield:

$$\frac{\partial}{\partial \mathbf{x}} \log p(\mathbf{x}) = \sum_K -p(k|\mathbf{x})\mathbf{C}_k^{-1}(\mathbf{x}-\mu_k) \tag{4.60}$$

Since $(\mathbf{AB})^T = \mathbf{B}^T\mathbf{A}^T$ and $\mathbf{C}$ is symmetrical, so $\left(\mathbf{C}^{-1}\right)^T = \left(\mathbf{C}^{-1}\right)$, we get the following expression for the metric:

$$\begin{aligned}
\mathbf{G} &= \nabla_{\mathbf{x}} \log p(\mathbf{x}) \left(\nabla_{\mathbf{x}} \log p(\mathbf{x})\right)^T && (4.61)\\
&= \sum_{k=1}^{K} -p(k|\mathbf{x})\mathbf{C}_k^{-1}(\mathbf{x}-\mu_k) \left(\sum_{l=1}^{K} -p(l|\mathbf{x})\mathbf{C}_l^{-1}(\mathbf{x}-\mu_l)\right)^T && (4.62)\\
&= \sum_{k=1}^{K}\sum_{l=1}^{K} p(k|\mathbf{x})p(l|\mathbf{x})\mathbf{C}_k^{-1}(\mathbf{x}-\mu_k)(\mathbf{x}-\mu_l)^T\mathbf{C}_l^{-1} && (4.63)
\end{aligned}$$

Globalizing this metric using the integral in 4.34, again yields an untractable integral which, as in the Tipping case, calls for analytical and numerical solutions.

Analytical Approximation

The same approximations used by Tipping, are suggested by Rattray in [26]. This again involves finding a constant metric along the path as with the Tipping metric, so the integral can be calculated without numerical computation.

$$G_*(\mathbf{x}_i, \mathbf{x}_j) = \frac{\sum_{k,l} P(k)P(l)\mathbf{C}_k^{-1}A_{kl}\mathbf{C}_l^{-1}}{\sum_{k,l} P(k)P(l)a_{kl}} \tag{4.64}$$

where the integral over individual components are given by

$$a_{kl} = \int_{t=0}^{1} p(\mathbf{x}|k)\, p(\mathbf{x}|l)\, dt \tag{4.65}$$

and

$$A_{kl} = \int_{t=0}^{1} (\mathbf{x} - \mu_k)(\mathbf{x} - \mu_k)^T p(\mathbf{x}|k)\, p(\mathbf{x}|l) dt \tag{4.66}$$

Obviously these integrals are to be evaluated over the two components versus one in the Tipping case. This again calls for a rewrite into quadratic forms and calls for partial integral, which has been verified to yield the following simplified results:

$$a_{kl} = \frac{e^{-\frac{\gamma}{2}}}{(2\pi)^d \sqrt{|\mathbf{C}_k||\mathbf{C}_l|}} f(\alpha, \beta) \tag{4.67}$$

$$A_{kl} = \frac{e^{-\frac{\gamma}{2}}}{(2\pi)^d \sqrt{|\mathbf{C}_k||\mathbf{C}_l|}} \left(\mathbf{w}_k \mathbf{w}_l^T f(\alpha, \beta) - \left(\mathbf{v}\mathbf{w}_l^T + \mathbf{w}_k^T \mathbf{v}^T \right) \frac{\partial f(\alpha,\beta)}{\partial \beta} + \mathbf{v}\mathbf{v}^T \frac{\partial^2 f(\alpha,\beta)}{\partial \beta} \right) \tag{4.68}$$

With α, β and γ given by

$$\alpha = \mathbf{v}^T \left(\mathbf{C}_k^{-1} + \mathbf{C}_l^{-1} \right) \mathbf{v} \tag{4.69}$$

$$\beta = \mathbf{v}^T \left(\mathbf{C}_k^{-1} \mathbf{w}_k + \mathbf{C}_l^{-1} \mathbf{w}_l \right) \tag{4.70}$$

$$\gamma = \mathbf{w}_k^T \mathbf{C}_k^{-1} \mathbf{w}_k + \mathbf{w}_l^T \mathbf{C}_l^{-1} \mathbf{w}_l \tag{4.71}$$

and f is then expressed by

$$f(\alpha, \beta) = \int_{t=0}^{1} e^{-\frac{\alpha t^2}{2} - \beta t} dt = \sqrt{\frac{\pi}{2\alpha}} e^{\frac{\beta^2}{2\alpha}} \left[erf\left(\frac{\beta - \alpha}{\sqrt{2\alpha}} \right) - erf\left(\frac{\beta}{\sqrt{2\alpha}} \right) \right] \tag{4.72}$$

The derivative of f, originating from the partial integration can be simplified to yield,

$$\frac{\partial f(\alpha, \beta)}{\partial \beta} = \frac{1}{\alpha} \left[\beta f(\alpha, \beta) + e^{-\frac{\alpha}{2} - \beta} - 1 \right] \tag{4.73}$$

$$\frac{\partial^2 f(\alpha, \beta)}{\partial \beta^2} = \frac{1}{\alpha} \left[\beta \frac{\partial f(\alpha, \beta)}{\partial \beta} + f(\alpha, \beta) - e^{-\frac{\alpha}{2} - \beta} \right] \tag{4.74}$$

This rather involved approximation and subsequent implementation, is validated on a simple 1D example and the real-world data set considered later on.

4.5.4 Supervised Riemannian Metric

While the formulations provided by Tipping and Rattray are based on purely unsupervised approaches, an obvious idea is to include available knowledge about the data in the form of so-called auxiliary information into the metric learning, effectively turning it into a supervised metric. In a music context this could be the information that some data point originates from e.g. a certain song, genre or artist, however it need not be formulated with the objective to classify into these classes, as explained later.

This idea has first been formulated by Kaski et al and described in several places [25, 15, 14]. The basic idea is to use the conditional probability of the class/label given the data vector, i.e $p(y|\mathbf{x})$ and define the metric in terms of change in this conditional distribution.

While Rattray's formulation was based on a simple use of Riemannian manifold without considering any parametrization of the space and only considering the topology provided by the log-likelihood, the supervised approach parameterizes the feature space by the use of the classes or as joined auxiliary information in e.g. [15], which here will be denoted y in order to underline that it is not necessarily the original classes of the data we are interested in, but some defined relevance, which could span several the original determined classes of the data.[6]

The metric is formulated in terms of the distributional "distance" between the conditional probability distributions $p(y|\mathbf{x})$ and $p(y|\mathbf{x}+d\mathbf{x})$, which in the context of information theory naturally leads to the use of the Kullback-Leibler divergence, and the distance becomes

$$\mathcal{D}_{KL}\left(p\left(y|\mathbf{x}\right)||p\left(y|\mathbf{x}+d\mathbf{x}\right)\right)=\int p\left(y|\mathbf{x}\right)\log\frac{p\left(y|\mathbf{x}\right)}{p\left(y|\mathbf{x}+d\mathbf{x}\right)}d\mathbf{x} \tag{4.75}$$

If assuming $p(y|\mathbf{x})$ is differentiable, it can be shown that local Kullback-Leibler divergence between the the two distributions $p(y|\mathbf{x})$ and $p(y|\mathbf{x}+d\mathbf{x})$ is given by (a proof is outlined in appendix A for completeness)

$$d^2\left(\mathbf{x},\mathbf{x}+d\mathbf{x}\right)=D_{KL}\left(p\left(y|\mathbf{x}\right),p\left(y|\mathbf{x}+d\mathbf{x}\right)\right) \tag{4.76}$$

$$=\tfrac{1}{2}d\mathbf{x}^T\mathbf{J}(\mathbf{x})d\mathbf{x} \tag{4.77}$$

[7] Where $\mathbf{J}(\mathbf{x})$ is the Fisher Information Matrix given by

$$J(\mathbf{x})=E_{p(y|\mathbf{x})}\left\{\frac{\partial\log p\left(y|\mathbf{x}\right)}{\partial\mathbf{x}}\left(\frac{\partial\log p\left(y|\mathbf{x}\right)}{\partial\mathbf{x}}\right)^T\right\} \tag{4.78}$$

Information geometry is used as a statistical inference method, and often used to describe the topology of a space described by a parametrization θ and in general information geometry the Fisher Information matrix is the natural metric describing a change in the distribution given an incremental change in the parameters θ, i.e. the objective is to investigate models in a geometric sense.

In this project we are interested in a distance, not between models, but between points in feature space given the knowledge of the posterior class probability $p\left(y|\mathbf{x}\right)$. Therefore the classic distribution, $p(\mathbf{x}|\theta)$, usually of interest in traditional information geometry is replaced by $p\left(y|\mathbf{x}\right)$, which results in the Fisher Information Matrix in 4.78.

The conditional probability $p\left(y|\mathbf{x}\right)$ of course has to be estimated or modelled for all $\mathbf{x}$. There are several ways of estimating this posterior, ranging from Parzen estimators to a direct modelling of $\prod_N p\left(y|\mathbf{x}_n\right)$ for all data points N, which is solved by a gradient decent optimization [25].

This thesis is limited to the investigation of the supervised Gaussian Mixture Model in 4.16, in which Bayes theorem can be used to give an estimate of $p\left(y|\mathbf{x}\right)$ (see later). However before considering this (in this context advanced model), a simple example will be given explaining the implications of using the Fisher Metric (Kaski and Fisher metric is used interchangeably).

[6] Obviously a redefinition of an original class will lead to the same result, and y is simply introduced for the purpose of distinction between concepts

[7] It should be noted that the constant is ignored in the further calculations, but this only changes the absolute size of the metric, not the relative distance between data points.

Example: Two-class Linear Discriminant in the Fisher/Kaski metric

In order to illustrate the properties of the Fisher/Kaski metric in a simple and easy interpretable situation, the metric is derived and illustrated for a two-class problem, using the simple linear discriminant with the logistic activation function (see e.g. [6]), which despite yielding an obvious result, does serve as an excellent example.

The linear discriminant function for a two-class problem is given by $\varphi(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$, and for $\varphi(\mathbf{x}) > 0$ the class is one. While such a binary classification is desirable in some cases, one desirable property in statistical modelling is the interpretation of probabilities, which is also required in the formulation of the Fisher/Kaski Metric. Therefore the logistic sigmoid activation function (*softmax* for multi class problems) is applied. It is in general given by $g(a) = 1/1 + e^{-a}$, where a is the activation.

In the two class problem the activation function can be expressed by the discriminant function $\varphi(\mathbf{x})$ if we interpret the output as the probability of class one ($y = 1$), i.e.

$$p(y = 1|x) = \frac{1}{1 + e^{-\varphi(\mathbf{x})}} \tag{4.79}$$

However, the symmetry of the two class problem can be expressed by $p(y = 2|\mathbf{x}) = 1 - p(y = 1|\mathbf{x})$. This naturally also extend to the derivative of $logp(y|\mathbf{x})$ used in 4.78

$$\frac{\partial \log p(y = 2|\mathbf{x})}{\partial \mathbf{x}} = -\frac{\partial \log p(y = 1|\mathbf{x})}{\partial \mathbf{x}} \tag{4.80}$$

Using the fact that $\frac{\partial \log p(y|\mathbf{x})}{\partial \mathbf{x}} = \frac{1}{p(y|\mathbf{x})}\frac{\partial p(y|\mathbf{x})}{\partial \mathbf{x}}$ we get for the Fisher/Kaski metric in 4.78

$$\begin{aligned}
\mathbf{J}(\mathbf{x}) &= \sum_{y=1}^{2} p(y|x)\frac{1}{p(y|x)}\frac{\partial \log p(y|\mathbf{x})}{\partial \mathbf{x}}\frac{1}{p(y|x)}\left(\frac{\partial \log p(y|\mathbf{x})}{\partial \mathbf{x}}\right)^T && (4.81)\\
&= \sum_{y=1}^{2} \frac{1}{p(y|x)}\frac{\partial \log p(y|\mathbf{x})}{\partial \mathbf{x}}\left(\frac{\partial \log p(y|\mathbf{x})}{\partial \mathbf{x}}\right)^T && (4.82)\\
&= \frac{1}{p(y = 1|x)}\frac{\partial p(y = 1|\mathbf{x})}{\partial \mathbf{x}}\left(\frac{\partial p(y = 1|\mathbf{x})}{\partial \mathbf{x}}\right)^T \\
&\quad + \frac{1}{p(y = 2|x)}\frac{\partial p(y = 2|\mathbf{x})}{\partial \mathbf{x}}\left(\frac{\partial p(y = 2|\mathbf{x})}{\partial \mathbf{x}}\right)^T && (4.83)\\
&= \left[\frac{1}{p(y = 1|\mathbf{x})} + \frac{1}{1 - p(y = 1|\mathbf{x})}\right]\frac{\partial p(y = 1|\mathbf{x})}{\partial \mathbf{x}}\left(\frac{\partial p(y = 1|\mathbf{x})}{\partial \mathbf{x}}\right)^T && (4.84)\\
&= p(y = 1|\mathbf{x})(1 - p(y = 1|\mathbf{x}))\frac{\partial \varphi(\mathbf{x})}{\partial \mathbf{x}}\left(\frac{\partial \varphi(\mathbf{x})}{\partial \mathbf{x}}\right)^T && (4.85)
\end{aligned}$$

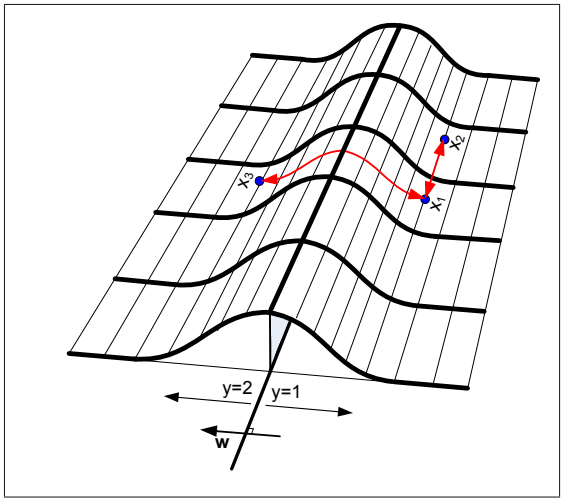


Figure 4.6: The gradient of $p(y|\mathbf{x})$ illustrated. The maximum of $p(y|\mathbf{x})$ is obtained when $p(y|\mathbf{x}) = 1/2$ leading to a maximum of $\frac{\partial p(y|\mathbf{x})}{\partial \mathbf{x}} = 1/4$. The principle is shown through the three example point in which the distance from $\mathbf{x}_1$ to $\mathbf{x}_2$ is zeros due to the non-changing gradient, and $\mathbf{x}_1$ to $\mathbf{x}_3$ seen to cross the decision border leading to a non-zeros result, depending on the properties of the vector $\mathbf{w}$

since

$$\frac{\partial}{\partial \mathbf{x}} \frac{1}{1+e^{-\varphi(\mathbf{x})}} = \frac{1}{\left(1+e^{-\varphi(\mathbf{x})}\right)^2} e^{-\varphi(\mathbf{x})} \frac{\partial \varphi(\mathbf{x})}{\partial \mathbf{x}} \tag{4.86}$$

$$= \frac{e^{-\varphi(\mathbf{x})}}{\left(1+e^{-\varphi(\mathbf{x})}\right)^2} \frac{\partial \varphi(\mathbf{x})}{\partial \mathbf{x}} \tag{4.87}$$

$$= \frac{e^{-\varphi(\mathbf{x})}}{1+e^{-\varphi(\mathbf{x})}} \frac{1}{1+e^{-\varphi(\mathbf{x})}} \frac{\partial \varphi(\mathbf{x})}{\partial \mathbf{x}} \tag{4.88}$$

$$= p(y=2|\mathbf{x})\, p(y=1|\mathbf{x}) \frac{\partial \varphi(\mathbf{x})}{\partial \mathbf{x}} \tag{4.89}$$

$$= (1-p(y=1|\mathbf{x}))\, p(y=1|\mathbf{x}) \frac{\partial \varphi(\mathbf{x})}{\partial \mathbf{x}} \tag{4.90}$$

Plugging the above and $\partial\varphi(\mathbf{x})/\partial\mathbf{x} = \mathbf{w}$ into 4.78, we obtain

$$\mathbf{J}(\mathbf{x}) = (1-p(y=1|\mathbf{x}))\, p(y=1|\mathbf{x})\, \mathbf{w}\mathbf{w}^T \tag{4.91}$$

The squared incremental distance d^2 is then expressed by

$$d^2(\mathbf{x}, \mathbf{x}+d\mathbf{x}) = d\mathbf{x}^T \mathbf{J}(\mathbf{x})\, d\mathbf{x} \tag{4.92}$$

$$= d\mathbf{x}^T (1-p(y=1|\mathbf{x}))\, p(y=1|\mathbf{x})\, \mathbf{w}\,\mathbf{w}^T d\mathbf{x} \tag{4.93}$$

or more perhaps more informative in the sense that the absolute value can been seen to be directly dependent on the direction, due to the scalar product of the weight vector $\mathbf{w}$ and $d\mathbf{x}$

$$d^2(\mathbf{x}, \mathbf{x}+d\mathbf{x}) = (1-p(y=1|\mathbf{x}))\, p(y=1|\mathbf{x}) \left(d\mathbf{x}\,\mathbf{w}^T\right)^2 \tag{4.94}$$

The analysis of $d^2(\mathbf{x}, \mathbf{x}+d\mathbf{x})$ is quite intuitive, although important for the following discussions: Consider a $d\mathbf{x}$ vector lying parallel with the contours of $(1-p(y=1|\mathbf{x}))\, p(y=1|\mathbf{x})$.

In this case there is no change in $p\left(y=1|\mathbf{x}\right)$ (and the vector product $d\mathbf{x}^T\mathbf{w}$ is zero as well), hence $d(\mathbf{x},\mathbf{x}+d\mathbf{x})=0$, as illustrated going from $\mathbf{x}_1$ to $\mathbf{x}_2$ in figure 4.6. The same results is obtained if $d\mathbf{x}$ is located in the *flat* area of the gradient which occurs when $p\left(y=1|\mathbf{x}\right)=0$ or $p\left(y=2|\mathbf{x}\right)=0$, i.e. there is no change in the label y. Generally these two intuitively easy situations implies that the distance reflects the original labeling of the data i.e. it maintains a zero distance with the class if no change in $p(y|\mathbf{x})$ is observed along the path of travel.

A much more interesting situation is encountered if $d\mathbf{x}$ is perpendicular to the decision boundary and hence parallel to $\mathbf{w}$. In this case we experience a change in $\left(1-p\left(y=1|\mathbf{x}\right)\right)p\left(y=1|\mathbf{x}\right)$ as seen in figure 4.6 going from $\mathbf{x}_1$ to $\mathbf{x}_3$. While the incremental change $d\mathbf{x}$ gives the (very) local distance in a given direction, the global distance from e.g. $\mathbf{x}_1$ to $\mathbf{x}_3$ will have to be obtained through integration over $\mathbf{x}$ along the path as previously mentioned.

This small example shows the main idea behind the Fisher/Kaski metric, i.e. that we obtain a non-zero distance in areas of changing class probability, provided we travel in a direction of change. The extreme case is obtained when fully crossing the decision boundary - and a perhaps more interesting intuitive result is when crossing two thought decision boundaries we end up with twice the distance (provided the decision boundary is of the same shape). This is in contrast to a normal classification which will discussed trough an example later.

Modeling with supervised Gaussian Mixture Models

The simple two class linear discriminant example above serves an a nice introduction to the properties of the Fisher/Kaski metric, but it is hardly applicant for many purposes. For a multi class problem, several methods can be used to model the conditional density, however here the proposal is to model $p\left(y,\mathbf{x}\right)$ with the following previously described supervised mixture model describing the joint probability of $\{\mathbf{x}_n,y_n\}$

$$p(\mathbf{x},y)=\sum_{k=1}^{K}p(\mathbf{x}|k)P(k)p(y|k) \tag{4.95}$$

With $p(\mathbf{x}|k)$ being the individual gaussian component parameterized by θ_k. Using Bayes theorem we get, as previously mentioned

$$\begin{aligned} p\left(y|\mathbf{x}\right) &= \frac{p\left(y,\mathbf{x}\right)}{p\left(\mathbf{x}\right)} &(4.96)\\ &= \frac{\sum_{k=1}^{K}p\left(y|k\right)p\left(\mathbf{x}|\theta_k\right)P\left(k\right)}{\sum_{k=1}^{K}p\left(\mathbf{x}|\theta_k\right)P\left(k\right)} &(4.97)\end{aligned}$$

Differentiating this expression is a rather involved task and the full derivation has been included in the appendix, expanding the derivation in [14] to a full covariance model used later on. The resulting metric is given by,

$$\frac{\partial\log p\left(y|\mathbf{x}\right)}{\partial\mathbf{x}}=\sum_{k=1}^{K}-\left[p\left(k|\mathbf{x},y\right)-p\left(k|\mathbf{x}\right)\right]\mathbf{C}_k^{-1}\left(\mathbf{x}-\mu_k\right) \tag{4.98}$$

Plugging this into the Fisher Information matrix in 4.78 we get

$$\frac{\partial\log p\left(y|\mathbf{x}\right)}{\partial\mathbf{x}}\left(\frac{\partial\log p\left(y|\mathbf{x}\right)}{\partial\mathbf{x}}\right)^T=\sum_{\mathrm{k,l}=1}^{K}\left[p\left(k|\mathbf{x},y\right)-p\left(k|\mathbf{x}\right)\right]\left[p\left(l|\mathbf{x},y\right)-p\left(l|\mathbf{x}\right)\right]\mathbf{Q}_{kl} \tag{4.99}$$

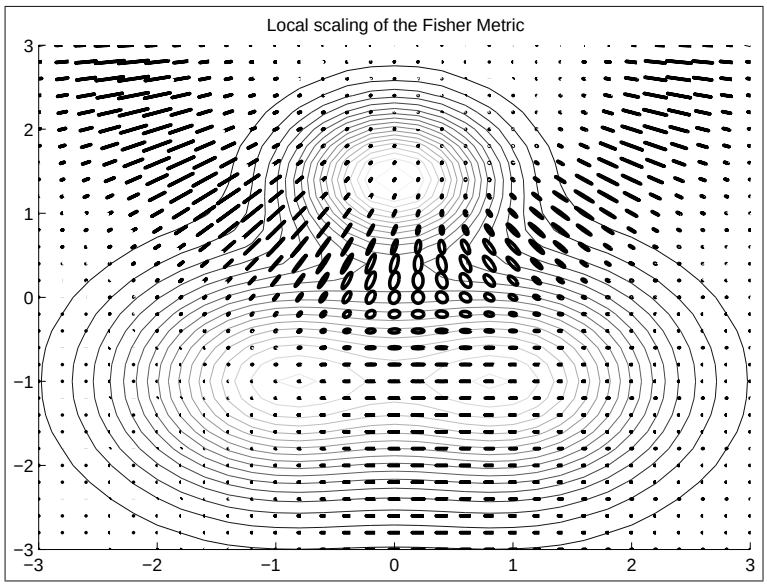


Figure 4.7: The local scaling in the Fisher Metric. Clock-wise from the top the three equally spaced distributions have the following parameters: P(n)=[1/3,2/3,2/3], $\sigma = [0.3, 0.7, 0.7]$

where $\mathbf{Q}_{kl} = \mathbf{C}_k^{-1} (\mathbf{x} - \mu_k) (\mathbf{x} - \mu_l)^T \mathbf{C}_l^{-1}$

Taking the expectation with regards to $p(y|\mathbf{x})$

$$J(\mathbf{x}) = E_{p(y|\mathbf{x})} \left\{ \sum_{k,l=1}^{K} [p(k|\mathbf{x}, y) - p(k|\mathbf{x})] [p(l|\mathbf{x}, y) - p(l|\mathbf{x})] \mathbf{Q}_{kl} \right\} \quad (4.100)$$

$$= \sum_{n=1}^{Y} p(y_n|\mathbf{x}) \sum_{k,l=1}^{K} [p(k|\mathbf{x}, y_n) - p(k|\mathbf{x})] [p(l|\mathbf{x}, y_n) - p(l|\mathbf{x})] \mathbf{Q}_{kl} \quad (4.101)$$

Note that an extra sum is introduced due to the expectation which together with the extra computation needed to find $p(k|\mathbf{x}, y)$ adds to the computational load.

No analytical approximation of the integral in 4.34 is suggested when using the Fisher/Kaski metric in the original papers, and none is attempted within the context of this thesis. [8]

4.5.5 Computational approximation to path integrals

A more or less ad hoc analytical approximation is provided for the Tipping and Rattray metrics when used in a global sense, i.e. by the use of path integrals in 4.34. No evaluation of these approximations is provided, but several computational/numerical approximations can be made to the rather demanding integral, when dealing with intractable expressions as in our case.

[8] Other possibilities has not been investigated in-dept, though, due to the relatively late inclusion of the interesting supervised metric.

The intractability is sometimes considered a tabu in statistical machine learning - however the interesting nature of the metric surpasses the computational problem - and we are forced into numerical and computational approximations (at least for the Fisher Metric) -or in case of the two unsupervised metrics, an analytical approximation can fairly easy be constructed.

Numerical Integration

In this thesis a numerical evaluation of the original - in general - intractable integral in 4.34 will be performed in order to evaluate the effect of the approximations in a practical setting. An adaptive Simpson quadrature algorithm will be used. A review of this algorithm can be found in [16].

An issue with this basic numerical integration - and any subsequent T-points approximations methods - is if the curve, along which the integration is performed, contains highly peaked values. In relation to the integral this means a large gradient of either $p(\mathbf{x})$ or especially $p(c|\mathbf{x})$ which may have a large gradient due to a sharp decision boundary between the defined classes.

T-Point Approximations

The path integral used in the Riemannian measure/distance can be approximated by a T-Point approximation, also suggested in [25] in which the path is represented by T-points from $\mathbf{x}_i$ to $\mathbf{x}_j$,

$$D_T(\mathbf{x}_i, \mathbf{x}_j) = \sum_{t=1}^{T} d_1\left(\mathbf{x}_1 + \tfrac{t-1}{T}\mathbf{v}, \mathbf{x}_1 + \tfrac{t-1}{T}\mathbf{v}\right)$$
$$\mathbf{v} = \mathbf{x}_i - \mathbf{x}_j$$

Obviously the simplest case yields a 1-point approximation, which though is found not to perform well in a K-nearest neighbour classifier (see e.g. [25]), however in [25] the approximation was made at $t = 0$ assuming the interesting area to be just around $\mathbf{x}$, but this may not be the general case, and in this thesis we define the T-point approximations to be made at $t = 0.5$ i.e. in between the T points on the straight line approximation. This has two main advantages in the authors mind. The main advantage is the fact that it symmetrizes the T-point approximation, so it obtains true metric properties (see 4.5.1), and it does not favour the one point used as reference over the other, which is considered an advantage when using it for retrieval, but in a clustering situation this argument may not be valid, since the reference point is now a cluster centroid.

In applications where the exact distance is important a more accurate approximation is suggested here, which is a quite obvious extension to the basic T-points approximation. By the definition of a number of points per unit distance, instead of a fixed value, an adaptive T-point approximation can be constructed[9]. This will guarantee a uniform accuracy throughout the space.

[9]This is the equivalent of using a numerical integration method with low resolution, corresponding to the numerical of T-points per unit distance

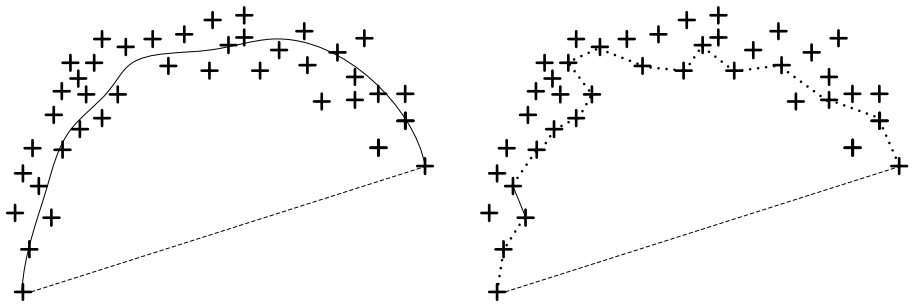


Figure 4.8: The geodesic path between the two points (solid line) are approximated by the dashed line in the right figure. In a Euclidean space the geodesic path is the straight line showed in both figures [adapted from [26]]

Graph Approximations to Geodesic Paths

The definition of a distance on a Riemannian manifold is not limited to straight Euclidean path between points, as seen in the genral definition of the curve length 4.32 and 4.33, but is actually given by the so-called geodesic path - which for the Euclidian vector space is the straight line. However, on a Riemannian manifold this is not necessarily the case given the non-linearities of the metric. In order to approximate the geodesic path, a graph approximation can be used.

An edge in the graph is represented by the pair-wise distance between two arbitrary points, which effectively means two other data points as shown in figure 4.8.

Using this approximation the geodesic path on the manifold can be found by the use of a dynamic programming algorithm in order to solve the following

$$D_{floyd}\left(\mathbf{x}_i, \mathbf{x}_j\right) = \min_{M, X \in \left\{x_1, \ldots, x_M^{'}\right\}} d\left(\mathbf{x}_i, \mathbf{x}_i^{'}\right) + \sum_{m=1}^{M-1} d\left(\mathbf{x}_m^{'} \mathbf{x}_{m+1}^{'}\right) + d\left(\mathbf{x}_M^{'} \mathbf{x}_j\right)$$

This problem can be solved with e.g. Floyd's algorithm. The approach is very computational heavy since it scales like $\mathcal{O}\left(n^3\right)$, and is therefore only applicant in real-world cases in which the inter-point distances do not change, i.e. a one time computation is performed of all pair-wise distances. Furthermore it is obviously only relevant in cases where there are points in the proximities (in the metric sense) in order to construct the approximating graph.

4.5.6 Examples and evaluation in 1D

In order to evaluate the effect of applying model based metrics, a simple 1-D problem is considered. The model consists of two classes described by three Gaussian's, with the middle distribution being the second class with only one gaussian. These labeled classes will obviously only influence the supervised Fisher Metric. The dotted line in figure 4.9 shows how the decision boundary between the classes. As shown using the local metric as an indicator in figure 4.7, and the argument given in the linear discriminant example, does

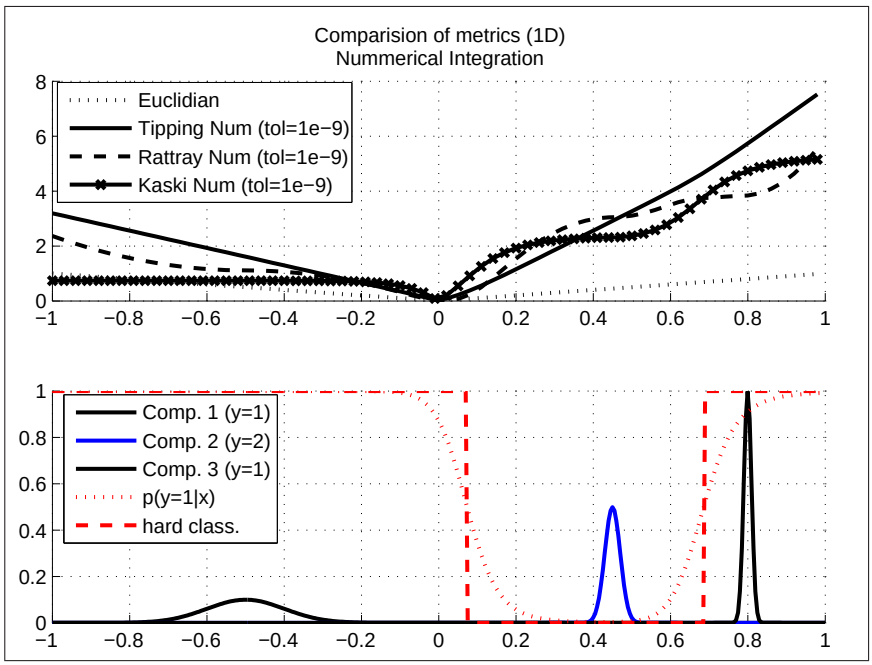


Figure 4.9: The Riemannian based metric distances evaluated using numerical integration. The model is given as $\sigma(k) = [0.1, 0.02, 0.01]$, $\mu(k) = [-0.5, 0.45, 0.8]$, $P(k) = [0.2, 0.6, 0.2]$ The red dashed line shows the hard classification border, while the dotted line shows the actual posterior $p(y|x)$. The distances are evaluated from $\mathbf{x} = 0$

the Fisher/Kaski distance only change in the vicinity of the decision boundary and when crossing it.

An quite interesting property to be noted is the Fisher/Kaski metric. Despite the point of reference is $x = 0$, i.e. in class 1, the distance to the other class 1 distribution is not zero, as would be the case in a pure classification measure. Furthermore it is again noted that the Fisher metric does not change within the class except at the decision boundary, which in this simple example is limited to a narrow area, but in a general case this can be a quite big area as seen in the clustering examples later.

The Tipping and Rattray measures are naturally unaffected by the exact point of separation between classes, but are not invariant to the in-class covariance contribution and log-likelihood changes, which obviously reflects the two (unsupervised) formulations.

Approximations in 1D

The analytical approximations made by Tipping and Rattray to their respective metrics, are evaluated in the simple 1D case (and later in a clustering example). The results of this can be seen for a few selected cases in figure 4.10.

It is seen that the Tipping approximation - compared with the numerical integration in figure 4.9 - in this simple case seems to be valid. Despite the same approximations being made to the Rattray formulation it seems the extra complexity of the metric derived for

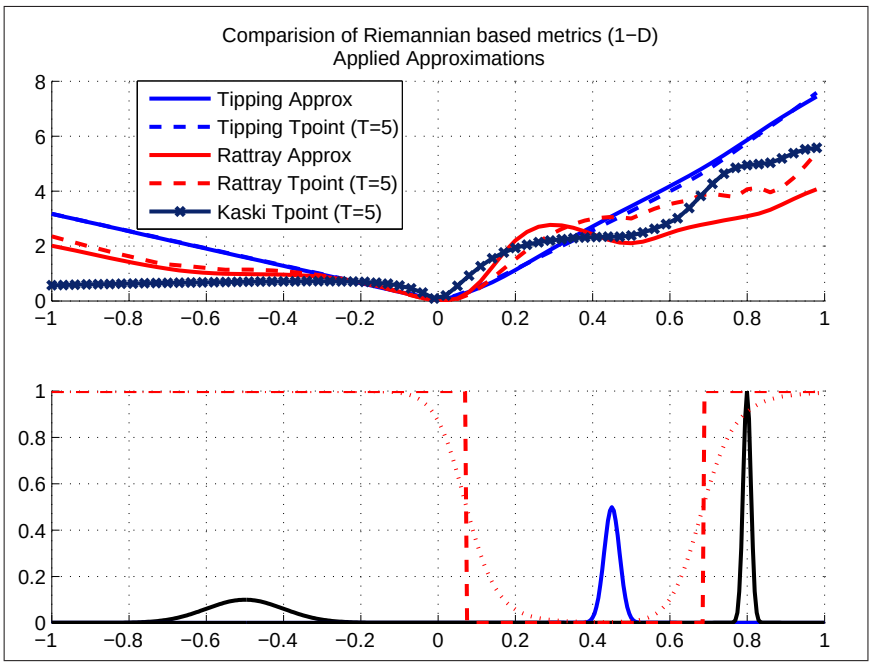


Figure 4.10: The Riemannian based metric evaluated in terms of their approximations which will be applied to the music set in chapter 6. The distances are evaluated from x=0

mixture models 4.63 adds to the error introduced by the analytical approximation. The reason for this has not been investigated further, though. The T=5 point approximations all seem to provide a good approximation to the "true" numerical integration, but a final conclusion should be based on the individual data set and fitted model, since this very small example obviously does not encapsulate all special cases.

While the T-point approximations are not particular exact at far away distances, due to the lower resolution of points along the straight line approximation - it should be noted that the approximations do hold for points close by due to the basic properties the $d\mathbf{x}\,\mathbf{F}(\mathbf{x})\,d\mathbf{x}$, which is an important point in applications of data mining where the closer points are often the ones of interest (e.g to a cluster center), and the distant points need not be calculated with high precision, which is one of the circumstances justifying application of the rather crude approximations.

4.6 Clustering with local metrics

In order to examine the actual "learning" properties of the metrics and evaluate the performance of various choices of approximations of distances on actual data sets, a number of clustering experiments are presented, which show the advantage of local metrics compared to Euclidian based distances. The data consists of two artificially generated sets in order to view the particular properties in various interesting cases. Further more a real-life set is used, namely the Phoneme data set.

The clustering is performed using the K-means algorithm described in 4.1, although the computational load of the metrics (especially the numerical integration) is quite demanding and the centroid of the clusters are limited to the data points themselves. This implies that inter-point distances need only be computed once, which is found acceptable for the purpose of investigating the metrics against each other.

In this toy setup we evaluate the performance of a resulting clustering $\mathbf{C}$ primarily by the purity of the clusters, evaluated against the true cluster configuration $\mathbf{E}$ generating the data.

Purity is defined by considering all points of a cluster $c_i \in \mathbf{C}$ as being classified as members of c_i's primary/dominant class, which is the class $\varepsilon_j \in \mathbf{E}$, with which c_i shares maximal number of elements. For cluster c_i purity is defined as the ratio between those elements shared by c_i and ε_j, to the total number of elements in c_i providing the maximum number of shared members, i.e.

$$Purity\,(c_i|\mathbf{E}) = \frac{1}{|\mathrm{c_i}|} \max_{\varepsilon_j \in \mathrm{E}} \{|\mathrm{c_i} \cap \varepsilon_j|\} \tag{4.102}$$

where $|c_i|$ is the number of points in cluster i and $|\varepsilon|$ is the number of points in the true class j. $|c \cap \varepsilon_j|$ is the number of elements shared by c_i and ε_j . For an overall evaluation of the final results the individual clusters are weighted by the number of members in the cluster, which results in the following

$$Purity\,(\mathbf{C}|\mathbf{E}) = \frac{1}{N} \sum_{c_i \in \mathbf{C}} \max_{\varepsilon_j \in \mathrm{E}} \{|\mathrm{c_i} \cap \varepsilon_j|\} \tag{4.103}$$

where N is the total number of points. Certain classes ε may not share maximal number of elements with any cluster given the above formulation, hence several different clusters may share the maximal intersection with the same class. In the primary evaluation of the metrics, we limit the clustering to be performed only with the true number of clusters, and indirectly ensuring that only one cluster shares the maximal intersection with a class, mainly for the purpose of illustration.

When dealing with multiple clusters in which the true number of clusters is unknown or can not be checked, clusters may share the maximal intersection with the same class and the purity measure can provide misleading results (same happens if the number of clusters grows), since the final clustering $\mathbf{C}$ may not reflect the true configuration but still return a large purity. In trivial and simple situations (like the two first presented) the purity may be sufficient, but in more complex situations, like the real-world example demonstrated later, a more strict measure is needed to easily evaluate the correspondence between the true configuration and the one obtained through clustering. For this purpose the Jaccard coefficient is used, which measures the agreement between an evaluated clustering configuration $\mathbf{C}$ and a pre-defined clustering $\mathbf{E}$ on assigning pairs of data to the same cluster versus different clusters.

The following hard valued functions (0/1), are defined for every pair of data **x** and **x'**:

$$Co - Assign_{\mathbf{C}}(x, x') = 1 \; if \, there \, exist \, c_i \in \mathbf{C}, such \, that \, (x, x') \in c_i \, otherwise \, 0$$

$$Co - Assign_{\mathbf{E}}(x, x') = 1 \; if \, there \, exist \, \varepsilon_i \in \mathbf{E}, such \, that \, (x, x') \in \varepsilon_j \, otherwise \, 0$$

A set of counts is then defined based on the co-assigned functions for each pair

$$A_{11} = \sum_{(x,x') \in \mathbf{C}} \min \{Co - Assign_{\mathbf{C}}(x, x'), Co - Assign_{\mathbf{E}}(x, x')\}$$

The number of relevant pairs assigned into the same cluster by both **E** and **C**;

$$A_{10} = \sum_{(x,x') \in \mathbf{C}} \min \{Co - Assign_{\mathbf{C}}(x, x'), 1 - Co - Assign_{\mathbf{E}}(x, x')\}$$

The number of pairs that have been assigned into the same cluster by **C** but not by **E**.

$$A_{01} = \sum_{(x,x') \in \mathbf{C}} \min \{1 - Co - Assign_{\mathbf{C}}(x, x'), Co - Assign_{\mathbf{E}}(x, x')\}$$

The number of pairs that have been assigned into the same cluster by **E** but not by **C**. Then the Jaccard coefficient, ignoring the A_{00} term:

$$Jaccard\,(\mathbf{C}|\mathbf{E}) = \frac{A_{11}}{A_{11} + A_{01} + A_{10}} \tag{4.104}$$

Which is considered to be a good measure - combined with the purity measure for given purpose of evaluation the metrics.

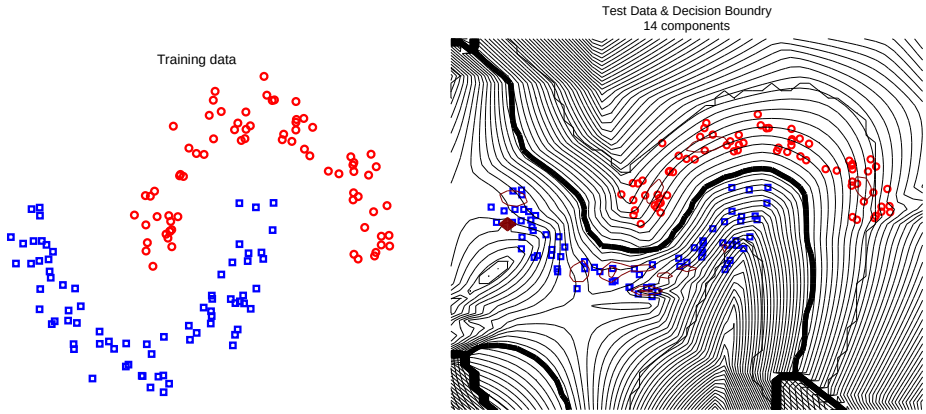
The initialization of the cluster centroids has been observed to have a significant influence on the final results as in almost all K-means configurations, therefore several (ten) initializations are performed and an estimation of the sensitivity to the initialization is performed based on the realistic initializations, which can be interpreted as loose estimation of the metrics robustness to the extract point of reference - relevant for further discussions concerning the retrieval in music. Moreover is various model sizes considered in order to evaluate - especially the supervised metric.

Clustering with metrics I: Curved data

This data set consists of two quite complex distributions, responsible for generating the curved data illustrated in figure 4.11(a) on which various sized models are fitted and a 14 components are shown in figure 4.11(b).

This example is mainly created to show the properties of the three metrics and Floyd's algorithm on a challenging data set and quite a few interesting situations can be explained through the use of this set especially in relation to the Fisher/Kaski metric, including various perhaps unfortunate properties of using numerical approximations.

The example included here is based on a 14 component mixture which provides reasonable results for all metrics, but results for 6,8,10,12 and 16 components are aggregated in tables in appendix E.



(a) (b)

Figure 4.11: Curved Data: (a) Training data (b) Example model fitted with 14 componnets resulting in a quite complex decision border at $p(y|x) = 1/2$ between the two clusters/classes. The black countours are the changes in $p(y|x)$ (logarithm)

The data is illustrated in figure 4.11(a) along with the decision boundary, furthermore the gradient of $p(y|\mathbf{x})$ is illustrated using the contour lines (logarithm applied first). This shows a quite complex nature of $\partial p(y|\mathbf{x})/\partial x$ and the distance between two points within the same class is not zero due to the changes in $\partial p(y|\mathbf{x})/\partial \mathbf{x}$. The situation is complicated further by the fact that the curved data combined with the straight line approximation can cause an inter-point distance to include two crossings of the decision boundary, obviously resulting in a point in the other class being closer. However, due to the K-means optimization of the centroids, such a situation does not generally occur when clustering, although the approximation may induce similar situations.

K=14	**Analyt**	**Num (1e-6)**	**T=1**	**T=5**	**T=15**
Euclidian	0.51±0.056 0.57				
Mahalanobis	0.56±0.14 0.72				
Tipping	0.43±0.12 0.63	0.5±0.083 0.64	0.42±0.13 0.52	0.49±0.046 0.64	0.47±0.081 0.64
Tipping-Floyd	0.56±0 0.56	0.58±0.058 0.6	0.58±0.058 0.6	0.58±0.058 0.6	0.58±0.058 0.6
Rattray	0.57±0.1 0.67	0.76±0 0.76	0.69±0.0084 0.69	0.68±0.022 0.69	0.68±0.028 0.69
Rattray-Floyd	0.57±0.23 0.91	0.97±0.097 1	1±0 1	1±0 1	1±0 1
Kaski		1±0 1	0.87±0 0.87	0.86±0.013 0.87	0.86±0.017 0.87
Kaski-Floyd		1±0 1	1±0 1	1±0 1	1±0 1

Table 4.3: Curved Data I: K = 14. Purity of the classes over 10 different K-means initializations including the maximum obtained (as second row)

In order to visualize the clustering, a few examples are included. The "true" behavior in terms of the numerical integration will be illustrated for each metric, including the Floyd version, for the 14 component case. Furthermore the T=15 point approximation is included and comments attached in a general sense also relating to the overall results.

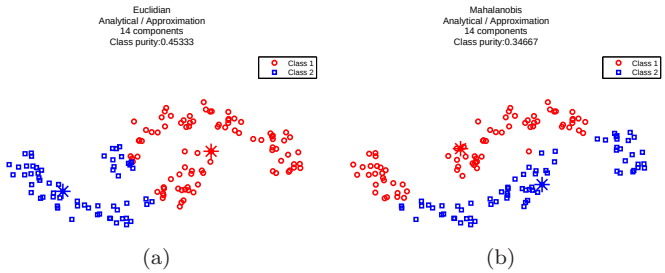


Figure 4.12: Example of basic metric. (a) Euclidian, obviously does not depend on the model, but varies only in the initializations. (b) Mahalanobis, generally a fair improvement over the Euclidian for the best initialization. However, when dealing with complex data the Mahalanobis is seldom sufficient as metric

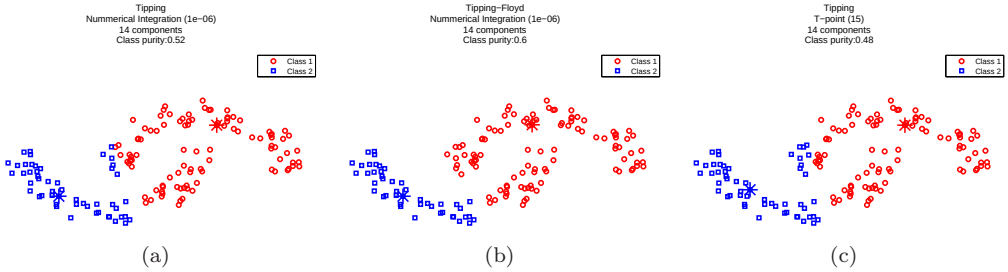


Figure 4.13: Example of Tipping metric. (a) Basic Tipping. All Tipping cases seems to perform poorly on this data set, indicating that such complex data is not beneficial to the weighted covariance formulation. (b) Tipping With Floyd (c) T-point (15) approximation

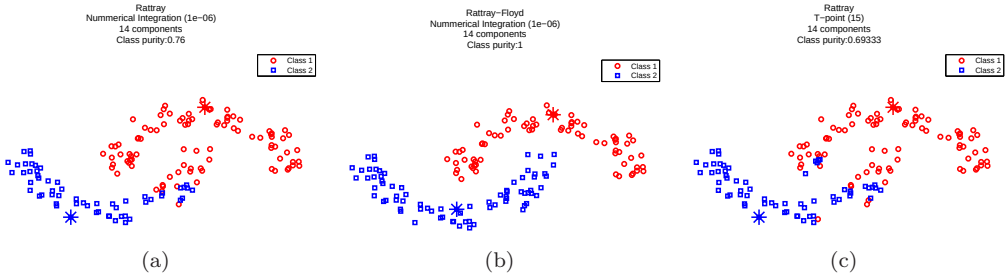


Figure 4.14: Example of Rattray metric. (a) Basic Rattray. Generally very dependent on the model, however not consistent since depends on the local log-likelihood - not the overall likelihood. (b) Rattray with Floyd almost always results in perfect clustering on this set (c) T-point (15) approximation of Rattray metric without Floyd, suggesting a more spurious result. However, T-point approximation often results in a slightly better result than numerical integration, which can be explained by lower sensitivity to non-representative changes in log-likelihood, i.e. a smoothing of the provided model. Generally the Rattray metric is highly dependent on a good, high density model and Floyd's algorithm in order to perform well.

The model based metrics does, except for the Tipping metric, provide a superior performance compared to standard metrics, like the Euclidian and Mahalanobis. It is noted that the

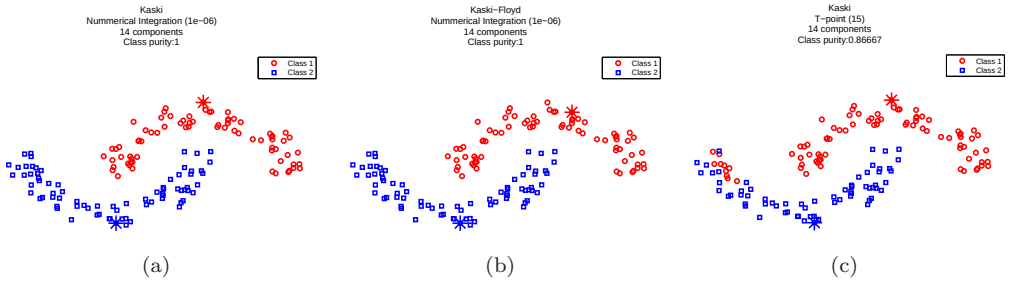


Figure 4.15: Example of Fisher/Kaski metrics. (a) Basic Kaski/Metric: Seen to cluster perfectly with the "true" metric (numerical integration).(b) With Floyd, obviously the same (c) T-point (15) approximation of Kaski/Metric without Floyd, illustrating the lower performance of the T-point approximation. Applying Floyd to this case, results in perfect clustering (see table) which is consistent for reasonable models.

mixture model fitted does have a relatively large influence on the results. It is especially noted that the numerical evaluation of the Rattray metric shows that the individual model determines the purity of the cluster. However, applying Floyd is justified using the Rattray metric leading to a perfect clustering regardless of the spurious clustering provided by the non-Floyd approach.

By the use of the purity measure it is noted that the performance of the T-point approximations in general shows equal performance when considering the best result obtained. In terms of standard deviation, the different T-point approximations also provide comparable results. However, one important example of better performance is obtained by the use of a T=15 approximation, i.e. when the model complexity seems to degrade robustness of the 1 and 5 point for the Fisher/Kaski metric at K=16. This indicates that higher dimensionality and hence more complex models will depend on the T-point approximation applied.

The analytical approximations provided by Tipping and Rattray, does with a few exceptions, provide lower purity of the clusters, which is quite disappointing, since a 1-point approximation is generally just as fast in a 2D situation as the evaluation of the error function necessary in the approximations. A final conclusion can again only be based on the individual data set.

The Fisher/Kaski metric is of course dependent on the model complexity and the basic T-point approximations generally favoring a quite complex model, but when applying Floyd to the Fisher/Kaski metric the model complexity plays a crucial role in the sense that a too complex model totally degrades the performance. Hence, the Floyd algorithm seems applicant only to the approximations when a reasonable model size is considered (here K=14 and K=16).

In conclusion: The Rattray and Fisher/Kaski metric provides superior performance compared to basic distance functions, with Kaski providing perfect clustering for the numerical integration over all models. However, generally highly dependent on the model for a perfect T-point performance This is the case for both basic T-point and and Floyd cases. Rattray is dependent on the Floyd's algorithm to provide the basis for the density formulation to hold, i.e. a connected high-density area. Tipping in general performs quite poorly on this data set.

Clustering with metrics II: Simple Gaussians

This toy data set consist of three pre-defined classes, created by 4 partly overlapping gaussian components (i.e. two are in the same class) as seen in figure 4.16(a). While the curved set was considered at a large range of component complexity, is the purpose here mainly to examine the metrics when a supervised approach intuitively should perform better, due to the gap between same class components. The data is furthermore generated to show the properties of the metrics versus the standard Euclidian like distances in situations where the individual directions of feature space have a profound influence on the clustering.

The training of the density models is performed on a 400 sample data set, and tested via clustering on another 200 point set. The BIC-optimal model of five is selected as representative.

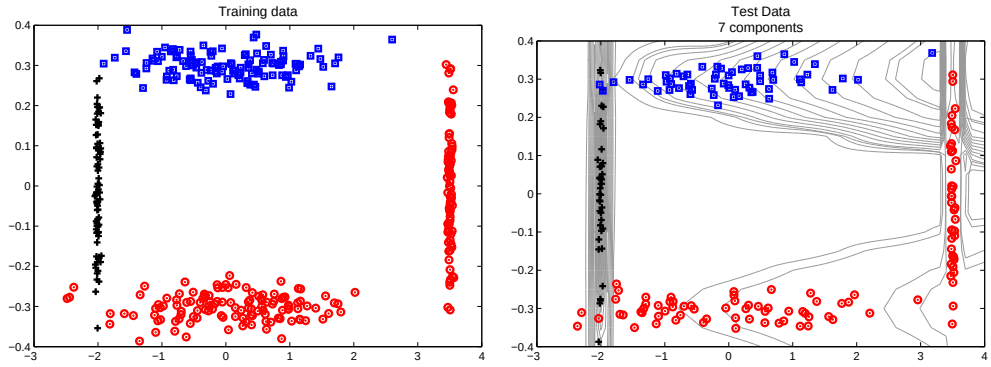


Figure 4.16: Simple Gaussians: (a) Training data. Notice the scale of the axis and the fact that the clusters are overlapping. (b) Example model fitted with 7 components resulting in a quite complex decision border which has been left out of illustration purposes. The gray contour lines depicts the changes in $p(y|x)$ (logarithm applied first)

The resulting clustering has primarily been evaluated using only the true number of clusters, i.e. three, using the purity measure, since this is believed to provide the best insight into especially the supervised metric. However, a few examples using a cluster per component is covered later in this section.

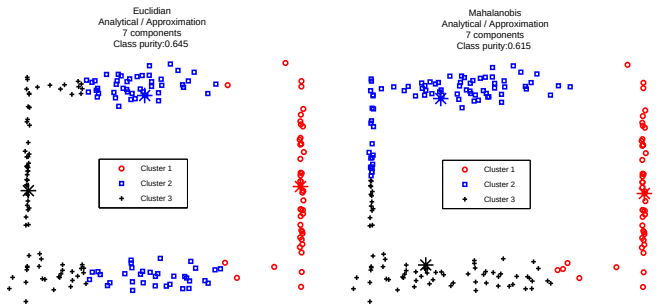
A number of selected results are shown in the table and illustrated in figure 4.17 to figure 4.20 for 7 components, which together with the accompanying comments covers the more interesting results.

A large number of interesting points can be drawn from the tables and examples, but only the main point will be mentioned.

The results generally shows that the true Fisher/Kaski (numerical) metric provides the marginally better result overall, considering the maximum purity and standard deviation obtained, however the simpler unsupervised metrics do in certain cases - although all highly dependent on the initializations - provide just as good results. This is is not surprising since the Tipping metric in essence is a locally weighted covariance metric, which for this data set is just what is needed. The model furthermore seems to provide the required connected, high density areas for the Rattray metric to perform well in the case of proper initialization.

K=7	Analyt	Num (1e-6)	T=1	T=5	T=15
Euclidian	0.62±0.12 0.67				
Mahalanobis	0.74±0.064 0.78				
Tipping	0.84±0.13 0.96	0.76±0.11 0.9	0.8±0.11 0.9	0.77±0.13 0.9	0.76±0.12 0.9
Tipping-Floyd	0.85±0.13 0.94	0.76±0.12 0.93	0.84±0.12 0.92	0.8±0.13 0.92	0.8±0.13 0.92
Rattray	0.84±0.081 0.89	0.69±0 0.69	0.71±0.093 0.85	0.7±0.08 0.85	0.74±0.077 0.84
Rattray-Floyd	0.81±0.093 0.86	0.78±0.14 0.96	0.79±0.1 0.96	0.7±0.15 0.96	0.74±0.063 0.96
Kaski		0.97±0 0.97	0.89±0.12 0.95	0.89±0.12 0.96	0.93±0.02 0.94
Kaski-Floyd		0.77±0.18 0.97	0.42±0.098 0.62	0.43±0.1 0.62	0.44±0.11 0.62

Table 4.4: Simple Gaussians: K = 7. Purity of the classes over 10 different K-means initializations including the maximum obtained (as second row)



(a) (b)

Figure 4.17: Example of basic metrics. (a) Euclidian. The very stretched components are too much for the Euclidian metric and despite being fixed to the data vectors, crosses the gap. The performance can be argued being worse than the purity value, due to this crossing. (b) Mahalanobis, does to some extend help on this set, however due to the three clusters the performance is limited (see later for example using four clusters)

However, due to the high-density areas, Rattray often finds the wrong configuration as shown in the examples and seen on the standard deviation.

One important factor in this context, is the sensitivity to the initializations in which only the basic Fisher/Kaski metric shows consistent results, due to the supervised approach as expected. In regards to the Fisher/Kaski metric it is noticeable that the Floyd shortest path approximation, does not generally help the T-point approximations, as seen in the illustration in figure 4.20(c).

Like in the curved data example above, does the T-point approximations provide quite consistent, best results independent of the number of points applied. However, the sensitivity to initialization is larger on this data set favoring a T=15 point approximation for the Kaski metric, which is due to the added number of decision boundaries and hence changes in the gradient of $p(y|\mathbf{x})$.

The cluster configuration with 3 cluster, was chosen to reflect the supervised metrics ability

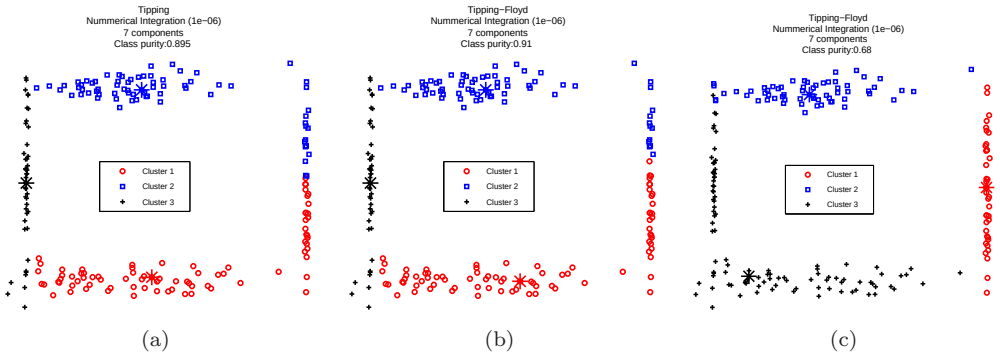


Figure 4.18: Example of the Tipping metric. (a) Basic Tipping. Showing that a local covariance weighting improves the global weighting performed by the Mahalanobis. (b) Tipping with Floyd does not improve the result significantly. (c) Example showing that the Tipping metric can provide a very pure clustering, however it does not reflect the original classes, and the purity is penalized. In general this calls for a supervised approach due to the gap between the class consisting of two components

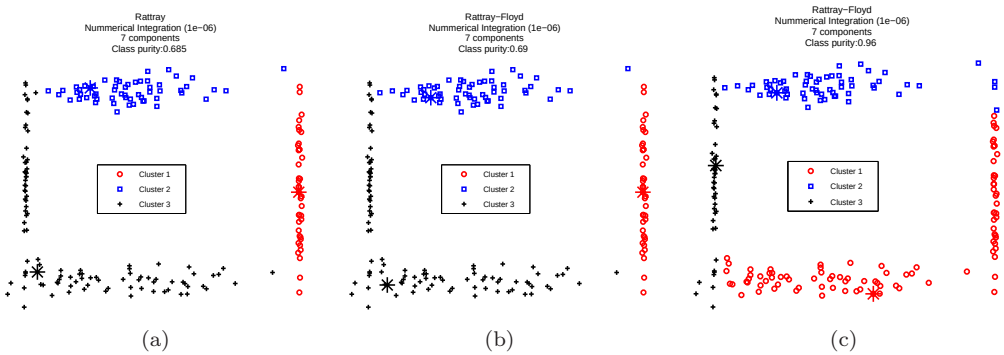
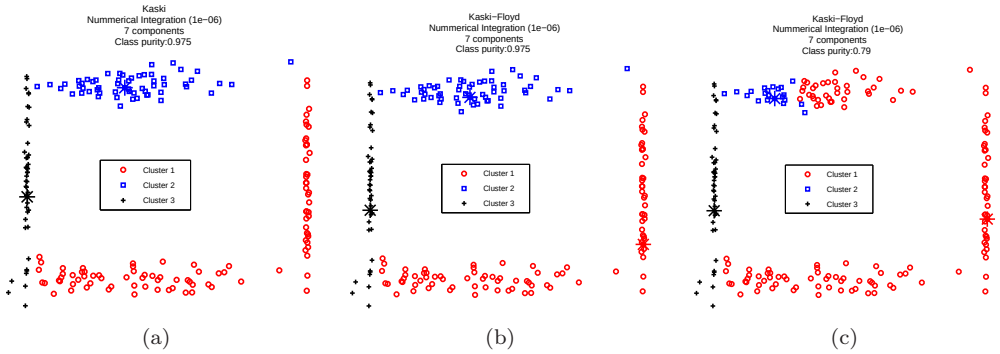


Figure 4.19: Example of Rattray metric. (a) Basic Rattray. At this example shows the lower performance of the Rattray metric. The purity of the clusters could be argued to be high however since the clustering does not reflect the original classes it is reduced, which shows the need for a supervised approach on this data set for consistent performance (b) Rattray with Floyd, not improving the basic result (c) An example of the Rattray metric with Floyd actually reflecting the true classes, however this is not generally the case as seen from the aggregated results

to handle such a situation, hence the evaluation was also based on the assumption that the clusters should reflect the true classes (the purity is reported in this assumption). However, this may not necessarily reflect the unsupervised metrics ability to generally find such stretched clusters and therefore a configuration with 4 clusters has been evaluated. A few examples shows how the metrics behave in such a situation, where the number of cluster does not match the defined ones. A more elaborate evaluation could potentially be performed using hierarchial clustering, in which the clusters are combined, but this is beyond the purpose and reasonable interpretation of this small example.

In conclusion: As shown using the curved data set the metrics are of course dependent on the model size, which makes the performance quite susceptible to overfitting etc. The learning metrics provide superior performance over the basic Euclidian and Mahalanobis. The use



(a) (b) (c)

Figure 4.20: Example of Fisher/Kaski metrics. (a) Basic Fisher/Kaski metric. Perfect clustering (b) Fisher/Kaski with Floyd. Again providing perfect clustering, however this is only due to the numerical integration. As the aggregated results shows, does the T-point approximations degrade the performance dramatically when applying Floyd. This is mainly due to the case where the inter-point distance between two points (or more) of originally different clusters is not calculated correctly, due to a very sharp change in $p(y|x)$. This causes the Floyd algorithm to perform very poorly since a short distance can now be constructed to another class, which causes the inclusion of many point in the other cluster if no significant $p(y|x)$ change is present within this cluster (see 4.16(b). (c) Example showing the problem of applying Floyd to the Fisher/Kaski metric on this data set, when using less precise T-point approximations.

of Floyd using the Fisher/Kaski metric and T-point approximations is quite destructible, as noted in the figure captions. The benefits of using a better T-point approximation for the Fisher/Kaski metric was more predominant on this set (shown on the sensibility to initializations).

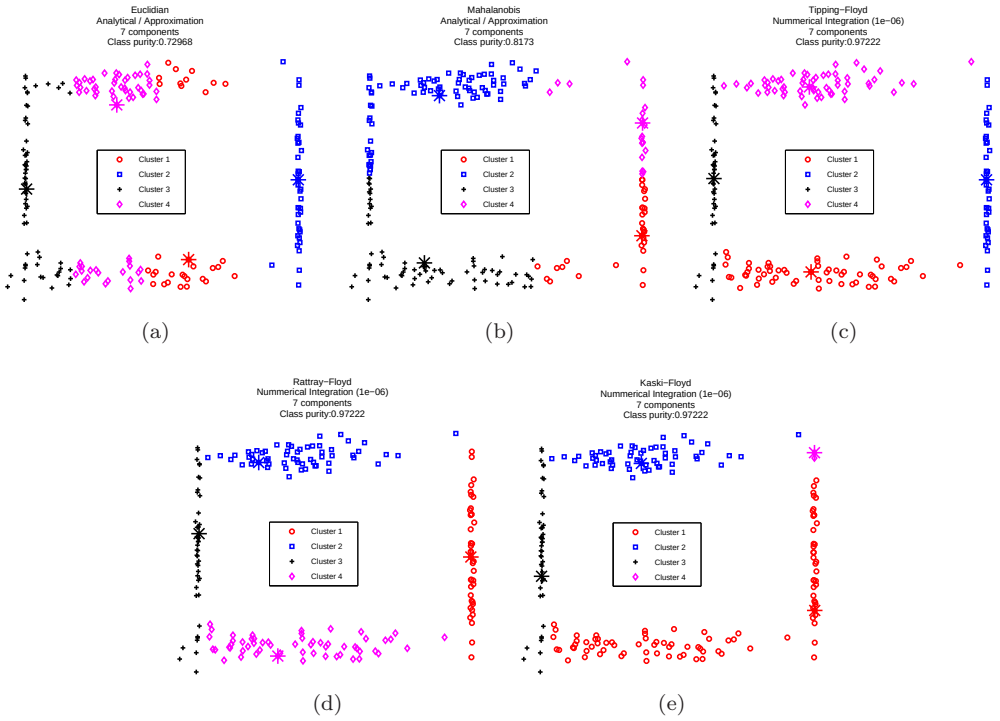


Figure 4.21: Clustering with 4 clusters. (a) Euclidian. Shows problems due to the stretched clusters (b) Mahalanobis. Seems to provide a some what better results than the Euclidian due to the global weighting. (c) Tipping-Floyd. Works very well, since the four clusters matches the stretched configuration. The basic Tipping does also perform quite good. (d) Rattray-Floyd. Same as Tipping providing a almost perfect clustering, given four clusters. (e) Kaski-Floyd. As seen in other Kaski-Floyd results, does the metric have a tendency to reduce one cluster to a minimum, which obviously is a good result and in a general exploration application such a behavior might reveal a with-in class cluster - effectively providing some sort of exploration

Clustering Real-world data set: The Phoneme Dataset

The phoneme data set considered here, is based on Finnish natural speech, in which 20 phoneme classes are identified and described by there mel-frequency cepstral coefficients in twenty dimensions. The phoneme data set originally consisted of two times 1828 data points, intended as one set for training and one for testing. In the context of examining the properties of the metrics via clustering, the data set is pruned to only 13 reasonable sized classes and ten dimensions. Further more the test set is halved in order to minimize the computational time[10] needed for the experiments.

The training is based on the best (BIC-wise) of five models, and only a 13 and a 20 component model is considered using a 5 and a 10 T-point approximation. However, in this case, both a diagonal and a full covariance model is considered in order to evaluate the effect of model structure/complexity in a data set with some relations to the music set of interest. The results are reported using both the purity and the Jaccard coefficients. The visualization of the "clustering" is done through a distance matrix showing the various cluster structures and inter-point distances for the best obtained result (Fisher/Kaski with diagonal, T=10) in figure 4.22

K=13	**Diagonal**		**Full**	
	T=5	**T=10**	**T=5**	**T=10**
Euclidian	0.72±0.033 (0.77) **0.54±0.053 (0.59)**			
Maha-lanobis	0.67±0.04 (0.75) **0.34±0.021 (0.37)**			
Tipping	0.77±0.031 (0.83) **0.59±0.11 (0.73)**	0.76±0.029 (0.8) **0.58±0.1 (0.72)**	0.75±0.03 (0.79) **0.61±0.098 (0.73)**	0.74±0.029 (0.78) **0.62±0.11 (0.74)**
Tipping-Floyd	0.77±0.03 (0.83) **0.59±0.11 (0.73)**	0.76±0.029 (0.8) **0.58±0.1 (0.72)**	0.75±0.03 (0.79) **0.61±0.097 (0.73)**	0.74±0.029 (0.78) **0.62±0.11 (0.74)**
Rattray	0.67±0.038 (0.71) **0.28±0.031 (0.34)**	0.68±0.029 (0.71) **0.3±0.027 (0.34)**	0.64±0.024 (0.68) **0.44±0.042 (0.49)**	0.69±0.02 (0.71) **0.44±0.037 (0.5)**
Rattray-Floyd	0.69±0.04 (0.77) **0.3±0.032 (0.35)**	0.71±0.045 (0.76) **0.31±0.019 (0.34)**	0.64±0.053 (0.72) **0.39±0.065 (0.48)**	0.69±0.033 (0.73) **0.42±0.062 (0.51)**
Kaski	0.75±0.03 (0.8) **0.7±0.1 (0.76)**	0.76±0.016 (0.79) **0.71±0.087 (0.76)**	0.68±0.036 (0.73) **0.39±0.024 (0.43)**	0.74±0.04 (0.78) **0.63±0.13 (0.74)**
Kaski-Floyd	0.74±0.035 (0.8) **0.65±0.13 (0.76)**	0.78±0.018 (0.8) **0.7±0.087 (0.76)**	0.75±0.041 (0.84) **0.26±0.05 (0.34)**	0.74±0.025 (0.79) **0.4±0.054 (0.47)**

Table 4.5: Phoneme clustering, 13 components: Both purity (first row) and Jaccard coefficients (second row) are shown, with the Jaccard being in bold letters. The mean and standard deviation over 10 random initializations are shown along with the maximum value obtained (in parentheses). See main text for comments

A few very notable results can be obtained from the tables considering the Jaccard coefficients, i.e. the correspondence between the true classes and the found clusters.

The Rattray metric is first of all very poor on this data set, and performs worse than the Euclidian distance in terms of the Jaccard coefficient. But looking at the purity gives a different answer, in which all measures (except a few) provides fairly pure clusters, but as mentioned, this is not the objective with this example. One problem with the Rattray formulation is the assumption of a high density area and considering this data set with many different sizes of clusters may violate this assumption, however this has not been considered further.

[10]The distance calculations on this data set has been performed individually on a x86 2.8 GHz machine still amounting 72-120 hours, dependent on the model size/complexity and approximation!

K=20	Diagonal		Full	
	T=5	**T=10**	**T=5**	**T=10**
Euclidian	0.72±0.033 (0.77) **0.54±0.053 (0.59)**			
Maha-lanobis	0.67±0.04 (0.75) **0.34±0.021 (0.37)**			
Tipping	0.77±0.016 (0.79) **0.63±0.099 (0.75)**	0.75±0.033 (0.79) **0.63±0.077 (0.73)**	0.74±0.017 (0.78) **0.64±0.088 (0.72)**	0.75±0.021 (0.79) **0.59±0.1 (0.74)**
Tipping-Floyd	0.77±0.016 (0.79) **0.63±0.099 (0.75)**	0.75±0.033 (0.79) **0.63±0.077 (0.73)**	0.74±0.02 (0.78) **0.64±0.084 (0.72)**	0.76±0.026 (0.79) **0.58±0.1 (0.74)**
Rattray	0.65±0.053 (0.74) **0.3±0.036 (0.35)**	0.71±0.029 (0.74) **0.32±0.025 (0.36)**	0.64±0.02 (0.67) **0.44±0.047 (0.49)**	0.66±0.026 (0.69) **0.45±0.061 (0.53)**
Rattray-Floyd	0.67±0.042 (0.72) **0.29±0.02 (0.31)**	0.73±0.027 (0.77) **0.33±0.021 (0.36)**	0.65±0.031 (0.7) **0.41±0.058 (0.49)**	0.64±0.04 (0.71) **0.47±0.067 (0.55)**
Kaski	0.8±0.021 (0.83) **0.72±0.056 (0.76)**	0.82±0.022 (0.85) **0.71±0.062 (0.78)**	0.63±0.039 (0.68) **0.43±0.068 (0.52)**	0.72±0.026 (0.75) **0.69±0.082 (0.76)**
Kaski-Floyd	0.76±0.028 (0.79) **0.57±0.043 (0.62)**	0.81±0.03 (0.85) **0.7±0.065 (0.77)**	0.68±0.015 (0.7) **0.22±0.02 (0.25)**	0.68±0.039 (0.73) **0.42±0.064 (0.5)**

Table 4.6: Phoneme clustering, 20 components: : Both purity (first row) and Jaccard coefficients (second row) are shown, with the Jaccard being in bold letters. The mean and standard deviation over 10 random initializations are shown along with the maximum value obtained (in parentheses). See main text for comments.

Disregarding the Rattray metric and looking at the Fisher/Kaski metric, a quite distinct difference in robustness to initializations is obtained, favoring the diagonal covariance structure in almost all cases. The model complexity is also noticed when evaluating the approximations, in which case, a T=10 point is required for the full covariance case. The difference between the 5 and the 10 point approximation is quite insignificant using a diagonal covariance matrix. Despite the diagonal case obviously providing a more robust clustering, the full covariance is still capable of achieving almost the same Jaccard ratio (74 vs. 76 for the 20 component case)

The Tipping measure is quite interesting in the sense that it is capable of achieving good maximum results, but it is considered rather un-robust across initializations. The relatively large difference between the Jaccard and the purity measure for the Tipping metric, reflects the unsupervised nature of this metric.

In conclusion: The phoneme set was used to evaluated the metrics in a real-world setting. It was shown that the supervised metric provides the better result in terms of the correspondence between the true classes and the found clusters, as expected. The example furthermore showed that the larger model slightly improves the performance of the Fisher/Kaski metric, although, the optimal model is not claimed to be found. The full covariance model provides a more complex decision boundary and the T=10 approximation provides the better results in this case.

As seen in the simpler examples, does the Floyd algorithm not necessarily help the metrics if the assumptions made in the formulation do not hold, e.g. a good model with connected high density areas for the Rattray metric and in general, a precise distance approximation, and this is not generally improved by the removal of half the data set for the metrics highly dependent on connected density areas.

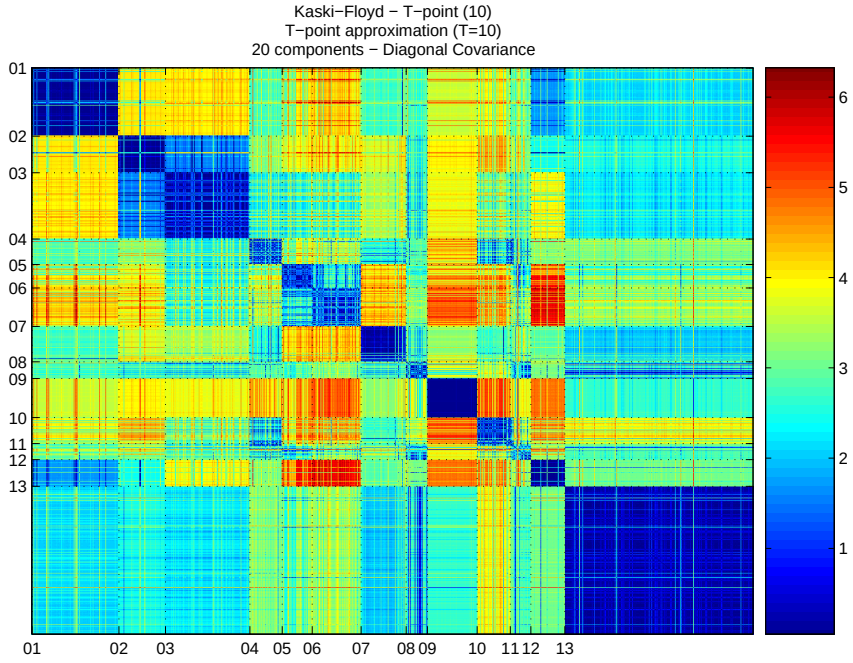


Figure 4.22: Example of distance matrix for the phoneme set. Shows the distance matrix for the Fisher/Kaski metric with a 20 component model, a T=10 point approximation and diagonal covariance matrix.

4.7 Metric Learning vs. Related Methods

While the intuitive feeling of the Euclidian distance is fairly clear - at least if the features have some degree of conceptual meaning - the use of a local metric describing the local variance or relevance can seem quite distant to other machine learning methods.

The focus in this thesis is on the explicit estimation of the local metric, $\mathbf{J}(\mathbf{x})$ and $\mathbf{G}(\mathbf{x})$, where the supervised metric $\mathbf{J}(\mathbf{x})$ was based on a local approximation to the Kullback-Leibler (KL) divergence. However, it is possible, as described in [24], to formulate a other supervised approach aimed at clustering, named discriminative clustering (DC) [24]. This approach is based on a direct minimization of a KL-divergence based cost function, but such an use of the principle will not be used here, since an explicit estimation of $\mathbf{J}(\mathbf{x})$ is believed to provide better insight into the local properties of the feature space. Furthermore, can the use of the explicit measure, $\mathbf{J}(\mathbf{x})$), in the K-means algorithm also be seen as a discriminative clustering technique [24], obtaining similar abilities as the formal formulation of DC.

Other approaches with objectives relevant to the metric principle, includes the use of linear and non-linear projections methods. Some of these include the option of a supervised labeling, such as partial least squares (PLS) and canonical correlation analysis (CCA) (very close related to PLS), while others are purely data driven, like principle component analysis (PCA). CCA, for example, is intended at analyzing associations between two sets of variables in terms of the cross-covariance between the two sets, leading to the concept of a latent subspace, in which the maximum cross-covariance is obtained. This in essence results in a possible projection of the first variable, i.e. the primary data, onto this subspace for further exploration in terms of both the labeled and true data space (see e.g. [30]). However, as with other projection methods this tend to destroy the meaning of the original space, assumed to be of crucial importance in the formulation of the Fisher/Kaski metric.

A supervised non-linear approach, which has already been applied to the music similarity task is the use of neural networks for a (supervised) non-linear mapping into a so-called anchor space [4] for further similarity estimation using a potentially simple measure. However, one issue is the interpretation of such a neural network, which is not always trivial due to the generally complex structure of these highly non-linear networks (see e.g. [6]).

The use of purely supervised methods for distance measures could be based directly on the conditional class probability, $p(y|\mathbf{x})$, and may seems quite obvious in some application. Certainly compared to the rather involved task of path integration of this conditional probability distribution. However, as already argued: it does not provide a distance directly in relation to the original feature space like the supervised Fisher/Kaski metric does. In some cases a classification based measure will even give a totally different response due to the crossing of several decision borders, as in the 1D example describe in this chapter. This will later be described for a hypothetical music situation. Depending on the task, a classification approach may be preferable, such as genre classification for example, for example. But when the objective is to locate similar songs based on a combination of both the defied relevance and the signal contents, one might prefer the option of a exploratory metric like the Fisher/Kaski formulation. An unsupervised alternative is then given by the Tipping and Rattray metrics, obviously depending on the task and feature space.

Projection methods and variations do often provide either dimensionality reduction and/or possibly a "simpler" feature space for a similarity measure, however they do not in general preserve the topology and in some sense destroys the meaning of the originally formulated

features. This is one of the reasons for formulating only perceptually motivated features in this project - in which case the Riemannian metric formulation can be considered an ideal choice for exploratory analysis; but hardly the first option considered, though.

One major problem using the metric approach, is the computational complexity due to the density estimation and the general formulation of the metrics, which will be discussed in the specific real-world context of music similarity (chapter 6)

4.8 Summary

In this chapter, the well-known Gaussian Mixture Model was described, for use in modelling music data. A supervised variant was presented based on modelling the joint probability of the data and labels i.e. $p(\mathbf{x}, y)$.

Considerations about practical training, in terms of overfitting and model selection was described, and a general setting using covariance regularization and early stopping was adopted as the predominant way of avoiding overfitting. The derivation of the Bayesian Information Criterion, was outlined and the BIC measure will be used in the actual fitting of music. In particular it is the hope that the use of BIC can improve well-known ways of music retrieval, described in chapter 5 and 6.

An alternative machine learning techniques was presented, based on the objective to provide better clustering and/or construct a more explorative datamining method. First two purely unsupervised metrics were described, which were able to locally weight the contribution of feature directions based on a Gaussian Mixture Model.

Furthermore a supervised approach was derived based on an already described approach by Kaski et al [14] extended with the option of a full covariance model for the gaussian mixture model. The supervised approach is based on estimating the Fisher Information Matrix formulated in terms of an indirect classification of data in feature space, i.e. the conditional probability $p(y|\mathbf{x})$.

While the formal, theoretical formulation of the three metrics themselves, i.e. $\mathbf{G}(\mathbf{x})$ and $\mathbf{J}(\mathbf{x})$, results in a trackable solution using the mixture models the required path integral adds a rather complicated dimension to the appealing theoretical formulation. The path integral was approximated for the two unsupervised metrics based on previous work, to yield a constant metric along a Euclidian straight line approximation. Furthermore, the gain of performing integration along a geodesic path was evaluated using a dynamic programming algorithm, namely Floyd's algorithm for a shortest-path search.

The approximations mentioned are all based on an engineering approach and the performance will in the end depend on the data considered. It is demonstrated that all metric based distances are capable of clustering data better than both the Euclidian and Mahalanobis distance - although quite dependent on the data set and model. The supervised metric showed superior performance across data sets, as expected, and on the audio data set provided some initial results on data relatively close to the one described in chapter 2 and 3.

Chapter 5

Music Similarity

A main motivation of this thesis is the examination of different ways of comparing and exploring music, i.e. formulated as finding a suitable similarity measure based on the two perceptually motivated feature types (Mel-frequency Cepstral Coefficients and two dominant pitches).

In the last few yeas a great deal of research and experimentation has gone into examining various aspect of music classification and retrieval. One of the main areas of interests is genre classification which has been the driving force in audio mining for quite a while. The results have only within the last few years shown to exceed 70-80% correct classification on a reasonable data set, which compared to for example speaker and speech recognition is a bit on the low side.

The tools used in this context spans from simple linear classifiers, K-Nearest neighbour to general non-linear models like Gaussian Mixture Models and neural-networks. Lately a discriminant classifier, the Support Vector Machines have shown to produce quite reasonable results on the good side of 70%.

As previously mentioned, does this thesis - and subsequent this chapter - not take the genre classification viewpoint defined as a hard assignment to a class in which the audio track is compared directly to a predefined genre (or artist). Although not totally detached from genre classification, the viewpoint is of a more exploratory nature, in which the clips and songs are compared to each other, i.e. the outcome will be a similarity measure. This measure can obviously be used in post-processing to do a K-Nearest Neighbor classification. The difference between the pure classification and explorative viewpoint should be seen in the quest for something that sounds similar and a desire to make exciting discoveries in the music which e.g. can be used to create a map as the example shown in figure 5.1 based on the idea of an *Island of Music* concept [23].

A variety of existing measure and distance functions have previously been examined in this context, spanning from simple Euclidean and Mahalanobis distances in feature space to

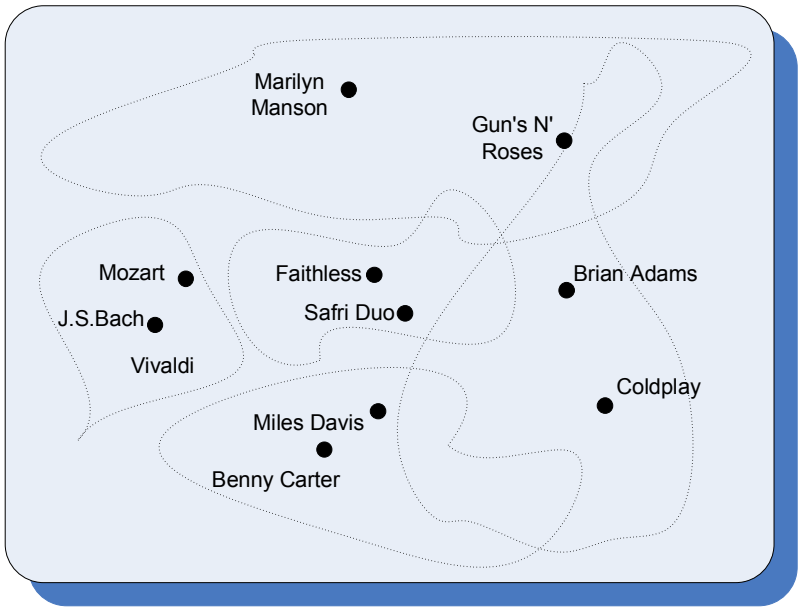


Figure 5.1: Example of one version of a music map. The artist are indirectly classified in terms of their genre defined in the appendix. The map reflects both the challenges in terms of the potential borders of i.e. genres, but is also illustrates the task of defining which is closer. If considering the Euclidian distance in example map we end up with *ColdPlay* being slightly closer to Brian Adams than *Gun N Roses*. One of the ideas in this thesis is to use the Riemannian metric defined as the basis for constructing these distances, but in the true feature space, retaining the meaning of the varios directions.. $\mathcal{X}$

information theoretic measure like the Earth Mover Distance and Kullback-Leibler (see e.g. [5]). Regardless of the final measure, a major trend in the music retrieval community has been to use a density model of the features (often timbre space defined by MFCC's), like presented in chapter 4. The main task of comparing e.g. two models has then been handled in different ways and is obviously the more interesting task. The trivial case of a single multivariate gaussian fitted to e.g each song (many data points) does call for the, in this context natural measure, namely the Kullback-Leibler divergence.

Aucouturier and Pachet [3, 2] suggests using the computational expensive sampling of the likelihood of one song given the other, which will also be used on the expanded feature set in this thesis. Pamptak [19] furthermore suggested using a vector quantization approach in which a signature of the song is described in terms of the cluster centers of a K-Means model and the likelihood is then estimated based on the centroids of the clusters, i.e. the code words. Mandel and Ellis proposes an even simpler approach using only one Gaussian component and comparing them using the Kullback-Leibler Divergence (see e.g. [5]). This idea is quite fundamental, and will be used and a variation of the Kullback-Leibler divergence - the Divergence Shape Distance [19] used in speech analysis - will be applied to the custom data set.

A different technique quite relevant to this project is the so-called *anchor space*, already mentioned, proposed by Berenzweig et. al. [4], in which a supervised mapping of music features is performed through a multi-layer neural network. This could be considered a

preprocessing step, but does potentially provide a anchor space in which the similarity measure is simple to calculate. This idea is actually quite different from others in the sense that a direct non-linear mapping is performed on each data sample, into a new space. The non-linear projection is in [4] performed using a neural network. One of the disadvantages using such a technique is the interpretation of the structure of the network, i.e. the task of analyzing the reason for a data point being transformed into a certain region is not easily extracted. In a large scale evaluation and application this might - and is quite acceptable - but on a smaller scale it could potentially be an important property to know the exact reason for a song being closer to one and not the other e.g. artist. The anchor space mapping is suggested trained using a set of subjective measures, such as rhythm and melody, which is conceptually quite close to the multi-dimensional scaling mentioned in chapter 2.

The performance of all these former attempts, ranging from density models and non-linear mapping based on density modelling and direct comparison of models has several places been noted not to be particular impressive on reasonable data sets (a glass ceiling of about 65% R-precision exits, and most pessimistic is the work presented in [3], which takes about the glass ceiling. The goal of the similarity aspect of this thesis is as mentioned not to solve this problem and provide an all time best, but to provide some insight into the details of both model structure and behavior of these well-known similarity functions given some variations of models etc.

In order to provide an alternative to the traditional methods, will the last part of this chapter address the problem of retaining the original space in audio mining and the principle of Riemannian based metrics will be used, primarily based on the assumption that the benefit of retaining the original feature space, as shown in the previous chapter, is beneficial to the given task at hand - which it very well could be in music, where one issue is to determine which features differentiates e.g. *Brian Adams* from *U2*.

5.1 Information theoretic measures

Similarity can be defined in many ways, and in this section the focus will be on some of the proposals made relating to information theory, in which the entropy and Kullback-Leibler plays an fundamental role.

5.1.1 Kullback-Leibler

The Kullback-Leibler divergence is a very fundamental concept in information theory, due to its relations with mutual information and coding theory. It is defined by:

$$\mathcal{D}_{KL}(p_1||p_0) = \int p_1(x) \ln \frac{p_1(x)}{p_0(x)} dx \tag{5.1}$$

The divergence is however not symmetrical, i.e. $D_{KL}(p_1||p_0) \neq \mathcal{D}_{KL}(p_0||p_1)$, which is required by the general formulation of a metric, and various attempts have been made to make symmetric - or distance measures - based on the fundamental Kullback-Leibler divergence.

Although not intended as a distance measure, the KL-divergence was symmetrized by cal-

culating the average between the two possible divergences, i.e.

$$\mathcal{D} = \frac{D_{KL}(p_1||p_0) + D_{KL}(p_0||p_1)}{2} \tag{5.2}$$

In the special case of a gaussian probability distribution this symmetrized divergence can be written explicit as

$$\mathcal{D} = \frac{1}{4}\left[tr\left(\mathbf{C}_l^{-1}\mathbf{C}_k\right) + tr\left(\mathbf{C}_k^{-1}\mathbf{C}_l\right)\right] - \frac{M}{2} + \frac{1}{4}\left(\mu_k - \mu_l\right)^T\left(\mathbf{C}_k^{-1} + \mathbf{C}_l^{-1}\right)\left(\mu_k - \mu_l\right) \tag{5.3}$$

With M being the dimension and tr is the trace operator. The derivation of this quite fundamental result is presented in the appendix.

Recently it has been suggested using a reduced Kullback-Leibler distance coined divergence shape distance (see e.g. [19]) in which the shape of the distribution and not its mean is included. This results in the following

$$\mathcal{D}_{SD} = \frac{1}{4}\left[tr\left(\mathbf{C}_l^{-1}\mathbf{C}_k\right) + tr\left(\mathbf{C}_k^{-1}\mathbf{C}_l\right)\right] - \frac{M}{2} \tag{5.4}$$

One fundamental limitation relating to all variants of the basic Kullback-Leibler is the fact that no analytical solution exits for mixtures of distributions. This limits the possibility to model complex data directly. There are various techniques to overcome this, like the Earth Mover Distance described next, which will also be used in this thesis, and applied to the custom database.

5.1.2 Earth Mover Distance

The Earth Mover Distance (EMD) is an attempt to overcome the restriction of the basic Kullback-Leibler divergence. The Earth Mover Distance originates from the image retrieval community, and has proven to work well for image retrieval applications and was original proposed in [27].

The idea is to define the work required to "move" one distribution into the other. In terms of two Gaussian Mixture Models p and q the EMD can be formulated as follows.

$$EMD = \frac{\sum_{i=1}^{K}\sum_{j=1}^{L} f_{ij}d\left(p_i, q_j\right)}{\sum_{i=1}^{K}\sum_{j=1}^{L} f_{ij}} \tag{5.5}$$

where $d\left(p_i, q_j\right)$ is the ground-distance between component p_i and q_j. Furthermore $[F]_i j = f_i j$ is optimized as a basic linear programming problem, subject to the following constraints

$$f_{ij} \geq 0, 1 \leq i \leq K, 1 \leq j \leq L \tag{5.6}$$

$$\sum_{i=1}^{K} f_{ij} \leq w_{p_i} \tag{5.7}$$

$$\sum_{j=1}^{L} f_{ij} \leq w_{q_j} \tag{5.8}$$

$$\sum_{i=1}^{K}\sum_{j=1}^{L} f_{ij} = \min\left(\sum_{i=1}^{K} w_{p_i}, \sum_{i=1}^{L} w_{q_j}\right) \tag{5.9}$$

Futhermore

$$w_p = \sum_{i=1}^{K} w_{p_i}, w_q = \sum_{j=1}^{L} w_{q_j} \tag{5.10}$$

The optimization of $[F]$ is done with use of a standard simplex algorithm which is provided by the author of [27].

The ground distance d, can in principle be any positive metric between two components, however dealing with individual Gaussian components, the natural choice is the Kullback-Leibler divergence, and since this thesis only considerers similarity measures in which the symmetry property holds, the symmetrical version of the Kullback-Leibler divergence defined in 5.3 is taken to be the ground-distance $d(p_i, q_j)$ between components. Furthermore the EMD is extended with the Divergence Shape Distance, which to the authors knowledge has not be examined on a music set before.

The EMD has previously been used in the music retrieval community by Logan and others [18, 5]. In [5] EMD variations using various model complexities were compared, and it was concluded that a simple K-means training with a diagonal covariance matrix, yielded better results than a more advanced EM training with full covariance matrix. Such an investigation is of course interesting in the sense that it relates to the robustness of the metric.

The full covariance model using the EM algorithm has, however, been used in several image retrieval applications (e.g. [32]), although often using the asymmetric Kullback-Leibler divergence, which in turn results in an asymmetric distance function i.e. $d(p||q) \neq d(p|q)$), which is not considered here. A full covariance structure is examined for the EMD distance using the KL divergence and is compared to the diagonal case on the custom data set, and the outcome of using the full covariance model is dependent on the feature space provided by the MFCC's and pitch.

While e.g. [5] and [18] only considers fixed model sizes for the EMD, a variable model size for each item has previously been used in image/texture classification (e.g. [32]) using the EMD, which will also be examined on one example in this project, through empirical experiments by the use of the Bayesian Information Criterion as the selection criterion.

5.2 Cross-Likelihood Ratio

A very popular method in the music retrieval community is to describe each song by a density model (often a Gaussian Mixture Model), and then compare the log-likelihood of one song given the other and vice versa. This is known as the (symmetric) Cross-Likelihood Ratio defined as:

$$d(\mathcal{M}_A, \mathcal{M}_B) = L\left(\mathcal{X}_A|\theta_A\right) + L\left(\mathcal{X}_B|\theta_B\right) - L\left(\mathcal{X}_B|\theta_A\right) - L\left(\mathcal{X}_A|\theta_B\right)$$

Where $\mathcal{M}_n$ is the model (of a clip/song) with the parametrization $\theta_{\mathbf{n}}$

The likelihood of the data can in principle be calculated from the real data - but given the often large databases a sampling is performed for each song (often 1500-2500 samples per song). Since most results [3, 5] are reported using a sampling approach, this approach will also be applied in this project, mainly to evaluate the effect of the pitch inclusion and to provide a reference for the metric based method describe in the next section.

5.3 Metric Based Retrieval & Datamining in music

The traditional audio similarity measures described above are all based directly on the individual density model and does, as described, consider the songs in a one-to-one comparison of the distribution and hence does not include the potential influence of another song/clip which might be close by in feature space. This roughly means that we are comparing density models not songs - although these are of course very tightly linked.

As mentioned several times, does this project consider a more local approach in which the effect of other items, e.g. songs, are included based on the local topology of the feature space (see chapter 4). Obviously such a metric requires a point of reference, which in the clustering examples was the centroid of the cluster and each individual data point. In the case of exploration by retrieval, which this thesis is limited to, such an individual approach is not applicant since all inter-point distances would have to be calculated (for this data set we are dealing with 1000 points pr clip). Several techniques exist of solving such a problem, of which the simplest is to represent the clip by its empirical mean, which is also the approach applied in the initial experiment considered in this thesis. So the final comparison between clips becomes a vector comparison of the vector pair $\{\mathbf{x}_{clip1}, \mathbf{x}_{clip2}\}$

Formally this is considered a very rough vector quantization in which the next step would be to use a better quantization consisting of more representative (more than one) so-called code words for each clip, leading to more points of reference. Such a vector quantization is usually performed using the K-means algorithm, but we have already enforced a mixture model on the training data, and the code words (in the form of centers) defined for this potentially supervised model, can be used as points of reference if the relationship between songs and component centers is known, e.g. $p(y|\mu_k)$. However, only the empirical mean approach is taken in the experiments provided in the next chapter, due to the novel approach of music similarity which needs to be examined in simple terms, but this could easily be extended to more points using e.g. vector quantization.

A quite similar approach to the use of metric learning in audio is as already mentioned to use e.g. a neural network method or potentially another projection method, like partially least squares or canonical correlation analysis; however this is not explored in-dept here. Such a projection could obviously be put into the framework of learning metrics for the audio case, and clips can again be compared using a vector quantization approach and metrics.

Another approach which might be considered also in similarity and mining applications is the use of a conditional probability comparison, perhaps formulated as a genre classification task. Considering a classification or using the continuous probability $p(y|\mathbf{x})$ as a measure of how close two songs are based one some model, we generally obtain the similarity only in relations to this model and the objective which underlies the training, which could be genre classification, but two songs can in theory be in the same genre without being close in feature space, which again relates to the principle of topology preservation. Consider the example in figure 5.2, in which a number of distributions - possible songs - are located in such a way that they are disconnected in feature space. A pure classification approach, e.g. using the mixture model, will result in all A-songs being classified as such. However using the supervised metric to calculate the similarity from A1 to to A3 will result in a double crossing of a decision boundary (assuming a straight line approximation, otherwise the B-circle must be entirely closed). This will lead to A1 being closer to B1 then A3 - but A1 is still closer to A2 than B1, which makes perfect sense if considering the local features. Using a Euclidian metric will also lead to B1 being closer to A1 than A3, but unlike the

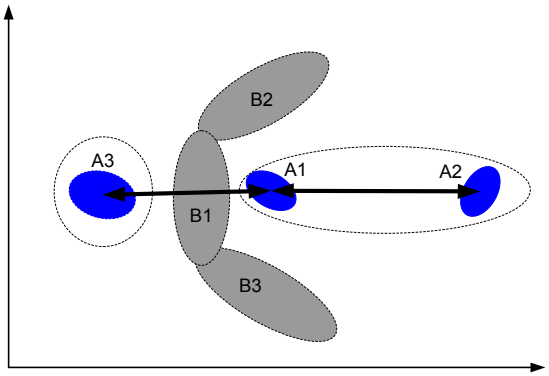


Figure 5.2: Example on the difference in a highly hypothetical situation. Similar aspects of this situation is actually present in the curved data set used for clustering, since a straight line approximation can risk crossing two decision borders leading to a "exploration" in terms of the features

Fisher/Kaski metric will A2 be further away than B1.

The example should only serve as motivation for exploring the metrics described in this thesis. The application of the metrics and especially the Fisher/Kaski metric can be extended much further than the simple retrieval task, on which the metrics is applied/evaluated - and possibly not optimized for. It was for example shown, through the clustering examples, that the Rattray metric was quite dependent on a quite good model and the Floyd algorithm, but using an extreme vector quantization we effectively remove this options and the Rattray metric hence the Rattray metric is not expected to perform well in a simple retrieval situation. The Tipping metric on the other hand shows a quite robust behavior and may be considered a local Mahalaobis distance which could potentially prove efficient as music similarity metric.

5.4 Summary

A number of well-known methods for comparing music in terms of similarity based on density models was described in this chapter, mainly the Kullback-Leibler based methods, and the Earth Mover Distance for mixture models. The Divergence Shape Distance - as a special case of the Kullback-Leibler Divergence - lately applied with success to audio segmentation applications, was introduced. Furthermore a well-known method based directly on the log-likelihood of the individual data samples was presented, namely the Cross Likelihood Ratio.

The possibility of using the metric distances formulated and derived in chapter 4, section 4.5, was discussed, and a proposition of quantizing the audio clip to its empirical mean vector for a simple retrieval situation, will be demonstrated in the following chapter.

Chapter 6

Experiments

This chapter contains the results obtained in a "explorative" setting, defined as the task of music retrieval. The traditional divergence based similarity functions (divergence based and Cross-Likelihood Ratio (CLR) will be applied with the purpose of evaluating various parameter choices and the feature set, in terms of the pitch feature and the ongoing problem of an normalized feature space versus un-normalized features.

After as short introduction to the evaluation approach, are the results presented and discussed in the following order

Song Retrieval The data set presented in chapter 2 is fitted with models on the song level, i.e. a gaussian density model is fitted to each individual song, for the evaluation of the traditional measures and their variations described in chapter 5.

Clip Retrieval The properties of the metric based distances are examined on the data set, although only one genre has been considered in this setting. The results will be compared with the well-known methods described in chapter 5.

6.1 Evaluation Methods

One of the difficult issues in music datamining is the lack of ground truth for a number of applications. In this context we need some idea of when music pieces are similar.

The data will, within this project, be processed by the hieratical assumption that a song is always closer to album than to an artist and an album is closer to an artist than a genre etc. This provides some (perhaps wrong) ground truth, in order to evaluate the results, but considered an relative logical choice based on the construction of the custom dataset in chapter 3.

6.1.1 Recall/Precision & R-Precision

A commonly used evaluation method in Information Retrieval (IR) systems is the use of the *Recall and Precision* paradigm, somewhat standardized in connection with TREC (Text REtrieval Conference) [7]. Given a database with a arbitrary number of *documents* N. Moreover a set of relevant documents, R, is identified. The cardinality of R is $\mid R \mid$. The set of retrieved documents, A (answer set), is compared with the number of relevant documents in the retrieved set. Using the cardinalities of the sets we obtain the following semi-objective[1]

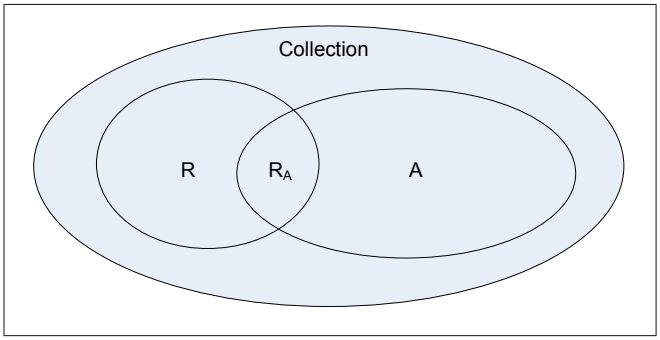


Figure 6.1: The concept of Recall and Precision. The set R, is the set relevant to a given query, and the set A is the set returned by a query.

evaluation measures,

$$\begin{aligned} \text{Recall} &= \frac{|R_A|}{|R|} \\ \text{Precision} &= \frac{|R_A|}{|A|} \end{aligned}$$

with $|R_A| = |A \cap R|$ - or in words

Precision The ability to retrieve top-ranked documents that are mostly relevant.

Recall The ability of the query to retrieve all of the relevant items in the collection.

These measures give different values depending on how many retrieved documents is needed/wanted for a given query, which results in a series of Recall/Precision results. In order to get an general result the values are averaged over all possible queries, and then plotted.

While the individual evaluation of queries in term of Recall/Precision does provide insight into the specific detail of the retrieved documents we define the final evaluation measure, namely the R-precision, which is defined a the Recall value obtain at the precision of which the number of relevant items (user defined) is retrieved.

[1] The term semi-objective is attached since the relevant set is still defined by human hand.

6.2 Results

This section provides the actual results obtained through a quite extensive investigation into the various properties of the traditional similarity measures, supplemented with initial results using the Riemannian based metrics. The investigation was initially based on the traditional audio similarity and density models with full covariance structure for retrieval of songs and clips. By using an initial data set, this setup was found to produce fairly robust results by selecting the BIC-optimal models among five trained, with same number of components.

This, however, proved to be an optimistic assumption in the full evaluation, especially for the EMD measure and the investigation was therefore extended with the option of a diagonal covariance matrix providing more robust results, as will be shown and further discussed. This obviously leads to a large variety of results of which the better are included in this section, while others are mentioned, but figures etc. are placed in appendix F.

In regards to the retrieval using the Riemannian based metrics only initial results will be presented due to the time frame of this project. The metrics will be evaluated in a clip retrieval setup and compared with the traditional measures.

6.2.1 Song Level Retrieval

The traditional measures, divergence based and Cross-Likelihood Ratio (CLR), are evaluated on the custom data set described in chapter 3. The following setups are considered:

Relevance The relevant set in the R-precision measure is defined as being the artist, i.e. the measure must return a song for the same artist.

Structure Covariance structure: Full / Diagonal

Size Number of components in the model (fixed for all songs)

Features Two feature set are evaluated, one using only the MFCCs and one also the pitch. Further more the normalization issues often considered in machine learning is again evaluated on the similarity measures.

BIC-selection across model sizes A BIC-selection suggestion is tested on the best obtained results from the above, in which the BIC-optimal model is selected for each song across all model sizes. This approach has been used in e.g. image/texture retrieval using the EMD [32].

The setup basically results in a total of eight individual complexity curves, 4 for each covariance structure. Despite the extra space required, it has been decided to include them all in order to illustrate the differences in a very visual and easily comparable way.

The models are trained five times for the same model size with random initialization of the K-means/EM algorithm. The BIC-optimal model (for each song of same size) is then selected as the one used for the similarity calculation. This BIC-approach was chosen based on the option of either reducing the number of setups, or neglecting the model variations returned

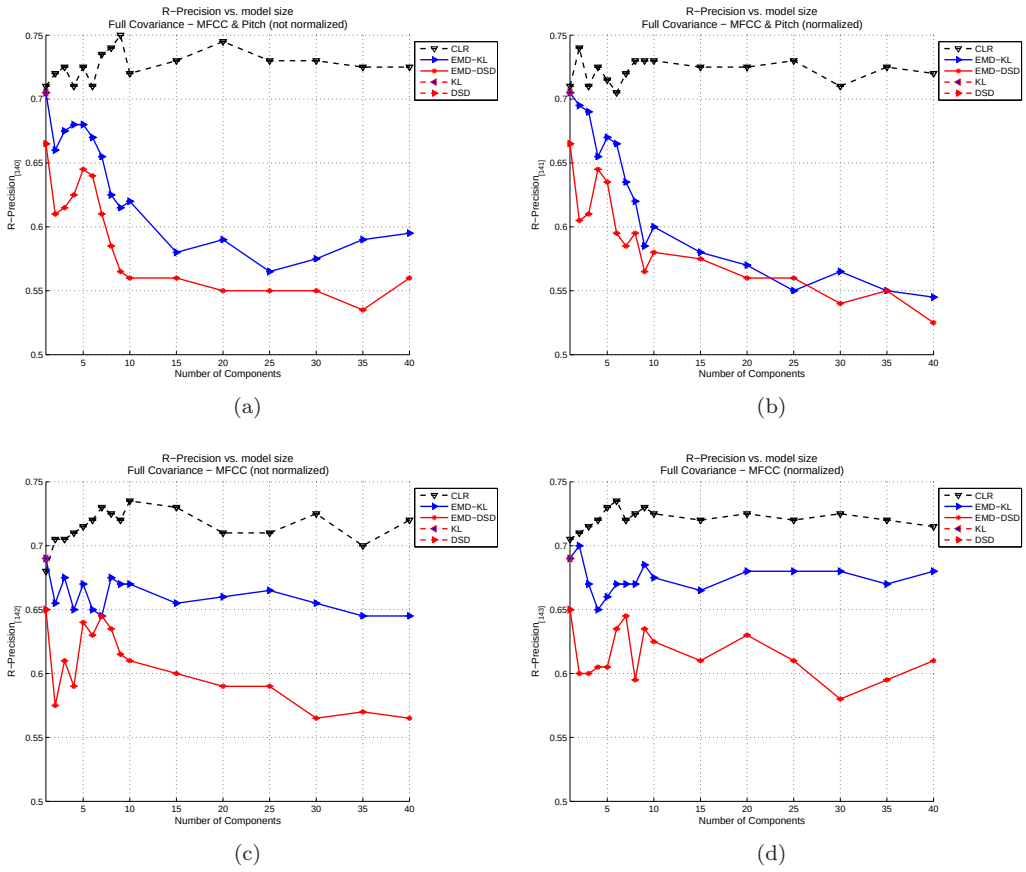


Figure 6.2: Song Retrieval: Full Covariance. The performance is very dependent on the individual model. It is noticeable that the CLR returns the best result using both a full covariance and the pitch features. Further does the KL and the DSD provide the overall best in this full covariance setting, suggesting that the potential of using a full covariance could be justified if a proper robust training is ensured. The EMD is very unreliable using the full covariance which is a general tendency through out the results. The EMD does only in a very few cases provide superior performance over the basic divergence, i.e. using one component. The performance of the simple KL is seen to be relatively high with 71% for the pitch case and 69% for the non-pitch case, indicating a small gain os using the pitch in this specific case. The BIC-optimal model being selected as the best of five, does not seem to improve the consistency across model size, and only the overall trend should be considered. There is no significant difference, considering the fluctuation, between the normalized and un-normalized data set.

by the EM algorithm. It was - due to the curiosity of the author - chosen to examine the mentioned model structures and settings leading to a single run of the measures utilizing only the BIC-optimal model for fixed sizes. Therefore must the comparison be made with this BIC selection in mind when considering the obtained quite fluctuating results. A different approach might have been preferable, however the trend of the results have been verified on a few individual setups, by avenging over random models, suggesting similar results in term of both the trend and absolute value obtained. An average over two runs, based on a test and a training set is then performed on the clip level, providing a smoother result, as described later.

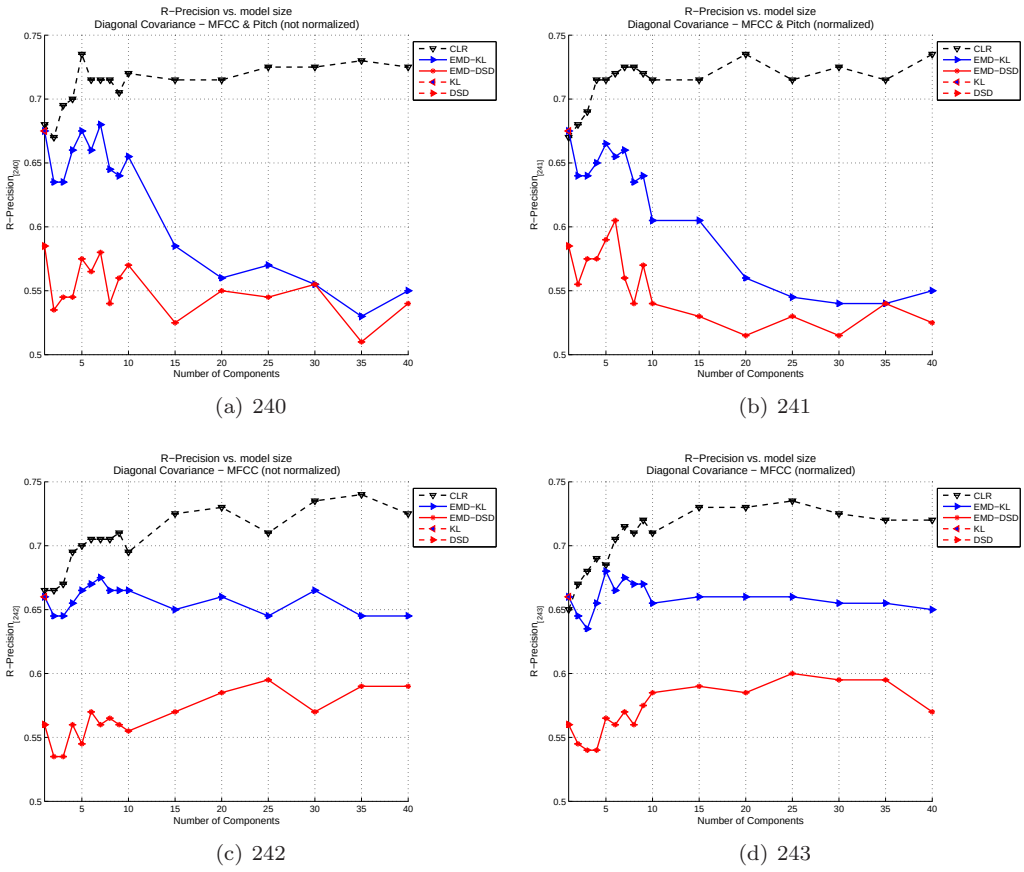


Figure 6.3: Song Retrieval: Diagonal Covariance. The consistency and robustness is seen to be better using the simpler covariance structure, and the EMD distance performs performs more as supposed to, i.e. better than the trivial ground distance, when not usign the apparently destructive pitch. No significant difference between the normalized and not-normalized data set.

Summary of song results

The results generally shows a very fluctuating response and the main findings for each setup is summarized in the figure captions. The main findings can be summarized as follows:

- The best individual result is obtained using a full covariance model. However this is based on a single run on the BIC optimal model (of five). This seems to suggest, that a full covariance is the better choice for the semi-discrete pitch feature, although the CLR using a diagonal covariance provides a best results very close to that of the full structure.
- The diagonal covariance structure provides more stable performance of the EMD compared to a full structure when not considering pitch hence providing the more robust models. The diagonal case shows equal performance using un-normalized versus normalized feature space.
- The Cross-Likelihood Ratio shows consistent, fairly robust results across all setups, around 73-75 % in retrieval rate.

- EMD is very dependent on the model applied, only working fully as intended using a diagonal covariance structure without pitch. The best obtained results using divergence based metrics is obtained using the basic ground distance, Kullback-Leibler and Divergence shape distance, with a full covariance, suggesting a very good relative result around 69-71%. This indicates some gain in including the pitch features if a proper, robust model can be constructed
- The use of the Divergence Shape distance does not provide any gain whatsoever over the standard Kullback-Leibler divergence, which has been seen in audio segmentation [19]. This is only considered as a minor result, and has not been investigated further.

In order to provide a BIC-optimal model across sizes (not variation in covariance structure), is the result using a diagonal covariance without pitch (figure 6.3(d)) singled out. This will only provide an indication of the model selection potential on the data set. The small experiment is quite extensive in terms of computation, since it - given the approach chosen - requires five models to be trained in the predefined interval from 1 to 40 components. The histogram of the returned model sizes is depicted in figure 6.4. The resulting R-precision, yields 0.7 for CLR, 0.67 for EMD and 0.62 for the EMD-DSD, which does not reach the maximum obtained using fixed model sizes.

The rather negative result should not be over-interpreted, and the evaluation should be performed on other setups. However, it does indicate that the BIC measure might not be optimal in the model selection for the similarity measures.

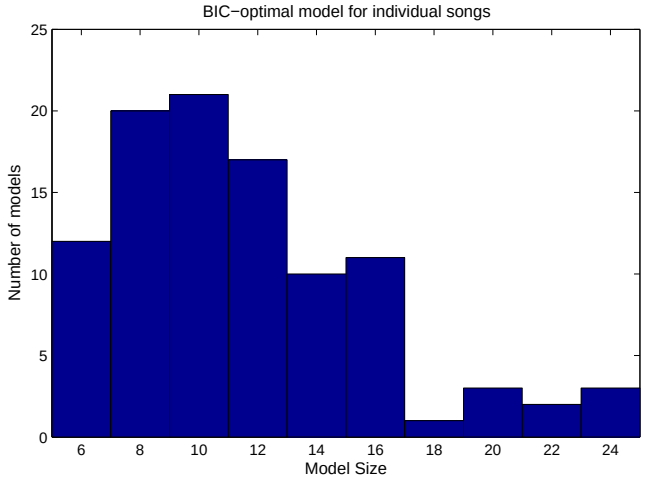


Figure 6.4: Histogram of the BIC-optimal models sizes for the diagonal case in a pure MFCC feature space. The majorty of the 100 models seems to provide the minimum BIC value at around 10 components

6.2.2 Clip Level Retrieval (Pop genre)

The traditional measures are evaluated on the clip level for the Pop genre, i.e. one model per clip, using the same setups as above (except a BIC-selection across model size). The setup was defined as

Relevance The relevant set in the R-precision measure is defined as being the artist, i.e. the measure must return a clip of one of two songs from the artist consisting of originally 20 clips, but due to a splitting (see later) this becomes 10 clips (5 for each song).

Structure Covariance structure: Full / Diagonal

Size Number of components in the model (fixed for all songs)

Features Two feature sets are evaluated, one using only the MFCCs and one including the pitch. Further more the normalization issues often considered in machine learning is again evaluated on the similarity measures.

The metric based similarity functions are evaluated using a T=5 point approximation for each formulation, i.e. Tipping, Rattray and Fisher/Kaski, and the analytical approximation of Tipping and Rattray. This parameter "tuning" is based on initial verification of both time consumption and precision, of which the first was prioritized higher, since the computation of a similarity matrix scales like n^2 with n being the number of clips. It is furthermore noted that the T=5 point approximations, did provide reasonable result when using it in clustering. Although the Phoneme set indicating some degradation in performance when comparing the 5-pint against the 10-point approximation.

The main objective is to verify, that the metrics are applicant in the music feature space, and to suggest a model structure (full/diagonal), and general motivation for further exploration using the metric approach. The metrics will furthermore be compared with each other; however models will be fitted using the supervised mixture model - which due to the restriction $\sum_{y=1}^{Y} p(y|k) = 1$, can not be fitted with model size below $K = Y$. This potentially favors the supervised metric - which are thereby defined as the most interesting metric in this setup.[2].

The clips, for each song, were divided into a training and a test set in order to provide an unbiased estimation of the generalization abilities of the metrics. The main motivation is based on the song retrieval results, which provided a rather fluctuating result and the fact that the metrics are based directly on the data set, i.e. the mean of one clip. However, no evaluation of the variation due to model uncertainty is provided and the BIC-selection of five trained models is still used. This is again done to save computational time of calculating the rather demanding similarity matrix between the 100 clips. Overall this should indicate that the estimation of the test error should not be over-interpreted. The relevant number of clips becomes - due to the splitting of the data set - five per song, as already mentioned. This means the labels/classes defined in the supervised training is the artist labels of which 10 exist for the individual genres.

Only the best/informative results for various covariance structures are illustrated for the metric and traditional measure, respectively. The left out results are included in appendix

[2]This approach was also taken in the clustering examples, but the size of the problem could in these cases be seen not to favor the supervised, except partly in the Phoneme data set

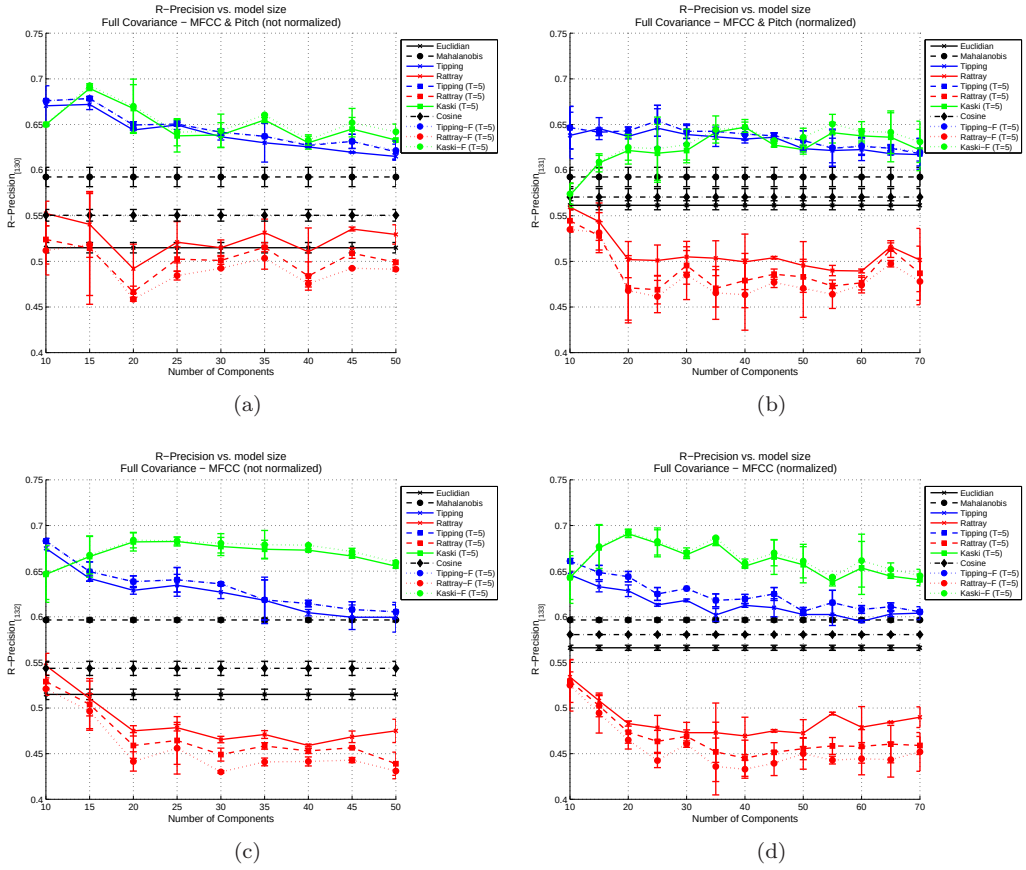


Figure 6.5: Clip Retrieval using Metrics: Full Covariance. The full covariance structure seems to work better for the metric case. Using the pitch, Kaski is better in the un-normalized space providing the best results overall, along with the normalized pure MFCC case. The performance of the Fisher/Kaski metric is quite good, benefiting from the conditional probability and performs better or equal to that of CLR in all cases. It is noticeable that the Tipping metric seems to obtain the same results as the Kaski metric in certain cases. It should also be noted that the Tipping metric may prefer smaller models than the used here as mentioned in the main text.

F, figure F.1 and F.2,respectively. The included results are for the local metric case a full covariance structure 6.5 and for the traditional a diagonal covariance structure, 6.6.

Summary of clip result: The results are, as mentioned, based on a split of the songs into 2x5 clips in order to provide some evaluation of the generalization error for the metrics based similarity measures. This approach ultimately results in a much more smoother response from the traditional similarity measures, and thus provides some idea of the performance in a general sense on this level.

The overall findings are:

- The supervised metric is generally the superior providing results on the good side of 65% (max 69%) for the full covariance models, while the diagonal does not seem to provide a suitable model for the T=5 approximation.

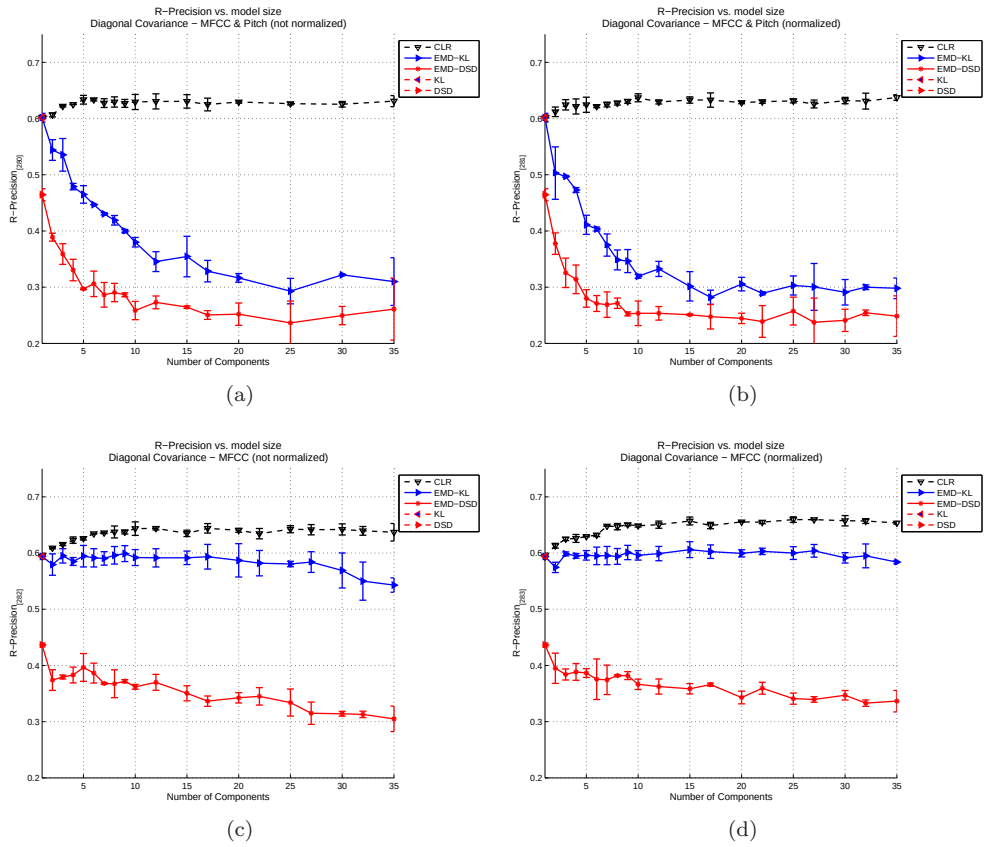


Figure 6.6: Clip Retrieval: Diagonal Covariance. Generally the diagonal covariance provides significant gain in the maximum obtained retrieval rate for CLR, but it does indicate - the main reason for including the structure investigation - that the EMD does perform slightly better in terms of its defined ground distance. However the absolute rate obtained is quite disappointing for the EMD distances since a smaller model in terms of components is seen to outperform the divergence based metrics when using a full covariance model. A further model simplification may be the way for improving the performance of the EMD, however the change must be quite large to reach the level of the full covariance using a single component.

- Rattray's metric performs quite poorly on the music set - which was also seen on the Phoneme set.

- The trends from the song retrieval is reflected in the traditional based similarity functions, in which the EMD has problems using a full covariance structure, but provides the better results. The CLR is again fairly consistent over all setups obtaining results in the range 63-65% with a full covariance structure being slightly better.

- The Fisher/Kaski and Tipping metric obtains significantly higher rater than the traditional measures including tractional geometric measures like the basic Euclidian, Cosine and Mahalanobis.

- Floyd generally does not aid the metrics, which is due to two main factors. One being the use of approximation which was shown to degrade the performance of Floyd using the approximations in the clustering examples. The other main reason lies in the

fact that we effective remove all the data points and the geodesic path can only be computed via the vector quantization representation of the clips.

- The inclusion of pitch using metrics is not consistent, but using the unnormalized space does indicate a small potential improvement (1%), but not generally significant. The traditional methods are generally unaffected by the normalization on the clip level.

Based on the fact, that the unsupervised metrics may provide better performance on smaller models, was the full covariance setup repeated using unsupervised training (the results are available in appendix F figure F.3. This test showed no improvement in the general level of Rattray's metric (best at one component), and the Tipping metric showed a trend in the test error towards models of size 8-10 or above, however, at no point did the unsupervised Tipping reach the level and general trend returned by the supervised training (maximum of Tipping metric is around 63-64%), indicating that the supervised training is beneficial also to unsupervised metrics, however the details should be exploited further.

6.3 Summary & Discussion

This chapter presented the experiments conducted on the custom data set. Two levels in the hierarchy was considered, namely the song level, which is comparable with other reported results on music similarity, e.g. [3, 2]. Several setups was considered, based on these previously reported results which mainly suggest simpler model structure, when using the EMD and the Cross-likelihood Ratio. The objective is to evaluate aspects model complexity, feature normalization and secondly the inclusion of an extra perceptual feature, i.e. the pitch.

It was found, using the custom data set, that the EMD is extremely sensitive to the model fitted, resulting in suboptimal, i.e. worse than the ground-distance, performance when considering all but a pure MFCC space and diagonal covariance structure. However, the performance in this situation did not reach that of the ground-distance - mainly the Kullback-Leibler (KL) divergence - using a full covariance matrix. A reletively good results was obtained for the KL divergence for the both the pitch and non-pitch cases - but best for the pitch case (71 %). This result is taken to imply that the inclusion of pitch may prove valuable if a suitable model can be fitted to the discrete like feature - or perhaps even more relevant; the raw pitch can potentially be applied in a pure discriminative model in which the discrete like nature is not critical setup (e.g. linear discriminant). The Divergence Shape Distance proved to be quite irrelevant in this similarity experiments.

Comparing with other similar experiments reveals a relatively good correspondence between this small setup and larger experiments [5], when considering the over all best. However, the performance obtained using the ground-distance in the full covariance setup is of course noticeable specially when considering the low cost of computing the KL-divergence, compared to the higher complexity of the EMD. This indicates, that the justification for using the computational heavier EMD over basic KL divergence, will depend on the models fitted and obviously not justified on this data set, when using a full covariance model. However, such a conclusion must be based on the individual data. Some guidance can of course be extracted based on the included results.

The CLR showed a quite high robustness in the song retrieval to both normalization, model structure, model size and feature set, mainly because it is based directly on the likelihood of the data given the models. It returned a quite high retrieval rates (73-75 %), compared with other experiments (65%, [3]). This, of course, is not generalizable due to the small data set considered. Furthermore, does the choice of a single run for each model size, based on the BIC-optimal model, not provide the most robust results for a final conclusion, but the overall trend is quite evident, though. A suggestion of using the BIC-optimal model for each across model size, proved to provide no gain, on the single example considered. Other selection criterions may improve the performance.

A major focus in this project is the application and properties of the local metrics described in chapter 4. The basic properties was illustrated through experiments in chapter 4, and the relatively good results - especially using the supervised Fisher/Kaski metric has motivated the use of these on the music data set. This is done though a direct comparison in feature space, providing a more explorative measure based on the defined auxiliary information (i.e. artist). The comparison was made using a crude vector quantization of the clips, combined with an unbiased estimation of the test error.

The results obtained in this low level test, showed the metric based similarity measures (Tipping and Kaski) to have superior performance over the traditional methods, given a full

covariance. The inclusion of pitch did not reveal any overall gain across setups - although this aspect has not been investigated further. However, a single result indicated that a good model may benefit from this feature given a optimal model, obtaining the best result of 69% R-precision.

The use of a full covariance structure is in contrast to the clustering results obtained in chapter 4, however, the objective is somewhat different in this point-to-point comparison, and the feature space is moreover different in the sense that the cepstral coefficients in the Phoneme set are not calculated using the setup described in chapter 2. The retrieval was based on a T=5 point approximation and a more accurate retrieval is most likely bound to yield even better results, which was demonstrated using the Phoneme set, were a T=10 point approximation was needed for good results on the more advanced model. Furthermore, is a better representation of the clip expected to pride a more informative exploration an possibly retrieval rate.

One important issue in regards to the music similarity discussion is the time consumption and scalability of the techniques, which was prioritized lower in the basic investigation of the metrics. General idea behind the metrics are of a global nature calling for a global density model describing all points and hence classes in feature space. Due to the relatively bad scalability of the EM algorithm, this very much limits the size of the problems, which can be analyzed with the metrics, at least without applying a more clever training. The metrics are in this thesis formulated in terms such a global model, and due to the estimation of the posterior component probabilities, the class probability (Fisher/Kaski case) and summation(s), will an increase in the model size contribute considerably to the computational load, independent of the numerical approximation applied. The metrics, however, scales linearly with the number of T-points (5 used here), but combined with the n^2-scaling in calculating a similarity matrix with n being the number of songs/clips, will the T-value have a significant influence on the overall time required. The T=5 point approximation was found reasonable for the purpose of illustrating the basic properties in music. In general is the pop genre problem considered to be on the limit of what is reasonable for a general retrieval task, when using the implemented EM algorithm and the metrics. A detailed complexity analysis including the influence the model complexity and numerical approximation is recommended for similar problems in order to evaluate the justification of applying such a learning metric approach - since it is definitely not insignificant as hereby noted - but more or less ignored in order to provide these basic results.[3]

A solution to the computational problems of both estimating the model and the similarity calculating, could be provided by a different estimation/model of the conditional distribution. It is, as previously mentioned, suggested by the original researchers due use a direct estimating of $p(y|\mathbf{x})$ using a gradient descent optimization, which will obviously be a natural next step from the initial results provided here. Such approach will probably show higher robust ness to the discrete like feature, compared with the supervised mixture model used in this initial examination.

All-in-all, does the standard measures perform relatively good, although with the EMD requiring a robust feature space and corresponding diagonal covariance model - and the CLR showing superior performance over the divergence based measures (EMD, KL, DSD) in the song retrieval. Initial results for the the local metrics, using only one approximation for the comparison, was found quite interesting for the Tipping and Fisher/Kaski metric,

[3]The computational time required in order to both fit a medium size model (K=20-30) and the subsequent similarity estimation, spans from approximately 8 to 15 hours on a Sun ∼1Ghz server using a Matlab implementation

based on a crude quantization of the clips. Superior performance was generally obtained using the supervised metric, which in essence is based on a indirect classification, and hence is expected to perform better in such a retrieval task.

Chapter 7

Summary & Conclusion

Summary

This thesis was aimed at providing some insight into the similarity measures currently used for music retrieval and exploratory datamining. This has involved the examination of the basic properties of music and relevant features for such a investigation. This lead to the choice of only using perceptually motivated features, namely the mel-frequency cepstral coefficients (MFCC), and the two estimated pitches, chosen as a supplement to the MFCC's for the similarity task. A custom data set was constructed, and described in chapter 2.

The traditional similarity measures are all based on density estimations, and for this purpose a well-known Gaussian Mixture Model was described in terms of formulation and training. A supervised density model was furthermore investigated and implemented, aimed at describing the joint probability between the primary data and the classes/labels. The supervised model is finally reformulated to describe the change in conditional probability of the labels given the data.

An alternative machine learning approach is then described based on the desire to create better clustering and other explorative techniques. The principle is based on Riemannian geometry and metrics applied to the feature space. Three metrics was descried, of which two was purely unsupervised metrics based on a locally weighted covariance formulation and the local changes log-likelihood, respectively. Finally, the perhaps more challenging choice of an supervise metric was described and derived in terms of the supervised mixture model, originally based on a reformulation of the Fisher Information Matrix.

The metrics were implemented and verified on a few simple data sets. The metrics and a few approximations were applied to basic clustering applications and three data sets were clustered, showing various properties of the metrics - overall suggesting the performance of a supervised clustering to be more robust and provide a better clustering in terms of purity, than the two unsupervised, which was also expected based on the formulation.

The focus was then shifted to the task of music similarity, originally motivating the examination of the learning metrics. The various similarity methods were evaluated on the custom data set using a variety of model structures and feature combinations, showing the sensitivity and perhaps unfortunate properties of the Earth Mover Distance compared with the simpler Kullback-Leibler divergence using a full covariance matrix. The results showed a maximum song retrieval rate of approximately 74-75% for a single model evaluation (CLR) using a full covariance model in an original unnormalized feature space including the pitch. The Cross-Likelihood Ratio showed relatively robust performance across setups.

The main results consisting of the clip level evaluation was then performed in the Pop genre, mainly to show the abilities of the learning metric, and especially the supervised metric. The performance trends of the traditional methods was repeated on this level, with a lower absolute level (maximum of 66% for the CLR). The potential of using the learning metrics in music mining and exploration was demonstrated and the Tipping metric, and especially the supervised Fisher/Kaski metric, showed superior performance over the traditional similarity measures (maximum of 69% for the Fisher/Kaski metric), indicating the potential of using a metric based approach.

Conclusion

The main contribution of this Master's Thesis lies in the challenging area of examining and applying the learning metric principle to music data. The conclusion, in this regard, is based on the initial results obtained through a similarity experiment, and hence only constitutes preliminary results using such metrics in music. However, the Tipping and especially the Fisher/Kaski metric is indeed concluded to generally provide superior retrieval results on the limited data set, justifying the use of the advanced methods - at least when not considering the computational issues. Due to the current computational load and numerical approximations, such a conclusion is difficult to transfer to music sets considerably larger than the one considered.

A final conclusion in regards to the performance of traditional measures and variations is primarily based on the more robust results obtained through averaging, and the robustness and performance of the Cross-Likelihood Ratio is concluded to provide the better option of the traditional measures on this data set. The main contribution of the thesis in this regard, lies in the examination of the model structure for the widely applied measures, and it must be concluded that the Earth Mover Distance is extremely sensitive to the covariance model used on the custom data set. On the custom dataset can the EMD only be justified if a full covariance model is not obtainable, since the performance of the basic Kullback-Leibler using a full covariance model. However, this will again depend on the data set and hence feature space considered.

Overall, this thesis has contributed with insight into a new metric based, explorative method for use in audio applications and a comparison of this technique with existing methods, definitely showed the potential of a more data driven approach. However, certain computational issues, in term of numerical approximation and computation resources, needs to be handled in order to a provide a generally feasible method for music similarity.

Further work

The main contribution of this thesis is in the metric learning area, and the reported results reflects the limited time frame of this project. Therefore is a number of suggestions for further research provided, based on the problems and limitations identified - especially in relation to the supervised Fisher/Kaski metric.

More Music Exploration The task of data exploration, was in this thesis limited to the relatively simple task of retrieval (and a few other examples), which is essence only indicates the true exploration properties of the learning metric principle. Further investigation into the explorative nature of the metric for music is an obvious next step from the basic investigation performed in this thesis.

A visualization of the results, was originally planed to appear in this text, which based on a Sammon mapping visualized the clips based on the calculated features. However, this explorative visualization did not make the final version, due to time constraints - but is an obvious next step.
Other potential experiments include a deeper analysis of e.g. the pop genre, than performed here. Using the local metric an obvious option would be to analysis the individual clips in terms of feature separating them from others - possibly providing a deeper in-sight into both the feature space and the music itself.

Projection with learning metrics and other methods In this thesis the supervised learning metric was limited to the explicit estimation of the metric, given by $\mathbf{J(x)}$. However, as described in e.g. [24], is an implicit exploration possibly by formulating a projection, which indirectly obtains the same result as the explicit estimation. Such an approach might prove valuable in the analysis of music, but requires a further insight into the nature of learning metrics.

Projections are in general a interesting area, and as mentioned previously, does other linear and non-linear projection methods exist with relevance to this area. E.g. neural networks, principle component analysis, partially least squares and canonical correlation analysis.

Computational issues The experiments and conclusion above was based on the assumption that the computational load of the similarity estimation and dataming techniques does not matter. However as discussed in chapter 6, is the calculation of the metric based distances orders of magnitudes larger than e.g. a simple Euclidian distance, effectively limiting the type of datamining application in which the learning metric principles can be applied. The number of experiments in this thesis was limited because of this, and the amount of data present in the pop genre is on the limit what is reasonable given the current implementation[1]. This obviously calls for optimized implementations, and it is probably worth considering an analytical approximation of the Fisher/Kaski metric, which has not investigated further in this thesis.

Traditional Similarity Methods The number of experiments using the traditional methods was in this thesis limited to the custom data set used. However, based on the results obtained here - mainly the fact that the EMD does not provide a generally robust measure - calls for a large scale evaluation with the same setup as described in chapter 6, if the deeper implications are to be evaluated.

[1] The code is implemented in Matlab - not providing the best performance when inner loops are required, as with the metrics describe in this thesis.

General Machine Learning Using Learning Metric The application of the learning metric principle, limited to the Fisher/Kaski metric, has been reported in e.g. [24, 14] for e.g. bankruptcy analysis and text clustering. However, due the interesting nature of the metrics and the preserved topology, it is believed that such a method can provide insight into many other problems - of which the music provides an very relevant example.

Bibliography

[1] Shun-Ichi. Amari and H. Nagaoka. *Methods of Information Geometry*. American Mathematical Society, 2001. 4.5.1, 4.5.1

[2] Jean-Julien Aucouturier, François Pachet, and Peter Hanappe. From sound sampling to song sampling. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, 2004. 2.3, 5, 6.3

[3] J.J Aucouturier and F. Pachet. Representing musical genre: A state of the art. *Journal of New Music Research*, 32(1), 2003. 2.2.3, 2.3, 5, 5.2, 6.3

[4] A. Berenzweig, D.P.W. Ellis, and S. Lawrence. Anchor space for classification and similarity measurement of music. In *Proceedings. 2003 International Conference on Multimedia and Expo (ICME 2003)*, volume 1, pages 29–32. IEEE, 2003. 4.7, 5

[5] Adam Berenzweig, Beth Logan, Daniel P.W. Ellis, and Brian Whitman. A large-scale evaluation of acoustic and subjective music-similarity measures. *Computer Music Journal*, 28(2):63–76, 2004. 2.2.3, 2.3, 5, 5.1.2, 5.2, 6.3

[6] C. M. Bishop. *Neural networks for pattern recognition*. Oxford Clarendon Press, 2nd edition, 1997. 4.3, 4.5.4, 4.7

[7] TREC (Text REtrieval Conference). http://trec.nist.gov/, 2006. 6.1.1

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the *em* algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. 4.2.1, 4.2.1

[9] Diana Deutsch, editor. *The Psychology Of Music*. Academic Press, 2nd edition, 1999. 2.1, 2.1.1, 2.2, 2.1.1, 2.1.2, 2.1.3, 2.2.2

[10] A. Gray. *Modern differential geometry of curves and surfaces)*. Boca Raton,Fla. CRC Press, 1993. 4.5.1

[11] J.M. Grey. Multidimensional perceptual scaling of musical timbres. *Journal of the Acoustical Society of America*, 61(5):1270–1277, 1977. 2.1.3, 2.2.2, 2.2.2

[12] A. S. Have. *Datamining on distributed medical databases*. PhD thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, 2003. Vejleder: Lars Kai Hansen. C

[13] David Howard and Jamie Angus. *Acoustics and Psychoacoustics (Music Technology Series)*. Focal Press, 1996. 2.1.1, 2.1.3

[14] S. Kaski, J. Sinkkonen, and J. Peltonen. Bankruptcy analysis with self-organizing maps in learning metrics. *Neural Networks, IEEE Transactions on*, 12(4):936 –947, 2001. 4.5.4, 4.5.4, 4.8, 7

[15] Samuel Kaski and Janne Sinkkonen. Metrics that learn relevance. In *Proceedings of IJCNN-2000, International Joint Conference on Neural Networks*, volume V, pages 547–552. IEEE Service Center, Piscataway, NJ, 2000. 4.5.4

[16] A.P. Klapuri. A perceptually motivated multiple-f0 estimation method. *Applications of Signal Processing to Audio and Acoustics, 2005. IEEE Workshop on*, pages 291–294, 2005. 2.2.2, 2.2.2, 2.4, 2.2.2, 2.3, 3, 4.5.5

[17] J. Larsen, A. S. Have, and L. K. Hansen. Probabilistic hierarchical clustering with labeled and unlabeled data. *International Journal of Knowledge-Based Intelligent Engineering Systems*, 6(1):56–62, 2002. 4.3, 4.3

[18] B. Logan and A. Salomon. A music similarity function based on signal analysis. pages 745–748. IEEE, 2001. 5.1.2

[19] Lie Lu and Hong-Jiang Zhang. Speaker change detection and tracking in real-time news broadcasting analysis. *Proceedings of the ACM International Multimedia Conference and Exhibition*, pages 602–610, 2002. 5, 5.1.1, 6.2.1

[20] Anders Meng, Peter Ahrendt, and Jan Larsen. Improving music genre classification by short-time feature integration. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 497–500, 2005. 2.2.4

[21] D.J. Miller and H.S. Uyar. A mixture of experts classifier with learning based on both labelled and unlabelled data. *Advances in Neural Information Processing Systems 9. Proceedings of the 1996 Conference*, pages 571–7, 1997. 4.3, 4.3

[22] Morgan Nelson and Ben Gold. *Speech and audio signal processing: processing and perception of speech and music*. Wiley, 1999. 2.1.1, 2.1.1, 2.2.2, 2.6

[23] Elias Pampalk, Simon Dixon, and Gerhard Widmer. Exploring music collections by browsing different views. *Computer Music Journal*, 28(2):49–62, 2004. 5

[24] Jaakko Peltonen. *Data Exploration with Learning Metrics (PhD Thesis)*. Thesis/dissertation, November 2004. 4.7, 7, A

[25] Jaakko Peltonen, Arto Klami, and Samuel Kaski. Improved learning of Riemannian metrics for exploratory analysis. *Neural Networks*, 17:1087–1100, 2004. Invited paper. 4.5.4, 4.5.4, 4.5.5

[26] M. Rattray. A model-based distance for clustering. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, 2000 (IJCNN 2000)*, volume 4, pages 13–16. IEEE Comput. Soc, 2000. 4.5.3, 4.5.3, 4.5.3, 4.8

[27] Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000. 5.1.2, 5.1.2

[28] Malcolm Slaney. Technical report: Auditory toolbox - version 2, 1998. 2.2.3, 1

[29] M. E. Tipping. Deriving cluster analytic distance functions from gaussian mixture models. In *Ninth International Conference on (Conf. Publ. No. 470) Artificial Neural Networks (ICANN)*, volume 2, pages 815–820. IEEE, 1999. 4.5.2, 4.5.2

[30] Jacob A. Wegelin. A survey of partial least squares (pls), with emphasis on the two-block case. Technical Report (No. 371),Dept. of Statistics, University of Washington, U.S.A, 2000. 4.7

[31] T. J. Willmore. *Riemannian geometry*. Oxford Clarendon Press (repr. 1996), 1993. 4.5.1

[32] Chan Wu and Huang. Texture classification based on finite gaussian mixture model. In *IEEE International workshop on Texture Analysis and Synthesis (Texture2003)*. IEEE, 2003. 4.4, 5.1.2, 6.2.1

Appendix A

Relation between Kullback-Leibler divergence and Fisher"s Information Matrix

This proofs follows one presented in [24].

Consider to close by distribution p and q. The Kullback-Leibler divergence can then be approximated in term of its Taylor expansion in regards to $e_i = p_i - q_i$. Doing so around zero can be expressed by [24]

$$D_{KL}(p,q) = \sum_y \frac{(p_i - q_i)^2}{2p_i} + \mathcal{O}\left(\max_i |p_i - q_i|^3\right) \tag{A.1}$$

If considering the conditional distributions, p becomes $p = p(y|\mathbf{x})$ and q becomes $q = p(y|\mathbf{x}')$, where $\mathbf{x}' = \mathbf{x} + d\mathbf{x}$. If applying yet another Taylor expansion for the difference between p and q, $p(y|x') - p(y|x)$ becomes

$$p(y|\mathbf{x}') - p(y|\mathbf{x}) = (\mathbf{x}' - \mathbf{x}) \frac{\partial}{\partial \mathbf{x}} p(y|\mathbf{x}) + \mathcal{O}\left(\|\mathbf{x}' - \mathbf{x}\|^2\right)$$

Combining the two Taylor expansions, yields

$$\begin{aligned} D_{KL}(p(y|\mathbf{x}), p(y|\mathbf{x}')) &= \sum_y \frac{(\mathbf{x}' - \mathbf{x})^T \frac{\partial}{\partial \mathbf{x}} p(y|\mathbf{x}) \left(\frac{\partial}{\partial \mathbf{x}} p(y|\mathbf{x})\right)^T (\mathbf{x}' - \mathbf{x})}{2p(y|\mathbf{x})} \\ &\quad + \sum_y \frac{2\mathcal{O}\left(\|\mathbf{x}' - \mathbf{x}\|^2\right) (\mathbf{x}' - \mathbf{x})^T \left(\frac{\partial}{\partial \mathbf{x}} p(y|\mathbf{x}) + \mathcal{O}\left(\|\mathbf{x}' - \mathbf{x}\|^2\right)\right)^2}{2p(y|\mathbf{x})} \\ &\quad + \mathcal{O}\left(\max_i |p(y|\mathbf{x}) - p(y|\mathbf{x}')|^3\right) \end{aligned}$$

Where the first term is a quadratic form given by $\frac{1}{2}d\mathbf{x}^T\mathbf{J}(\mathbf{x})d\mathbf{x}$ Considering The second term, the gradient of $p(y|x)$ is considered constant around the expansion and the term becomes $\mathcal{O}\left(\|\mathbf{x}'-\mathbf{x}\|^3\right)$. The third terms is also $\mathcal{O}\left(\|\mathbf{x}'-\mathbf{x}\|^3\right)$ since $p(y|x')-p(c|x)$ contributes with each $\mathcal{O}\left(\|\mathbf{x}'-\mathbf{x}\|\right)$ for each y. I.e. the expression becomes

$$D_{KL}\left(p\left(y|\mathbf{x}\right),p\left(y|\mathbf{x}'\right)\right)=\frac{1}{2}d\mathbf{x}^T\mathbf{J}(\mathbf{x})d\mathbf{x}+\mathcal{O}\left(\|d\mathbf{x}\|^3\right)$$

Which gives the results in 4.77.

Appendix B

Derivation of the Fisher/Kaski metric

B.1 Supervised Riemannian Metric

This section contains the full proof of the supervised metric based on the supervised gaussian mixture model.

The metric is defined from the Fisher Information matrix in 4.78 and repeated here for convenience

$$J(\mathbf{x}) = E_{p(y|\mathbf{x})}\left\{\frac{\partial \log p\left(y|\mathbf{x}\right)}{\partial \mathbf{x}}\left(\frac{\partial \log\left(p\left(y|\mathbf{x}\right)\right)}{\partial \mathbf{x}}\right)^T\right\} \tag{B.1}$$

$$= \int_{y\in Y} p(y|\mathbf{x})\frac{\partial \log p\left(y|\mathbf{x}\right)}{\partial \mathbf{x}}\left(\frac{\partial \log\left(p\left(y|\mathbf{x}\right)\right)}{\partial \mathbf{x}}\right)^T dy \tag{B.2}$$

Where Y is the set of labels $Y = \{y_1, y_2 ... y_n\}$, and the integral effectively becomes a sum over the discrete values of y.

The conditional probability given the supervised mixture model modelling the joint proba-

bility $p(\mathbf{x}, y)$, can be written as

$$\begin{aligned} p\left(y|\mathbf{x}\right) &= \frac{p\left(y,\mathbf{x}\right)}{p\left(\mathbf{x}\right)} && \text{(B.3)} \\ &= \sum_{k=1}^{K} p\left(y|k\right) p\left(k|\mathbf{x}\right) && \text{(B.4)} \\ &= \sum_{k=1}^{K} p\left(y|k\right) \frac{p\left(\mathbf{x}|k\right) P\left(k\right)}{p\left(\mathbf{x}\right)} && \text{(B.5)} \\ &= \sum_{k=1}^{K} p\left(y|k\right) \frac{p\left(\mathbf{x}|k\right) P\left(k\right)}{\sum_{k'=1}^{K} p\left(\mathbf{x}|k'\right) P\left(k'\right)} && \text{(B.6)} \\ &= \frac{\sum\limits_{k=1}^{K} p\left(y|k\right) p\left(\mathbf{x}|k\right) P\left(k\right)}{\sum\limits_{k=1}^{K} p\left(\mathbf{x}|k\right) P\left(k\right)} && \text{(B.7)} \end{aligned}$$

Using the basic chain rule in regards to the partial derivative

$$\frac{\partial \log\left(p(y|\mathbf{x})\right)}{\partial \mathbf{x}} = \frac{1}{p(y|\mathbf{x})} \frac{\partial p(y|\mathbf{x})}{\partial \mathbf{x}} \qquad \text{(B.8)}$$

$$\text{(B.9)}$$

Using basic rule for differentiating factions and plugging in $p\left(y|x\right)$ from B.7

$$\begin{aligned} \frac{\partial p(y|\mathbf{x})}{\partial \mathbf{x}} &= \frac{\frac{\partial \sum_{k=1}^{K} p(y|k) p(\mathbf{x}|k) P(k)}{\partial \mathbf{x}} \sum_{k=1}^{K} p\left(\mathbf{x}|\theta_k\right) P\left(k\right)}{\left(\sum_{k=1}^{K} p\left(\mathbf{x}|\theta_k\right) P\left(k\right)\right)^2} \\ &\quad - \frac{\frac{\partial \sum_{k=1}^{K} p(\mathbf{x}|k) P(k)}{\partial \mathbf{x}} \sum_{k=1}^{K} p\left(y|k\right) p\left(\mathbf{x}|k\right) P\left(k\right)}{\left(\sum_{k=1}^{K} p\left(\mathbf{x}|k\right) P\left(k\right)\right)^2} \end{aligned} \qquad \text{(B.10)}$$

First term, numerator: Differentiating and sums are interchangeable, i.e

$$\frac{\partial \sum_{k=1}^{K} p\left(y|k\right) p\left(\mathbf{x}|k\right) P\left(k\right)}{\partial \mathbf{x}} = \sum\nolimits_{k=1}^{K} \left\{ P\left(k\right) p\left(y|k\right) \frac{\partial p\left(\mathbf{x}|k\right)}{\partial \mathbf{x}} \right\} \qquad \text{(B.11)}$$

Second term, numerator:

$$\frac{\partial \sum_{k=1}^{K} p\left(\mathbf{x}|k\right) P\left(k\right)}{\partial \mathbf{x}} = \sum\nolimits_{k=1}^{K} \left\{ P(k) \frac{\partial p\left(\mathbf{x}|k\right)}{\partial \mathbf{x}} \right\} \qquad \text{(B.12)}$$

Substituting B.11 and B.12 into B.10 yields (after a small rearrangement of factors)

$$\begin{aligned}\frac{\partial p(y|\mathbf{x})}{\partial \mathbf{x}} &= \frac{\sum_{k=1}^{K} p(\mathbf{x}|k) P(k) \sum_{k=1}^{K} P(k) p(y|k) \frac{\partial p(\mathbf{x}|\theta_k)}{\partial \mathbf{x}}}{\left(\sum_{k=1}^{K} p(\mathbf{x}|k) P(k)\right)^2} \\ &\quad - \frac{\sum_{k=1}^{K} p(y|k)\, p(\mathbf{x}|k) P(k) \sum_{k=1}^{K} P(k) \frac{\partial p(\mathbf{x}|\theta_k)}{\partial \mathbf{x}}}{\left(\sum_{k=1}^{K} p(\mathbf{x}|k) P(k)\right)^2} \qquad \text{(B.13)} \\ &= \frac{\sum_{k=1}^{K} \left\{ P(k) p(y|k) \frac{\partial p(\mathbf{x}|k)}{\partial \mathbf{x}} \right\} - p(y|x) \sum_{k=1}^{K} \left\{ P(k) \frac{\partial p(\mathbf{x}|k)}{\partial \mathbf{x}} \right\}}{\sum_{k=1}^{K} p(\mathbf{x}|k) P(k)} \qquad \text{(B.14)}\end{aligned}$$

Since $p(y|\mathbf{x})$ is constant in regards to k

$$\begin{aligned}\frac{\partial p(y|\mathbf{x})}{\partial \mathbf{x}} &= \frac{\sum_{k=1}^{K} P(k) \left[p(y|k) - p(y|\mathbf{x}) \right] \frac{\partial p(\mathbf{x}|k)}{\partial \mathbf{x}}}{\sum_{k=1}^{K} p(\mathbf{x}|k) P(k)} \\ &= \sum_{k=1}^{K} P(k) \frac{\left[p(y|k) - p(y|\mathbf{x}) \right]}{\sum_{k'=1}^{K} p(\mathbf{x}|k') P(k')} \frac{\partial p(\mathbf{x}|k)}{\partial \mathbf{x}} \qquad \text{(B.15)}\end{aligned}$$

We get the following temporary expression for the partial derivative

$$\frac{\partial \log p(y|\mathbf{x})}{\partial \mathbf{x}} = \sum\nolimits_{k=1}^{K} \left[\frac{P(k) p(y|k) - P(k) p(y|\mathbf{x})}{p(y|x) \sum_{k'=1}^{K} p(\mathbf{x}|k') P(k')} \right] \frac{\partial p(\mathbf{x}|k)}{\partial \mathbf{x}} \qquad \text{(B.16)}$$

The factor in the brackets can be simplified by noting that the derivative of the pdf is

$$\frac{\partial p(\mathbf{x}|k)}{\partial \mathbf{x}} = -\mathbf{C}_k^{-1} (\mathbf{x} - \mu_k) p(\mathbf{x}|k) \qquad \text{(B.17)}$$

and

$$p(y|x) = \frac{\sum_{k=1}^{K} p(y|k) p(\mathbf{x}|k) P(k)}{p(\mathbf{x})} \qquad \text{(B.18)}$$

Using the above and multiplying the $p(\mathbf{x}|k)$ factor from the pdf derivative in B.17 into the brackets in B.16 we get to the simplification

$$\begin{aligned}p(\mathbf{x}|k) \frac{P(k) p(y|k) - P(k) p(y|\mathbf{x})}{p(y|x) p(\mathbf{x})} &= \frac{p(\mathbf{x}) P(k) p(y|k) p(\mathbf{x}|k)}{p(\mathbf{x}) \sum_{k=1}^{K} p(y|k) p(\mathbf{x}|k) P(k)} - \frac{P(k) p(\mathbf{x}|k)}{p(\mathbf{x})} \\ &= \frac{P(k) p(y|k) p(\mathbf{x}|k)}{\sum_{k=1}^{K} p(y|k) p(\mathbf{x}|k) P(k)} - \frac{P(k) p(\mathbf{x}|k)}{\sum_{k=1}^{K} p(\mathbf{x}|k) P(k)} \\ &= p(k|\mathbf{x}, y) - p(k|\mathbf{x}) \qquad \text{(B.19)}\end{aligned}$$

The full derivative of the conditional probability can be now written as

$$\frac{\partial \log p(y|\mathbf{x})}{\partial \mathbf{x}} = \sum_{k=1}^{K} - \left[p(k|\mathbf{x}, y) - p(k|\mathbf{x}) \right] \mathbf{C}_k^{-1} (\mathbf{x} - \mu_k) \qquad \text{(B.20)}$$

and

$$\frac{\partial \log p\left(y|\mathbf{x}\right)}{\partial \mathbf{x}}\left(\frac{\partial \log p\left(y|\mathbf{x}\right)}{\partial \mathbf{x}}\right)^{T}=\sum_{\mathrm{k}=1}^{K}\sum_{l=1}^{K}\left[p\left(k|\mathbf{x},y\right)-p\left(k|\mathbf{x}\right)\right]\left[p\left(l|\mathbf{x},y\right)-p\left(l|\mathbf{x}\right)\right]\mathbf{Q}_{kl} \quad \text{(B.21)}$$

with $\mathbf{Q}_{kl}=\mathbf{C}_{k}^{-1}\left(\mathbf{x}-\mu_{k}\right)\left(\mathbf{x}-\mu_{l}\right)^{T}\mathbf{C}_{l}^{-1}$. Now the expectation i.r.t $p\left(y|\mathbf{x}\right)$ is taken i.e.

$$\mathbf{J}\left(\mathbf{x}\right)=E_{p(y|\mathbf{x})}\left\{\sum_{\mathrm{k}=1}^{K}\sum_{l=1}^{K}\left[p\left(k|\mathbf{x},y\right)-p\left(k|\mathbf{x}\right)\right]\left[p\left(l|\mathbf{x},y\right)-p\left(l|\mathbf{x}\right)\right]\mathbf{Q}_{kl}\right\} \quad \text{(B.22)}$$

or when dealing with discrete values of y, as in this case

$$\mathbf{J}\left(\mathbf{x}\right)=\sum_{n}^{Y}p\left(y_{n}|\mathbf{x}\right)\sum_{\mathrm{k}=1}^{K}\sum_{l=1}^{K}\left[p\left(k|\mathbf{x},y_{n}\right)-p\left(k|\mathbf{x}\right)\right]\left[p\left(l|\mathbf{x},y_{n}\right)-p\left(l|\mathbf{x}\right)\right]\mathbf{Q}_{kl} \quad \text{(B.23)}$$

Appendix C

Kullback-Leibler Divergence

This appendix contains a derivation of the analytical Kullback-Leibler divergence. The proof follows one in [12]

$$D_{KL}\left(p\left(\mathbf{x}|k\right)||p\left(\mathbf{x}|l\right)\right) = \int p\left(\mathbf{x}|k\right) \ln \frac{p(\mathbf{x}|k))}{p(\mathbf{x}|l)} d\mathbf{x}$$
$$D_J = \frac{1}{2} \int p\left(\mathbf{x}|k\right) \ln \frac{p(\mathbf{x}|k)}{p(\mathbf{x}|l)} d\mathbf{x} + \frac{1}{2} \int p\left(\mathbf{x}|l\right) \ln \frac{p(\mathbf{x}|l))}{p(\mathbf{x}|k)} d\mathbf{x}$$

The Gaussian pdf's are given by

$$p\left(\mathbf{x}|k\right) = (2\pi)^{-\frac{M}{2}} \left|\mathbf{C}_k\right|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\left(\mathbf{x}-\mu_k\right)^T \mathbf{C}_k \left(\mathbf{x}-\mu_k\right)\right)$$
$$p\left(\mathbf{x}|l\right) = (2\pi)^{-\frac{M}{2}} \left|\mathbf{C}_l\right|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\left(\mathbf{x}-\mu_l\right)^T \mathbf{C}_k \left(\mathbf{x}-\mu_l\right)\right)$$

Considering the faction given in the KL expression yields, by inserting the pdf's

$$\begin{aligned} \ln \frac{p\left(\mathbf{x}|l\right))}{p\left(\mathbf{x}|k\right)} &= \ln \frac{(2\pi)^{-\frac{M}{2}} \left|\mathbf{C}_k\right|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\left(\mathbf{x}-\mu_k\right)^T \mathbf{C}_k^{-1} \left(\mathbf{x}-\mu_k\right)\right)}{(2\pi)^{-\frac{M}{2}} \left|\mathbf{C}_l\right|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\left(\mathbf{x}-\mu_l\right)^T \mathbf{C}_l^{-1} \left(\mathbf{x}-\mu_l\right)\right)} \\ &= \frac{1}{2}\ln\left(\left|\mathbf{C}_l\right|\right) - \frac{1}{2}\ln\left(\left|\mathbf{C}_k\right|\right) - \frac{1}{2}\left(\mathbf{x}-\mu_k\right)^T \mathbf{C}_k^{-1} \left(\mathbf{x}-\mu_k\right) - \frac{1}{2}\left(\mathbf{x}-\mu_l\right)^T \mathbf{C}_l^{-1} \left(\mathbf{x}-\mu_l\right) \end{aligned}$$

Since

$$\left(\mathbf{x}-\mu\right)^T \mathbf{C}^{-1} \left(\mathbf{x}-\mu\right) = tr\left(\mathbf{C}^{-1} \left(\mathbf{x}-\mu\right)\left(\mathbf{x}-\mu\right)^T\right) \tag{C.1}$$

we obtain

$$\int p\left(\mathbf{x}|k\right) \ln \frac{p(\mathbf{x}|k)}{p(\mathbf{x}|l)} d\mathbf{x} = \frac{1}{2} \int p\left(\mathbf{x}|k\right) \ln\left|\mathbf{C}_l\right| d\mathbf{x}$$
$$-\frac{1}{2} \int p\left(\mathbf{x}|l\right) \ln\left|\mathbf{C}_k\right| dx$$
$$-\frac{1}{2} \int p\left(\mathbf{x}|k\right) \left(\mathbf{x}-\mu_k\right)^T \mathbf{C}_k^{-1} \left(\mathbf{x}-\mu_k\right) d\mathbf{x}$$
$$+\frac{1}{2} \int p\left(\mathbf{x}|k\right) \left(\mathbf{x}-\mu_l\right)^T \mathbf{C}_l^{-1} \left(\mathbf{x}-\mu_l\right) d\mathbf{x}$$

Furthermore exploiting the linearity of the trace operator the integral becomes

$$\begin{aligned} -\frac{1}{2}\int p\left(\mathbf{x}|k\right)\left(\mathbf{x}-\mu_k\right)^T\mathbf{C}_k^{-1}\left(\mathbf{x}-\mu_k\right)d\mathbf{x} &= -\int\frac{1}{2}p\left(\mathbf{x}|k\right)tr\left(\mathbf{C}_k^{-1}\left(\mathbf{x}-\mu_k\right)\left(\mathbf{x}-\mu_k\right)^T\right)d\mathbf{x} \\ &= -\frac{1}{2}tr\left(\mathbf{C}_k^{-1}\int p\left(\mathbf{x}|k\right)\left(\mathbf{x}-\mu_k\right)\left(\mathbf{x}-\mu_k\right)^T d\mathbf{x}\right) \\ &= -\frac{1}{2}tr\left(\mathbf{C}_k^{-1}\mathbf{C}_k\right) \\ &= -\frac{M}{2} \end{aligned}$$

$$\frac{1}{2}\int p\left(\mathbf{x}|k\right)\left(\mathbf{x}-\mu_l\right)^T\mathbf{C}_l^{-1}\left(\mathbf{x}-\mu_l\right)d\mathbf{x} = \frac{1}{2}\int p\left(\mathbf{x}|k\right)tr\left(\mathbf{C}_l^{-1}\left(\mathbf{x}-\mu_l\right)\left(\mathbf{x}-\mu_l\right)^T\right)d\mathbf{x} \quad \text{(C.2)}$$

$$= \frac{1}{2}\int p\left(\mathbf{x}|k\right)tr\left(\mathbf{C}_l^{-1}\left(\mathbf{x}-\mu_l\right)\left(\mathbf{x}-\mu_l\right)^T\right)d\mathbf{x} \quad \text{(C.3)}$$

$$= \frac{1}{2}tr\left(\mathbf{C}_l^{-1}\int p\left(\mathbf{x}|k\right)\left(\mathbf{x}-\mu_l\right)\left(\mathbf{x}-\mu_l\right)^T d\mathbf{x}\right) \quad \text{(C.4)}$$

$$= \frac{1}{2}tr\left(\mathbf{C}_l^{-1}\int p\left(\mathbf{x}|k\right)\left(\left(\mathbf{x}-\mu_k\right)\left(\mu_k-\mu_l\right)\right)\left(\left(\mathbf{x}-\mu_k\right)\left(\mu_k-\mu_l\right)\right)^T d\mathbf{x}\right) \quad \text{(C.5)}$$

$$= \frac{1}{2}tr\Big(\mathbf{C}_l^{-1}\int p\left(\mathbf{x}|k\right)\Big(\left(\mathbf{x}-\mu_k\right)\left(\mathbf{x}-\mu_k\right)^T \quad \text{(C.6)}$$

$$+\left(\mathbf{x}-\mu_k\right)\left(\mu_k-\mu_l\right)^T+\left(\mu_k-\mu_l\right)\left(\mathbf{x}-\mu_k\right)^T+\left(\mu_k-\mu_l\right)\left(\mu_k-\mu_l\right)^T\Big)d\mathbf{x}\Big) \quad \text{(C.7)}$$

$$= \frac{1}{2}tr\left(\mathbf{C}_l^{-1}\left(\mathbf{C}_k+0+0+\left(\mu_k-\mu_l\right)\left(\mu_k-\mu_l\right)^T\right)\right) \quad \text{(C.8)}$$

$$= \frac{1}{2}tr\left(\mathbf{C}_l^{-1}\mathbf{C}_k\right)+\frac{1}{2}\left(\mu_k-\mu_l\right)^T\mathbf{C}_l^{-1}\left(\mu_k-\mu_l\right) \quad \text{(C.9)}$$

Now the KL divergence can be expressed by

$$\int p\left(\mathbf{x}|k\right)\ln\frac{p\left(\mathbf{x}|k\right)}{p\left(\mathbf{x}|l\right)}d\mathbf{x} = \frac{1}{2}\ln\left(|\mathbf{C}_l|\right)-\frac{1}{2}\ln\left(|\mathbf{C}_k|\right)-\frac{1}{2}tr\left(\mathbf{C}_k^{-1}\mathbf{C}_k\right) \quad \text{(C.10)}$$

$$+\frac{1}{2}tr\left(\mathbf{C}_l^{-1}\mathbf{C}_k\right)+\frac{1}{2}\left(\mu_k-\mu_l\right)^T\mathbf{C}_l^{-1}\left(\mu_k-\mu_l\right) \quad \text{(C.11)}$$

$$\int p\left(\mathbf{x}|l\right)\ln\frac{p\left(\mathbf{x}|l\right)}{p\left(\mathbf{x}|k\right)}d\mathbf{x} = \frac{1}{2}\ln\left(|\mathbf{C}_k|\right)-\frac{1}{2}\ln\left(|\mathbf{C}_l|\right)-\frac{1}{2}tr\left(\mathbf{C}_l^{-1}\mathbf{C}_l\right) \quad \text{(C.12)}$$

$$+\frac{1}{2}tr\left(\mathbf{C}_k^{-1}\mathbf{C}_l\right)+\frac{1}{2}\left(\mu_l-\mu_k\right)^T\mathbf{C}_k^{-1}\left(\mu_l-\mu_k\right) \quad \text{(C.13)}$$

In order to finally obtain

$$D_{sym} = \frac{1}{2}\int p\left(\mathbf{x}|k\right)\ln\frac{p\left(\mathbf{x}|k\right)}{p\left(\mathbf{x}|l\right)}d\mathbf{x}+\frac{1}{2}\int p\left(\mathbf{x}|l\right)\ln\frac{p\left(\mathbf{x}|l\right))}{p\left(\mathbf{x}|k\right)}d\mathbf{x} \quad \text{(C.14)}$$

$$= -\frac{M}{2}+\frac{1}{4}\left(tr\left(\mathbf{C}_k^{-1}\mathbf{C}_l\right)+tr\left(\mathbf{C}_l^{-1}\mathbf{C}_k\right)\right)+\frac{1}{4}\left(\mu_k-\mu_l\right)^T\left(\mathbf{C}_k^{-1}+\mathbf{C}_l^{-1}\right)\left(\mu_k-\mu_l\right) \quad \text{(C.15)}$$

$$\quad \text{(C.16)}$$

Appendix D

Path Integral Approximations - 1D Evaluation

This appendix contain supplementary evaluation of various T-point approximations for the three metrics, futher comments can be found in chapter 4.

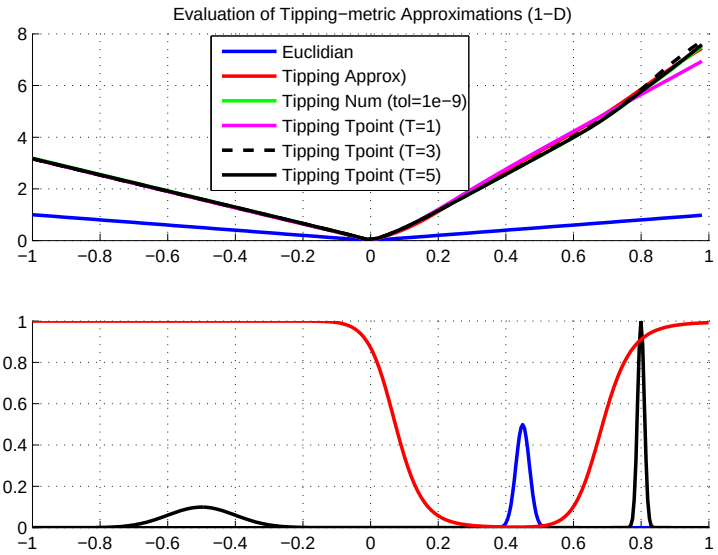


Figure D.1: Tipping approximations.

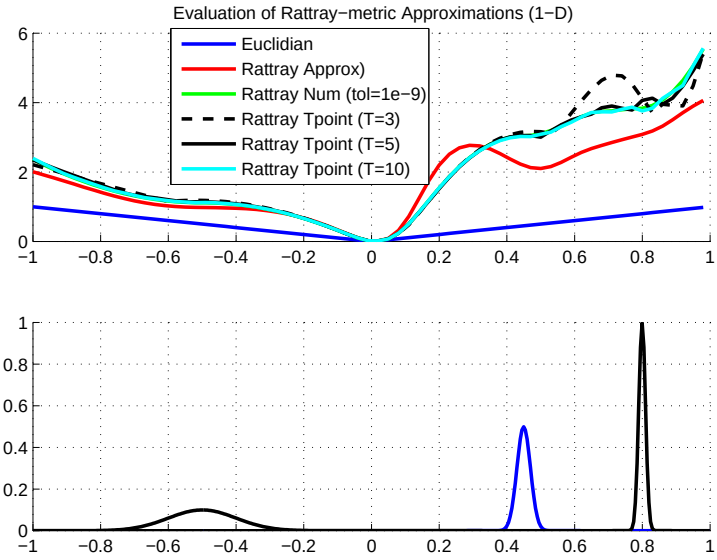


Figure D.2: Rattray approximations.

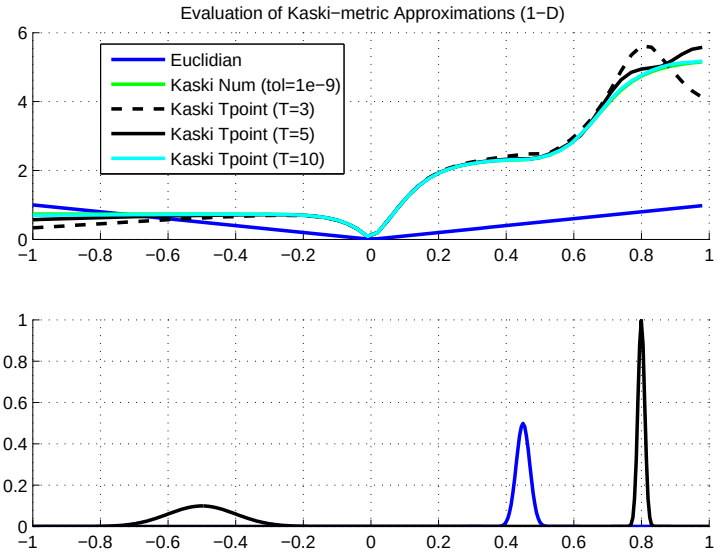


Figure D.3: Kaski approximations.

Appendix E

Extended Clustering Results

This appendix contains the clustering results, obtained in the evaluation of the metrics in chapter 4.

E.1 Curved Data

K=6	**Analyt**	**Num (1e-6)**	**T=1**	**T=5**	**T=15**
Euclidian	0.54±0.042 0.57				
Mahalanobis	0.57±0.1 0.72				
Tipping	0.65±0.095 0.76	0.52±0 0.52	0.52±0 0.52	0.52±0 0.52	0.52±0 0.52
Tipping-Floyd	0.62±0.0088 0.64	0.59±0.02 0.61	0.6±0.028 0.63	0.59±0.027 0.63	0.6±0.028 0.63
Rattray	0.58±0.037 0.59	0.65±0 0.65	0.69±0.036 0.73	0.71±0.037 0.73	0.7±0.041 0.73
Rattray-Floyd	1±0 1	0.99±0 0.99	0.99±0 0.99	0.99±0 0.99	0.99±0 0.99
Kaski		1±0 1	0.8±0.085 0.83	0.83±0 0.83	0.8±0.092 0.83
Kaski-Floyd		1±0 1	0.64±0 0.64	0.64±0 0.64	0.64±0 0.64

Table E.1: Curved Data I: K = 6. Purity of the classes over 10 different k-means initializations including the maximum obtained (as second row)

K=8	**Analyt**	**Num (1e-6)**	**T=1**	**T=5**	**T=15**
Euclidian	0.5±0.063 0.57				
Mahalanobis	0.59±0.085 0.72				
Tipping	0.5±0.048 0.56	0.49±0.056 0.59	0.52±0 0.52	0.52±0 0.52	0.51±0.046 0.52
Tipping-Floyd	0.48±0 0.48	0.67±0.013 0.68	0.65±0.025 0.68	0.65±0.028 0.68	0.66±0.024 0.68
Rattray	0.76±0 0.76	0.71±0 0.71	0.71±0.093 0.8	0.71±0.087 0.8	0.73±0.076 0.8
Rattray-Floyd	1±0 1	1±0 1	1±0 1	1±0 1	1±0 1
Kaski		0.99±0.007 1	0.84±0 0.84	0.84±0 0.84	0.84±0 0.84
Kaski-Floyd		1±0 1	0.64±0 0.64	0.64±0 0.64	0.64±0 0.64

Table E.2: Curved Data: K = 8. Purity of the classes over 10 different k-means initializations including the maximum obtained (as second row)

K=10	**Analyt**	**Num (1e-6)**	**T=1**	**T=5**	**T=15**
Euclidian	0.53±0.048 0.57				
Mahalanobis	0.55±0.11 0.72				
Tipping	0.53±0.02 0.55	0.52±0.093 0.65	0.56±0.074 0.68	0.6±0.033 0.68	0.61±0.028 0.64
Tipping-Floyd	0.52±0.009 0.53	0.69±0.084 0.72	0.83±0.046 0.85	0.81±0.059 0.85	0.8±0.061 0.85
Rattray	0.62±0.099 0.79	0.68±0 0.68	0.7±0.033 0.72	0.71±0.031 0.72	0.71±0.022 0.72
Rattray-Floyd	0.63±0 0.63	1±0 1	1±0 1	1±0 1	1±0 1
Kaski		0.99±0.007 1	0.84±0 0.84	0.84±0 0.84	0.84±0 0.84
Kaski-Floyd		1±0 1	0.59±0 0.59	0.59±0 0.59	0.59±0 0.59

Table E.3: Curved Data: K = 10. Purity of the classes over 10 different k-means initializations including the maximum obtained (as second row)

K=12	**Analyt**	**Num (1e-6)**	**T=1**	**T=5**	**T=15**
Euclidian	0.55±0.043 0.57				
Mahalanobis	0.57±0.1 0.72				
Tipping	0.51±0.053 0.59	0.55±0.047 0.6	0.55±0.049 0.6	0.52±0.024 0.71	0.51±0.095 0.71
Tipping-Floyd	0.59±0 0.59	0.63±0.007 0.64	0.62±0.064 0.65	0.57±0.098 0.65	0.65±0.013 0.65
Rattray	0.67±0.047 0.69	0.79±0 0.79	0.8±0.083 0.91	0.76±0.053 0.91	0.87±0.071 0.91
Rattray-Floyd	0.99±0 0.99	1±0 1	1±0 1	1±0 1	1±0 1
Kaski		1±0 1	0.86±0.017 0.87	0.85±0.017 0.87	0.83±0.065 0.87
Kaski-Floyd		1±0 1	1±0 1	1±0 1	1±0 1

Table E.4: Curved Data I: K = 12. Purity of the classes over 10 different k-means initializations including the maximum obtained (as second row)

K=14	**Analyt**	**Num (1e-6)**	**T=1**	**T=5**	**T=15**
Euclidian	0.51±0.056 0.57				
Mahalanobis	0.56±0.14 0.72				
Tipping	0.43±0.12 0.63	0.5±0.083 0.64	0.42±0.13 0.52	0.49±0.046 0.64	0.47±0.081 0.64
Tipping-Floyd	0.56±0 0.56	0.58±0.058 0.6	0.58±0.058 0.6	0.58±0.058 0.6	0.58±0.058 0.6
Rattray	0.57±0.1 0.67	0.76±0 0.76	0.69±0.0084 0.69	0.68±0.022 0.69	0.68±0.028 0.69
Rattray-Floyd	0.57±0.23 0.91	0.97±0.097 1	1±0 1	1±0 1	1±0 1
Kaski		1±0 1	0.87±0 0.87	0.86±0.013 0.87	0.86±0.017 0.87
Kaski-Floyd		1±0 1	1±0 1	1±0 1	1±0 1

Table E.5: Curved Data I: K = 14. Purity of the classes over 10 different K-means initializations including the maximum obtained (as second row)

K=16	**Analyt**	**Num (1e-6)**	**T=1**	**T=5**	**T=15**
Euclidian	0.49±0.077 0.57	0±0 0	0±0 0	0±0 0	0±0 0
Mahalanobis	0.49±0.15 0.72	0±0 0	0±0 0	0±0 0	0±0 0
Tipping	0.59±0.19 0.77	0.39±0 0.39	0.42±0.058 0.44	0.44±0 0.44	0.44±0 0.44
Tipping-Floyd	0.57±0.0067 0.57	0.43±0 0.43	0.48±5.9e-017 0.48	0.48±5.9e-017 0.48	0.48±5.9e-017 0.48
Rattray	0.31±0.13 0.53	0.44±0 0.44	0.51±0.02 0.83	0.55±0.1 0.83	0.54±0.11 0.83
Rattray-Floyd	0.33±0.026 0.87	0.46±0.19 0.71	0.89±0 0.89	0.89±0 0.89	0.89±0 0.89
Kaski	0±0 0	1±0 1	0.84±0.11 0.91	0.88±0.031 0.91	0.89±0 0.91
Kaski-Floyd	0±0 0	1±0 1	0.42±0.2 0.52	0.42±0.19 0.52	0.47±0.15 0.52

Table E.6: Curved Data: K = 16. Purity of the classes over 10 different k-means initializations including the maximum obtained (as second row)

E.2 Simple Gaussians

K=5	**Analyt**	**Num (1e-6)**	**T=1**	**T=5**	**T=15**
Euclidian	0.61±0.12 0.67				
Mahalanobis	0.72±0.076 0.81				
Tipping	0.8±0.13 0.97	0.83±0.082 0.88	0.83±0.11 0.95	0.85±0.094 0.95	0.86±0.095 0.9
Tipping-Floyd	0.81±0.2 0.96	0.8±0.19 0.96	0.75±0.15 0.95	0.82±0.16 0.95	0.87±0.12 0.95
Rattray	0.86±0.08 0.93	0.57±0.13 0.69	0.71±0.12 0.89	0.76±0.1 0.89	0.75±0.097 0.88
Rattray-Floyd	0.76±0.2 0.92	0.85±0.16 0.97	0.77±0.15 0.98	0.82±0.15 0.98	0.83±0.12 0.98
Kaski		0.97±0 0.97	0.88±0.17 0.97	0.96±0.012 0.97	0.96±0.012 0.97
Kaski-Floyd		0.75±0.091 0.8	0.57±0.1 0.63	0.49±0.14 0.63	0.46±0.13 0.63

Table E.7: Simple Gaussians: K = 5. Purity of the classes over 10 different k-means initializations including the maximum obtained (as second row)

K=7	**Analyt**	**Num (1e-6)**	**T=1**	**T=5**	**T=15**
Euclidian	0.62±0.12 0.67				
Mahalanobis	0.74±0.064 0.78				
Tipping	0.84±0.13 0.96	0.76±0.11 0.9	0.8±0.11 0.9	0.77±0.13 0.9	0.76±0.12 0.9
Tipping-Floyd	0.85±0.13 0.94	0.76±0.12 0.93	0.84±0.12 0.92	0.8±0.13 0.92	0.8±0.13 0.92
Rattray	0.84±0.081 0.89	0.69±0 0.69	0.71±0.093 0.85	0.7±0.08 0.85	0.74±0.077 0.84
Rattray-Floyd	0.81±0.093 0.86	0.78±0.14 0.96	0.79±0.1 0.96	0.7±0.15 0.96	0.74±0.063 0.96
Kaski		0.97±0 0.97	0.89±0.12 0.95	0.89±0.12 0.96	0.93±0.02 0.94
Kaski-Floyd		0.77±0.18 0.97	0.42±0.098 0.62	0.43±0.1 0.62	0.44±0.11 0.62

Table E.8: Simple Gaussians: K = 7. Purity of the classes over 10 different K-means initializations including the maximum obtained (as second row)

K=10	**Analyt**	**Num (1e-6)**	**T=1**	**T=5**	**T=15**
Euclidian	0.58±0.14 0.67				
Mahalanobis	0.68±0.13 0.78				
Tipping	0.76±0.083 0.94	0.7±0.021 0.75	0.68±0.098 0.73	0.71±0 0.73	0.71±0 0.73
Tipping-Floyd	0.71±0.19 0.88	0.71±0.034 0.76	0.65±0.089 0.68	0.68±0 0.68	0.68±0 0.68
Rattray	0.83±0.074 0.89	0.77±0.19 0.95	0.76±0.18 0.95	0.87±0.13 0.95	0.83±0.14 0.95
Rattray-Floyd	0.74±0.088 0.81	0.69±0.17 0.95	0.59±0.14 0.95	0.74±0.15 0.95	0.68±0.13 0.95
Kaski	0±0 0	0.95±0.017 0.95	0.7±0.14 0.82	0.77±0.047 0.82	0.74±0.062 0.83
Kaski-Floyd		0.97±0 0.97	0.48±0.071 0.64	0.46±0.11 0.64	0.44±0.13 0.64

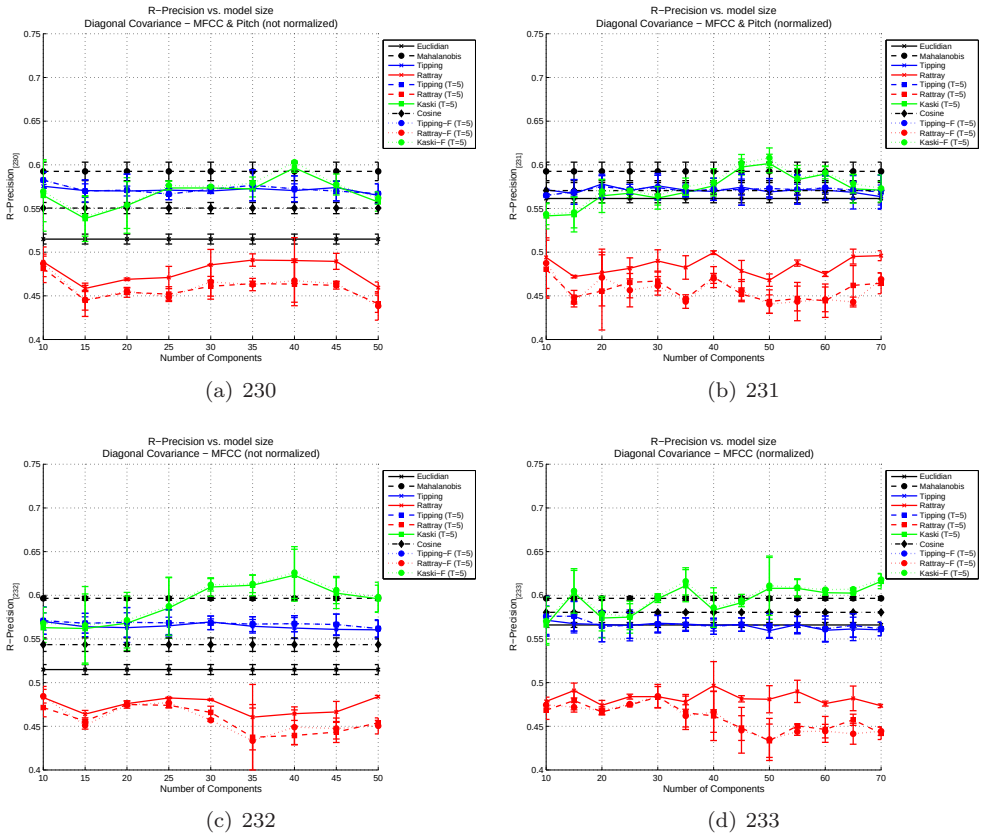
Table E.9: Simple Gaussians: K = 10. Purity of the classes over 10 different k-means initializations including the maximum obtained (as second row)

Appendix F

Retrieval Results - Extra Results

This appendix contains retrieval results left-out in chapter 6

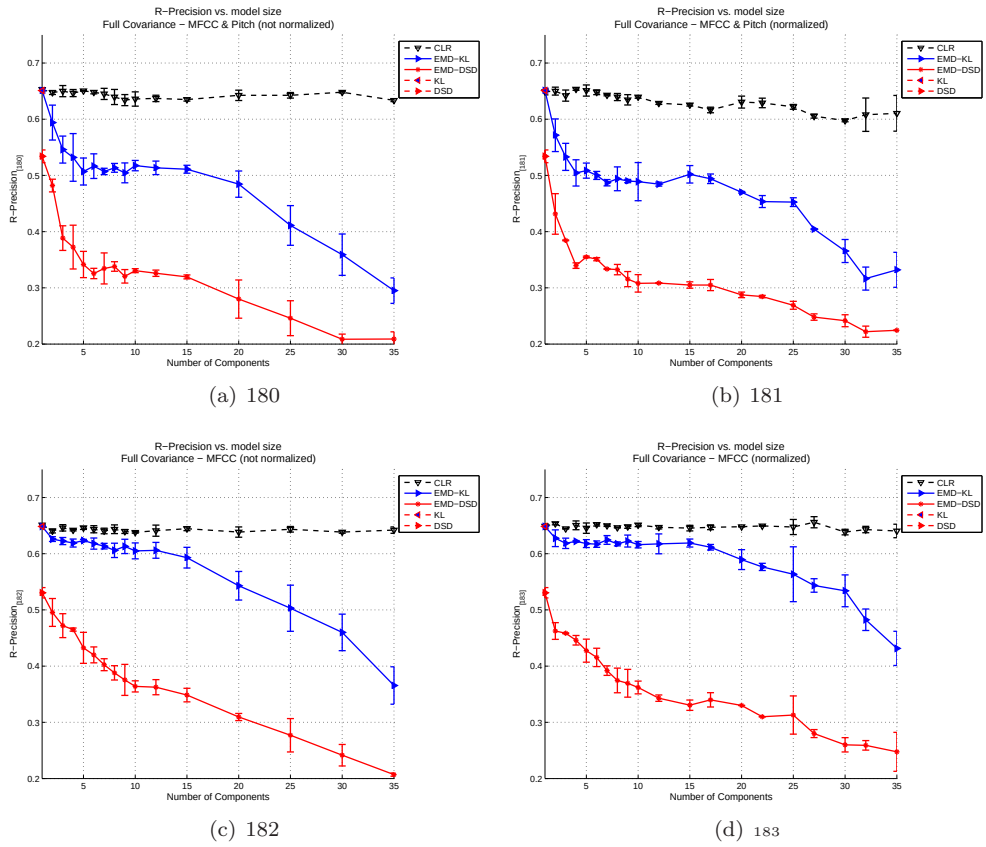
F.1 Clip Retrieval



(a) 230 (b) 231

(c) 232 (d) 233

Figure F.1: Clip Retrieval using Metrics: Diagonal Covariance. Generally not a good performance compared with the full covariances, which is somewhat in contrast to the phoneme data set. It is worth noticing that the non-pitch results provides the better results, indicating that the diagonal models are not able to capture the the discrete-like nature of the pitch.



(a) 180 (b) 181

(c) 182 (d) 183

Figure F.2: Clip Retrieval: Full Covariance. The EMD again has problems with the full covariance, however the basic ground distance, KL and DSD provides a absolute higher than the diagonal case, which is quite noticeable since the EMD in essence becomes more or less useless compared to the choice of using a single full covariance gaussian on the current data set. The maximum obtained results for the CLR is approx. the same as the full covariance but the model sizes tends to be lower (based on the normalized data set).

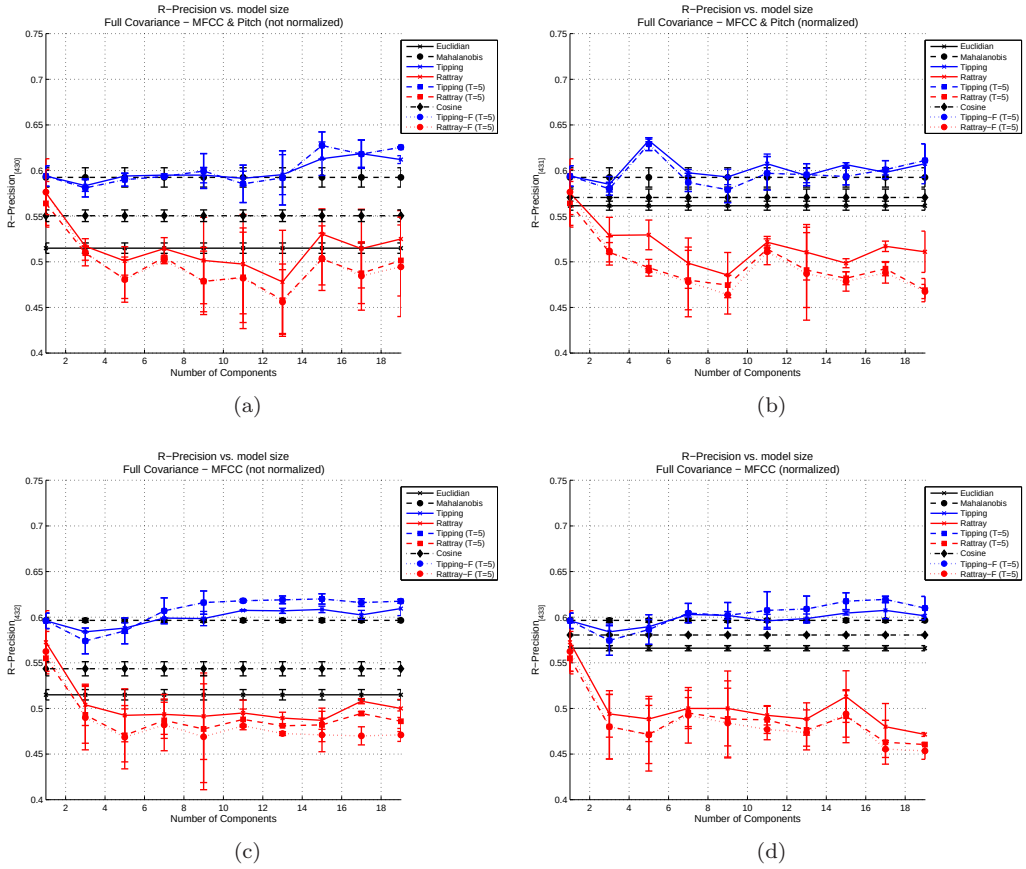


Figure F.3: Clip Retrieval using Metrics: Full Covariance - only unsupervised metrics. The plots show a verification of the unsupervised metrics using smaller models, since the training using supervised models indicated a better performance and trend towards lower model sizes. The results obtained shows that utilizing smaller models does only improve the overall trend of the Rattray metric, which still has major problems in the music set except for the trivial case of one component. The Tipping metric converges to a maximum at around 10 components, at which point we have results from the supervised training. Although an interpolation of results across model types is very dangerous is it still noticeable that the maximum is obtained at 10 components leading to the unsafe conclusion that Tipping does has its maximum at 10 components for the corresponding supervised model and hence does not provide better results than the supervised metric despite the trend of the curve.

Appendix G

Music Dataset - Artists, Songs and Genres

Genre	Artist/Composer	
Classical		
bach00-09	J.S.Bach	Clavier Concerto in F minor Presto
bach10-19	J.S.Bach	Concerto in C major for two claviers Overture
chop00-09	Chopin	Scherzo #2, B minor, Op. 31 - Presto
chop10-19	Chopin	Ballade #4, F minor, Op. 52 - Andante con moto
hayd00-09	Haydn	Symph. #6 - Le Matin - Adagio Andante Adagio
hayd10-19	Haydn	Symph. #8 - Le Soir - Allegro Molto
lisz00-09	Liszt	Symphonic Poem No. 4
lisz10-19	Liszt	Hungarian Rhapsody for Piano No. 5 in e minor
morz00-09	Mozart	Symph. #45, D major, K.46 III. Menuetto-Trio
morz10-19	Mozart	Symph. #46 C major, K.96 III. Menuetto-Trio
niel00-09	C. Nielsen	Symph. #2 - Allegro comodo e flemmatico
niel10-19	C. Nielsen	Symph. #1 - Allegro orgoglioso
tcha00-09	Tchaikovsky	The Nutcracker - Waltz of the Flowers
tcha10-19	Tchaikovsky	The Swan Lake - Waltz
tele00-09	Telemann	Suite in G major
tele10-19	Telemann	Suite in B flat major
viva00-09	A. Vivaldi	Spring (Concerto #1, E Major, Op. 8,1) Allegro
viva10-19	A. Vivaldi	Autumn (Concerto #3, F Major, Op. 8,3) Allegro

Heavy/HardRock

bsab00-09	Black Sabbath	Under The Sun
bsab10-19	Black Sabbath	Iron Man
dist00-09	Distributed	Prayer
dist10-19	Distributed	Breathe
down00-09	System Of A Down	Know
down10-19	System Of A Down	Darts
fear00-09	Empty Vision	Fear Factoty
fear10-19	Millinium	Fear Factoty
guns00-09	Gun's N' Roses	Right Next Door To Hell
guns10-19	Gun's N' Roses	Don't Damn Me
iron00-09	Iron Maiden	Aces High
iron10-19	Iron Maiden	The Number of The beast
juda00-09	Judas Priest	Hell Patrol
juda10-19	Judas Priest	Night Crawler
korn00-09	Korn	Here To Stay
korn10-19	Korn	Bottled Up Inside
mans00-09	Marilyn Manson	Get Your Gunn
mans10-19	Marilyn Manson	Dogma
metl00-09	Metallica	Cure
metl10-19	Metallica	Ronnie

Jazz

cart00-09	Benny Carter	Come On Back
cart10-19	Benny Carter	Titmouse
chet00-09	Chet Baker	Tadd's Delight
chet10-19	Chet Baker	Mating Call
davi00-09	Miles Davis	The Shrpent's Tooth
davi10-19	Miles Davis	No Line
duke00-09	Duke Ellington	The Tattooed Bride
duke10-19	Duke Ellington	Vagabonds
evan00-09	Bill evans	Peri's Scope
evan10-19	Bill evans	Blue In Garden
fats00-09	Fats Waller	Functionizn'
fats10-19	Fats Waller	Sugar Rose
larm00-09	Louis Armstrong	It's All in the Game
larm10-19	Louis Armstrong	Angle Chile
osca00-09	Oscar Peterson	Blues for Smedley
osca10-19	Oscar Peterson	Squeaky's Blues
venu00-09	Joe Venuti	Samba de Orpheus
venu10-19	Joe Venuti	Take The A Train

Trance

daru00-09	Darude	Exstacy
daru10-19	Darude	Sandstorm
fait00-09	Faithless	God Is A DJ
fait10-19	Faithless	Insomnia
ianv00-09	Ian van Dahl	Castles In the Sky
ianv10-19	Ian van Dahl	I Can't Let You Go
infi00-09	Infinity (Juan Atkis)	Skyway
infi10-19	Infinity (Juan Atkis)	Body Oil
infr00-09	Infernal	Kalinka
infr10-19	Infernal	Hammond Place
paul00-09	Paul Van Dyk	Autumn
paul10-19	Paul Van Dyk	Out There
perx00-09	Percy X	Break It Down
perx10-19	Percy X	By Night
sduo00-09	Safri Duo	Amazonas
sduo10-19	Safri Duo	Prelude
svgi00-09	Svenson & Gielen	The Beauty of Silence
svgi10-19	Svenson & Gielen	Twisted
sysf00-09	System F	Soul on Soul
sysf10-19	System F	Out Of The Blue

Pop/SoftRock

adam00-09	Brian Adams	Summer of '69
adam10-19	Brian Adams	Run to you
card00-09	Cardigans	Erase and rewind
card10-19	Cardigans	Hanging Around
cold00-09	Coldplay	Low
cold10-19	Coldplay	Talk
eury00-09	Eurytmics	Thor In My Side
eury10-19	Eurytmics	When Tomorrow Comes
garb00-09	Garbage	Special
garb10-19	Garbage	Temptation Waits
inxs00-09	INXS	Suicide Blonde
inxs10-19	INXS	Heaven Sent
mado00-09	Madonna	Like a Virgin
mado10-19	Madonna	Papa Don't Preach
robw00-09	Robbie Williams	Hot Fudge
robw10-19	Robbie Williams	Rock DJ
stmc00-09	Stereo MC	Fade Away
stmc10-19	Stereo MC	Step It Up
utwo00-09	U2	Zoo Station
utwo10-19	U2	Even Better Than the real thing

Appendix H

Feature Plots - Detailed view of the POP genre

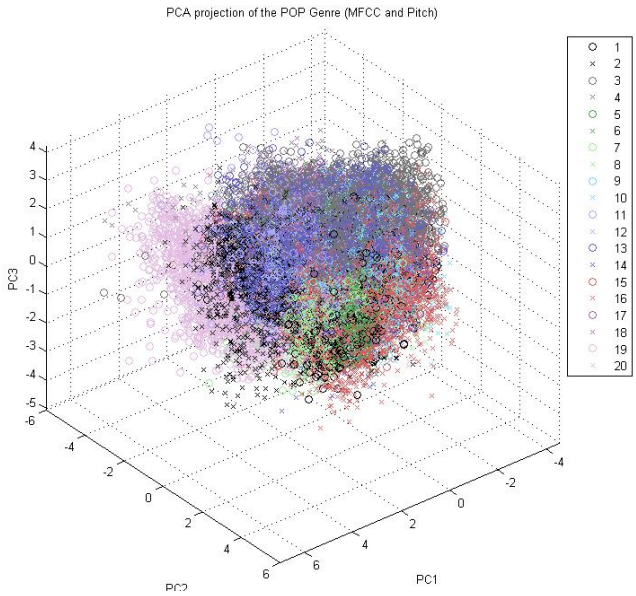


Figure H.1: PCA projection of the pop genre. Songs (from the same artist (two) are the same color/nuance, only differenc eis the marker style.

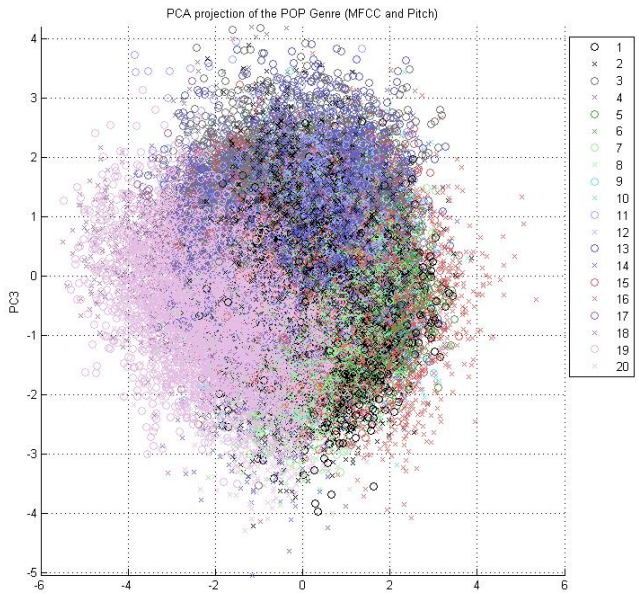


Figure H.2: PCA projection of the pop genre. Songs (from the same artist (two) are the same color/nuance, only differenc eis the marker style.