Objective Speech Quality Estimation using Gaussian Mixture Models

by

Tiago Henrique Falk

A thesis submitted to the
Department of Electrical and Computer Engineering
in conformity with the requirements for
the degree of Master of Science (Engineering)

Queen's University
Kingston, Ontario, Canada
May 2005

Abstract

In this thesis, we propose the use of Gaussian mixture models (GMMs) as simple, yet effective predictors of perceived speech quality. A large pool of perceptual distortion features is extracted from speech files. Initially, statistical data mining algorithms are used to sift out the most relevant variables from the pool. We show that the five most salient feature variables are sufficient to construct good GMM-based estimators of subjective listening quality. It is shown, however, that the features selected by the data mining schemes limit the performance of the proposed voice quality predictor. To this end, a novel feature selection algorithm that directly optimizes GMM prediction performance is also proposed. The algorithm performs N-survivor search, trading complexity and accuracy via the parameter N. Comparisons with PESQ, the current "state-of-art" speech quality estimation algorithm, show that the proposed algorithm incurs, on average, 26.12% higher correlation and 18.04% lower root-mean-squared error. Tested on unseen data the proposed algorithm is capable of reducing $RMSE$ by an average 41% relative to PESQ.

Acknowledgements

First, I would like to express my sincerest gratitude to Dr. Wai-Yip Geoffrey Chan, who during these past two years has served not only as my supervisor, but as my mentor. His profound insights and knowledge have greatly influenced my academic career. I will always be thankful for his help and I look forward for another four years of doctoral studies under his supervision.

I would also like to thank OAS/LASPAU for funding my studies. To all members of my examination committee, Multimedia Compression Lab, ECE staff and faculty, "Thank you!" I would not have been here today if it were not for the love and care of my family. To my parents who have always supported, encouraged and taught me to be the person I am today; to my siblings who have always stood by my side, a special "Thank you" goes out your way! Moreover, as a small token of my gratitude and love, I dedicate this thesis to my late grandfather Paulo.

The most special "Thank you" I have saved for last. I saved it for the person who left all behind to be by my side as we moved to Canada; the person who has been by my side, in the happiest and in the toughest moments. For the person who spent sleepless nights as I was away but, nonetheless, cheered and encouraged that I work harder to be able to attend international conferences. This special person is my wife, Andrezza Falk. Thank you for all your support. **I love you**!!!

Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Description
ACR	Absolute Category Rating
BSD	Bark Spectral Distortion
CART	Classification And Regression Trees
CCR	Comparison Category Rating
CMOS	Comparison Mean Opinion Score
DCR	Degradation Category Rating
DMOS	Degradation Mean Opinion Score
EM	Expectation-Maximization Algorithm
EVRC	Enhanced Variable Rate Codec
GMMs	Gaussian Mixture Models
ITU-T	International Telecommunications Union – Telecommunications
MARS	Multivariate Adaptive Regression Splines
MNB	Measuring Normalizing Block
MOS	Mean Opinion Score
PESQ	Perceptual Evaluation of Speech Quality
PSQM	Perceptual Speech Quality Measure
SDMQA	Statistical Data Mining Quality Assessment
SFS	Sequential Forward Selection Algorithm
SNR	Signal-to-Noise Ratio
VoIP	Voice Over Internet Protocol

Chapter 1

Introduction

This thesis proposes a novel predictor of perceived speech quality. More specifically, Gaussian mixture densities are employed to map perceptual features extracted from speech signals to a quality rating. In this chapter, the motivations for this work are described in Section 1.1. Section 1.2 summarizes the major contributions of this thesis. Lastly, Section 1.3 outlines how this thesis is organized.

1.1 Motivations

The telecommunications industry is going through a phase of rapid development. According to Infonetics Research,[1] in 2004 cable voice-over-internet use rose an astounding 900% when compared to 2003. Wireless telephony use rises about an average 14.5% a year. New technologies are emerging continuously; voice-over-internet, or VoIP, is the fastest growing telephony service today, bringing in revenues in the order of several billions of dollars a year. This technological boom has left networks that

[1]Infonetics Research is an international market research firm covering the data networking and telecommunications industries. More information can be found at www.infonetics.com

are heterogeneous and complex, making it extremely difficult for telephone service providers to identify the root cause of voice quality problems. Since speech quality is a major contributor to customer satisfaction, measurement of quality has become critically important for the service provider. In fact, for wireless carriers, infrastructure and maintenance costs are also directly related to customer satisfaction [1]. Today, the demand for newer and more efficient methods of measuring the quality of voice signals is on the rise, motivating the research described in this thesis.

Traditionally, the most reliable way to measure the quality of a speech signal was through the use of subjective speech quality assessment tests. In such tests, human listeners are asked to rate the quality of the speech signal they just heard according to a five-point scale. The average of the listeners scores is the subjective mean opinion score (MOS) [2, 3]. Subjective tests are highly unsuitable for online quality measurement and are also very expensive and time consuming. The research described in this thesis is motivated by the fact that objective methods have replaced subjective testing, allowing computer programs to automate speech quality measurement in real time, making them suitable for field applications.

In a nutshell, objective speech quality assessment consists of extracting perceptual features that carry information regarding the quality of a speech signal. Feature extraction, combined with a mapping from these features to a quality rating emulates the "cognitive" behavior of a human's perception of speech quality. The International Telecommunications Union ITU-T P.862 standard, also known as Perceptual Evaluation of Speech Quality (PESQ) [4] is the latest objective quality measurement standard algorithm. Nevertheless, the algorithm still falls short of the accuracy that can be obtained from subjective listening tests with sizable listener panels.

Recently, data mining techniques have been proposed to improve the accuracy of auditory-model based quality measurement [5]. A large number of perceptual features are extracted to create a rich pool of candidate features. Feature selection algorithms are used to discard noisy or redundant features. The selected features are then mapped to an objective MOS. Objective methods aim to deliver MOSs that are highly correlated with the MOSs obtained from subjective listening experiments. To this end, new mappings are sought that maximize quality prediction accuracy. Newer and more efficient algorithms that can iteratively perform feature selection, whilst updating the mapping functions, could open doors to newer generations of speech quality predictors.

1.2 Thesis Contributions

This thesis contributes to the field of objective speech quality assessment in at least two instances:

1. Gaussian mixture models (GMMs) are proposed as possible mapping functions between perceptual features and perceived voice quality. A total of 209 perceptual features are extracted from a speech signal and, initially, common feature selection algorithms, such as classification and regression trees (CART) and multivariate adaptive regression splines (MARS) are used to select the most salient features. Simulation shows that GMM-based speech quality predictors outperform PESQ, the current "state-of-art" voice quality predictor algorithm.

   By using CART and MARS for feature selection, the first steps towards an efficient algorithm are taken. Careful analysis of the proposed GMM estimators,

however, show that the proposed algorithm has certain limitations, e.g., diagonal covariance GMMs only provide modest improvement over PESQ. These limitations occur, mainly, due to characteristics present in the features selected by CART or MARS. The motive is simple, CART/MARS selects features that are optimal for CART or MARS mapping and not for GMM mapping. Nevertheless, full covariance GMMs are shown to overcome these shortcomings. Publications that arise from this contribution are [6–8].

2. A feature selection algorithm, that directly optimizes GMM estimation performance, is proposed and demonstrates that there exists a considerable gap in performance relative to GMM estimators trained on CART/MARS selected features. The algorithm performs N-survivor search allowing the compromise between complexity and performance to be adjusted via the parameter N. If D features are desired, the algorithm iterates D times. With single survivor search, the single feature that is chosen per iteration is the one that minimizes squared GMM regression error. It is shown that, at the cost of higher computational complexity, performance improvement can be obtained with the use of N-survivor search, $N > 1$. Considerable improvement over PESQ is reported. Publications relating to this contribution are [9, 10].

1.3 Outline of Thesis

Chapter 2 introduces "state-of-art" speech quality measurement methods, including subjective and objective methods. Section 2.2 introduces three types of subjective tests, as prescribed in ITU-T Recommendation P.800/P.830; Section 2.3 presents

objective methods. Methods that fall in the class of intrusive measurement are described in Section 2.3.1, together with the description of PESQ and SDMQA. Section 2.3.2 covers non-intrusive measurement methods and briefly presents examples of parameter-based models and signal-based algorithms.

Chapter 3 introduces the techniques of Gaussian mixture models. GMM properties and definitions are given in Section 3.2. GMM parameter estimation is the focus of Section 3.3 where the expectation-maximization (EM) algorithm is introduced. GMM-based estimators, the heart of this thesis, are presented in Section 3.4.

In Chapter 4, the first steps towards a novel GMM-based speech quality predictor are given. A large pool of 209 feature variables are extracted from speech signals and CART or MARS are used to select the top ranked features to be mapped by a GMM estimator. A detailed description of the proposed algorithm is given in Section 4.2. Advantages and disadvantages of the proposed algorithm are discussed in Section 4.3.

Chapter 5 describes an improved GMM-based voice quality measurement algorithm. The features selected by CART or MARS are shown to limit the performance of the proposed estimator and a new feature selection algorithm that directly optimizes GMM estimation is proposed. Section 5.2 describes the proposed feature selection algorithm. Section 5.3 evaluates the proposed algorithms by comparing with GMM estimators trained on features selected by CART or MARS. Comparisons to PESQ are also carried out in this section.

Lastly, Chapter 6 provides the conclusions of this thesis and suggests possible future research directions.

Chapter 2

Speech Quality Assessment

2.1 Introduction

Speech quality is a major contributor to the end user's perception of quality of service. As networks become more heterogeneous and complex, and new technologies interoperate with legacy equipment, identifying the root cause of voice quality problems can be a challenging task. The evaluation and assurance of speech quality has, consequently, become critically important for telephone service providers, especially for wireless carriers whose infrastructure and maintenance costs are directly related to customer satisfaction [1].

Voice quality is a subjective opinion, based on the user's reaction to the speech signal they actually heard. Subjective methods make use of a listener panel to measure speech quality on to a scale from 1 to 5, with 1 corresponding to unsatisfactory speech quality and 5 corresponding to excellent speech quality. The average of the listener scores is the subjective Mean Opinion Score, MOS [2, 3]. This has been the most reliable method of speech quality assessment but it is very expensive and time

consuming, making it unsuitable for frequent or rapid applications. These shortcomings can be overcome by using objective measurement methods, which replace the listener panel with a computational algorithm. Objective methods aim to deliver MOSs that are highly correlated with the MOSs obtained from subjective listening experiments.

Objective quality assessment tests can be classified as *intrusive* or *non-intrusive* as shown in Figure 2.1. Intrusive measurement depends on some form of comparison between the reference and degraded speech signals to predict the subjective MOS. The International Telecommunications Union ITU-T P.862 standard, also known as Perceptual Evaluation of Speech Quality (PESQ) [4] is the latest intrusive objective quality measurement standard algorithm.

In some situations an intrusive approach may not be applicable because the input speech signal may be unavailable. Non-intrusive measurement depends only on the degraded speech signal and can be further classified as signal-based or parameter-based. Signal-based approaches predict voice quality by directly analyzing the degraded speech signal. Models have been proposed in [11–13], and only recently has ITU-T released P.563 as its non-intrusive objective quality measurement standard algorithm [14]. Parameter-based methods can predict communication quality directly from network and/or terminal parameters; such models include the ITU-T E-model [15]. Moreover, a recently proposed method by Sun [16] estimates listening quality via parameter-based methods.

In this chapter, subjective quality measurement methods are described in Section 2.2. Objective methods will be presented in further detail in Section 2.3, including intrusive (Section 2.3.1) and non-intrusive methods (Section 2.3.2). This thesis will

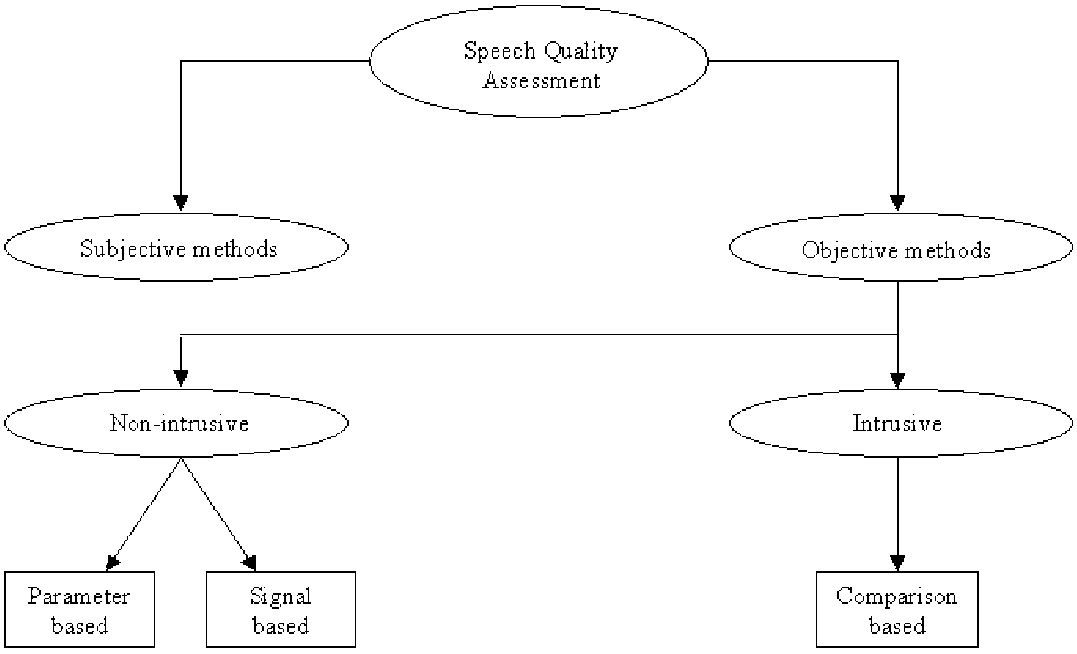


Figure 2.1: Classification of speech quality assessment methods

focus on a novel intrusive method for speech quality prediction.

2.2 Subjective Speech Quality Assessment

Traditionally, the most reliable way to measure the quality of a speech signal was through the use of subjective speech quality assessment tests. A speech file is played to a group of listeners, who are asked to rate the quality of this speech signal. Subjective tests are very reliable, given that a large listening panel is used. In [17], factors, such as listener variability, are shown to affect the reliability of subjective tests. Larger listener panels are shown to improve accuracy and repeatability of subjective voice quality assessment. In most tests, the number of listeners ranges from 16 to 64

listeners (half male, half female), where the maximum limit is established by cost and time limitations [18]. Today, as computational algorithms quickly replace listener panels, subjective tests are still performed serving as benchmarks for newly proposed objective methods.

Clearly, subjective methods are important in the task of speech quality estimation. ITU-T Recommendations P.800 [3] and P.830 [2] contain methods and guidelines for conducting subjective evaluations of transmission quality in order to obtain reliable and reproducible test results. P.800 describes three methods of listening-opinion subjective tests: absolute category rating (ACR), degradation category rating (DCR) and comparison category rating (CCR). Both P.800 and P.830 describe the necessary controlled laboratory conditions in which tests are to be carried out.

2.2.1 Absolute category rating (ACR) test

ACR testing is the most commonly used subjective method and listeners are instructed to rate the processed speech material presented to them according to a 5-point listening quality scale as shown in Table 2.1. Listeners are not given reference speech files for comparisons and are asked to rate the "absolute" quality of the speech samples. The average of the listener scores is the subjective mean opinion score, MOS.

2.2.2 Degradation category rating (DCR) test

In DCR testing, listeners are presented with a clean reference speech signal before they are presented with the processed signal. The listeners are then instructed to rate the perceived degradation of the processed speech material when compared to

Table 2.1: Subjective opinion scale for ACR testing

Category	Speech Quality
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

Table 2.2: Subjective opinion scale for DCR testing

Category	Level of Degradation
5	Imperceptible
4	Just perceptible but not annoying
3	Perceptible and slightly annoying
2	Annoying but not objectionable
1	Very annoying and objectionable

the unprocessed material. Ratings are also based on a 5-point listening quality scale as shown in Table 2.2. This type of test is more sensitive to degradations introduced by the system being tested [18] and are suitable for evaluating good speech quality [3]. The average of the listener scores composes the degradation MOS, also known as DMOS. As opposed to MOS, different experiments using DMOS ratings can only be compared if they share the same reference signals, or different reference signals but of the same quality [19].

2.2.3 Comparison category rating (CCR) test

The CCR test, a refinement of the DCR test, asks listeners to identify the quality of the processed speech sample relative to its unprocessed counterpart using a two-sided

Table 2.3: Subjective opinion scale for CCR testing

Category	Quality of Second Sample Compared to Quality of First Sample
3	Much better
2	Better
1	Slightly better
0	About the same
-1	Slightly worse
-2	Worse
-3	Much worse

rating scale, as given by Table 2.3. For CCR testing, in half of the trials, the unprocessed sample is followed by the processed sample. On the remaining trials, the order is reversed. In effect, two judgements are provided with one response: "which sample has better quality?" and "how much better is it?" The CCR testing improves on DCR as it provides the possibility to assess speech processing that can either degrade (e.g., speech compression algorithms) or improve (e.g., speech enhancement algorithms) the quality of speech. Most importantly, though, CCR testing is used to minimize biases that occur due to the order in which the speech materials are presented in the DCR test. The average of the listener scores is the comparison MOS or CMOS.

All MOS tests have to be carried out in restricted laboratories, following strict norms. These requirements are necessary in order to obtain accurate and repeatable results. Unfortunately, these requirements make subjective tests very expensive and time consuming, i.e., unsuitable for frequent or rapid applications. Today, most of the research in speech quality measurement, in one way or another, tries to identify and model audible distortions through an objective process based on human perception. Objective methods can be implemented by computer programs and can be used in

real time measurement of speech quality. Objective speech quality assessment is the topic of the next section.

2.3 Objective Speech Quality Estimation

Objective speech quality assessment has replaced expensive and time consuming subjective methods, allowing computer programs to automate speech quality measurement in real time, making them suitable for field applications. In fact, objective measurement is the only viable means of measuring voice quality, for the purpose of real-time call monitoring, on a network-wide scale.

Objective algorithms can be classified as intrusive or non-intrusive as shown in Figure 2.2. Intrusive measurement systems depend on some form of distance metric between two input signals – a reference (clean) and a degraded speech signal at the output of the system under test – to predict the subjective MOS, namely, $\hat{MOS}$. These systems are referred to as intrusive due to the injection of a voice signal known to the algorithm into the transmit end. Non-intrusive measurement, on the other hand, depends only on the degraded speech signal to find $\hat{MOS}$. Non-intrusive methods are often referred to as "passive" in the sense that a test signal is not required.

2.3.1 Intrusive Speech Quality Estimation

Over the years, many different speech parameters have been used to measure voice quality [20]. Classical waveform speech coding algorithms used signal-to-noise ratio (SNR) and the segmental SNR [21] to estimate the quality of speech in waveform-preserving systems. Newer generation speech coders do not preserve the waveform so

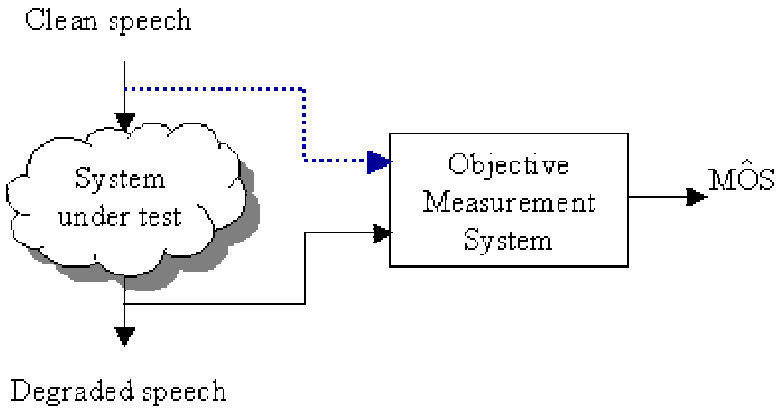


Figure 2.2: Objective quality measurement: intrusive (solid + dashed lines) and non-intrusive (solid line only)

these measures are of little relevance. In [17], measures of distortions in the short-time spectral envelopes of speech are introduced such that matching waveforms are not needed in order to produce zero distortion.

Measurement algorithms that exploit human auditory perception rather than just the acoustic features of speech provide more accurate prediction of subjective quality. This has been the focus of current objective quality assessment research. It is known that the peripheral auditory system of human preprocesses information and "compact" feature extraction is done in higher-level brain functions. The human decision is based on this compacted data. An adequate model should emulate this biological preprocessing and higher-level functions, and deliver ratings that have high correlations with the subjective results. The preprocessing part is relatively well understood but the higher-level brain functions are difficult to model.

Various algorithms and schemes have attempted to model the higher-level brain

functions, including BSD (Bark spectral distortion) [22], MNB (measuring normalizing block) [23, 24], PSQM (perceptual speech quality measure) [25], and the current "state-of-art" PESQ (perceptual evaluation of speech quality) [4]. Recently, in [5], an approach is introduced that uses statistical data mining techniques to improve the accuracy of auditory-model based quality measurement; performance improvement over PESQ was reported.

The major difference among the aforementioned algorithms is in the post-processing of the auditory error surface. MNB uses a hierarchical structure of integration over a range of time and frequency intervals. PESQ performs integration in three steps, first over frequency, then over short-time utterance intervals and finally over the whole speech signal. Different p values are used in the L_p norm integrations of the three steps. The different methods of integration, though they may not resemble cognitive processes, achieve their respective degrees of effectiveness through using subjectively scored speech data to calibrate the mapping from the integrated measure to the estimated subjective MOS.

The method proposed in [5] (henceforth denoted SDMQA – statistical data mining quality assessment) differs from the above algorithms in the sense that it classifies perceptual distortions under a variety of contexts. Distortions with the same context are integrated to a value which the authors call a "feature". A large pool of context-dependent feature measurements is created and statistical data mining tools are used to find salient features in the pool. Features are selected to produce the best estimator of the subjective MOS value.

In the sequel, PESQ and SDMQA are described in more detail. The objective of this thesis is to propose a novel intrusive speech quality measurement algorithm using

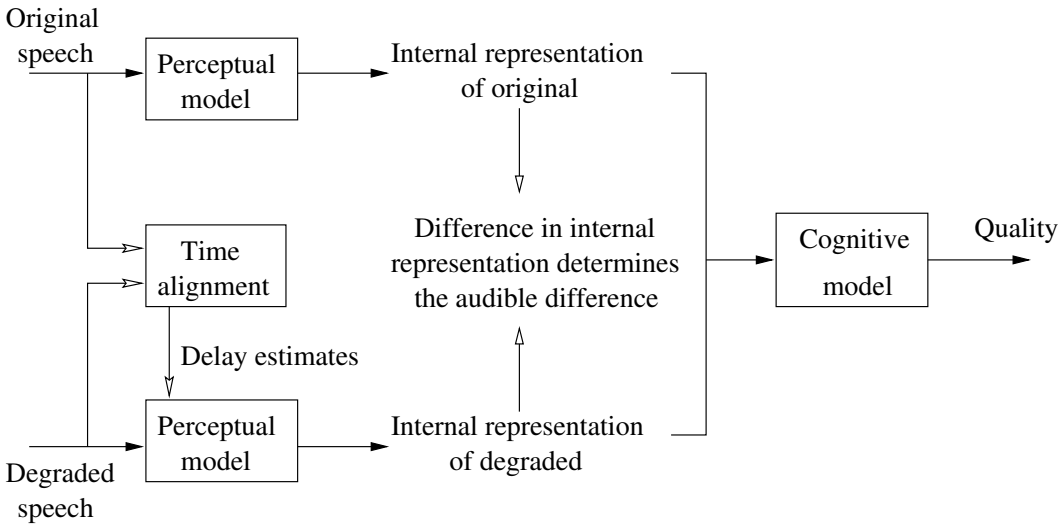


Figure 2.3: PESQ algorithm architecture

SDMQA for feature extraction. For this reason a more in-depth description of the SDMQA algorithm is given. PESQ is used for benchmarking the proposed algorithm performance so a more concise description of the algorithm is given.

2.3.1.1 Perceptual Evaluation of Speech Quality (PESQ)

The International Telecommunications Union ITU-T P.862 standard, also known as Perceptual Evaluation of Speech Quality (henceforth PESQ) [4] is the latest objective quality measurement standard algorithm. PESQ replaced its predecessor, PSQM, in 2001 as it was capable of identifying time delay, allowing alignment between the original and degraded speech signals. Delay compensation is essential for quality measurement of voice packets that are subject to delay variation in the network. Furthermore, PESQ, unlike PSQM, directly calculates the objective MOS score on a 5-point scale. An overview of the PESQ algorithm architecture is depicted in Figure 2.3.

The first step of PESQ is to compute a series of delays between the original input and the degraded output. Based on these delays, PESQ compares the original signal with the aligned degraded output using a perceptual model. The signals are transformed to a representation that is analogous to the psychophysical representation of audio signals in the human auditory system by means of perceptual frequencies and compressive loudness scaling. A more in-depth description of the psychophysical (internal) representation of audio signals is given in Section 2.3.1.2. The difference in internal representations of the degraded and reference speech signals is then calculated, representing the audible difference between the two signals. A positive difference indicates components such as noise are present; a negative difference indicates components have been omitted.

Lastly, the cognitive model evaluates audible errors by computing noise disturbances for the individual time-frequency bins. Two types of disturbances are calculated: asymmetrical and symmetrical. Asymmetry processing uses a scaling factor to apply different weights to positive and negative disturbances. The asymmetrical disturbance is such that only bins with positive disturbances remain. The symmetrical disturbances are calculated by averaging, over frequency bands, a measure of absolute audible errors and asymmetrical disturbances, and a measure of audible errors that are significantly louder than a reference threshold.

The predicted MOS is calculated as a linear combination of the average disturbance value and the average asymmetrical frame disturbance value. The final PESQ score falls in the range of -0.5 and 4.5. It is stated in [26] that in most cases the output range will be a MOS-like score between 1 and 4.5. Nonetheless, a mapping from the PESQ raw score to the ACR listening quality scale is proposed in [27]. The

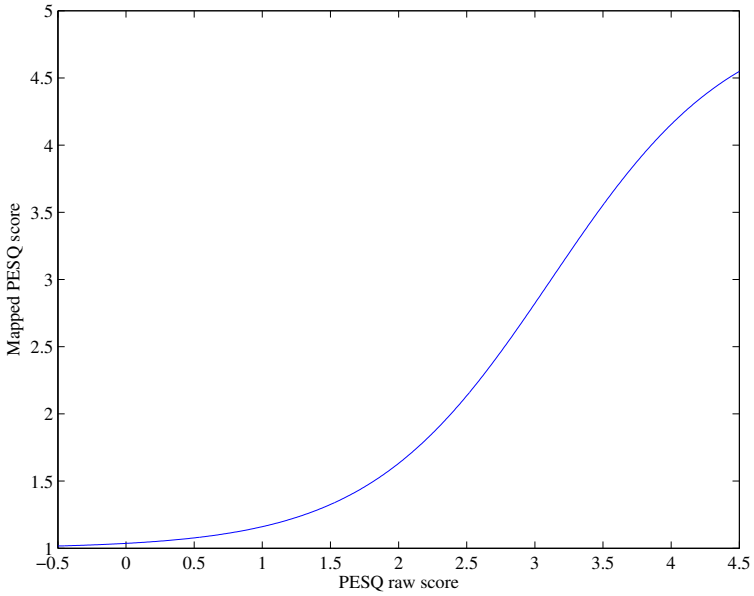


Figure 2.4: Mapping from PESQ raw score to ACR listening quality scale

mapping is given by the following equation:

$$\hat{MOS} = 0.999 + \frac{4}{1 + \exp(-1.4945 * PESQ_{raw} + 4.6607)} \tag{2.1}$$

and is depicted in Figure 2.4.

2.3.1.2 Statistical Data Mining Quality Assessment (SDMQA)

The architecture of the SDMQA scheme is depicted in Figure 2.5. Similar to PESQ, the auditory processing modules decompose the input speech signals into power distributions over time-frequency and then convert them to auditory excitations on a loudness scale. The cognitive mapping module interprets the differences (distortions) between the auditory excitations of the clean and the degraded speech signals. In effect, the cognitive module "integrates" the distortions over time and frequency to arrive at a predicted quality score.

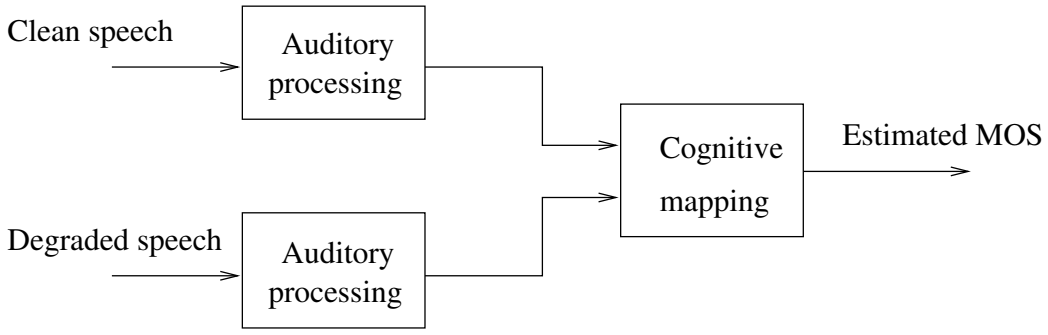


Figure 2.5: SDMQA algorithm architecture

In SDMQA, a plethora of contexts under which distortion events occur is created. Distortions with the same context are integrated to a value which the authors call a feature. Straightforward L_2-norm integration (i.e., root-mean-squared) is used to compute the feature value. From the pool of candidate features, data mining techniques are used to find a small subset of features and the best way to combine them to estimate the speech quality.

The auditory processing block found in Figure 2.5 can be further decomposed as depicted in Figure 2.6. Human auditory processing of acoustic signals is commonly modelled by signal decomposition through a bank of filters whose bandwidths increase with filter center frequency according to the bark scale [28]. Seven bands, each with bandwidth of about 2.4 bark are used and a 128-point FFT is used to produce a 7-point power spectrum every 10 ms. The spectral power coefficients are grouped into 7 bands and the coefficients in each band are summed to produce a total of 7 subband power samples. The samples are then converted to the subjective loudness density using Zwicker's power law [28]:

$$L(f) = L_0 \left(\frac{E_{TQ}(f)}{s(f)E_0} \right)^k \left[\left(1 - s(f) + \frac{s(f)E(f)}{E_{TQ}(f)} \right)^k - 1 \right] \tag{2.2}$$

where the exponent $k = 0.23$, $L_0 = 0.068$, E_0 is the reference excitation level, $E_{TQ}(f)$

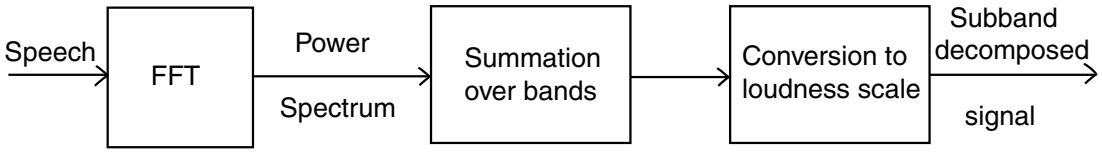


Figure 2.6: Processing steps in an auditory processing module

is the excitation threshold at frequency f, $E(f)$ is the input excitation at frequency f, and $s(f)$ is the threshold ratio.

The "cognitive mapping" block can be decomposed as illustrated in Figure 2.7. The decomposed clean and degraded speech signals from the auditory processing modules are first subtracted to obtain their absolute difference, which is called the "distortion." The distortion over the whole speech signal can be organized into a two-dimensional array, representing a distortion surface over time-frequency. The goal of the cognitive mapping is to aggregate cognitively similar distortion events through time segmentation and distortion severity classification.

Time segmentation labels the speech frames as "active" or "inactive." Active frames are further classified into voiced or unvoiced. Segmentation separates the different types of speech frames so that they can exert separate influence on the speech quality estimate. The total distortion of each frame is given severity classifications of "low", "medium", or "high" by simple thresholding. The aim is to sift out the significant distortion events. Distortion samples in time-frequency bins are thus labelled according to their frequency band, time-segmentation type, and severity level. The distortion samples with the same composite-label value belong to the same context. For instance, the above seven subbands, three time segmentation labels and three distortion classification labels are combined to create $7 \times 3 \times 3 = 63$ contexts. Each context contributes a feature to the candidate pool for mining.

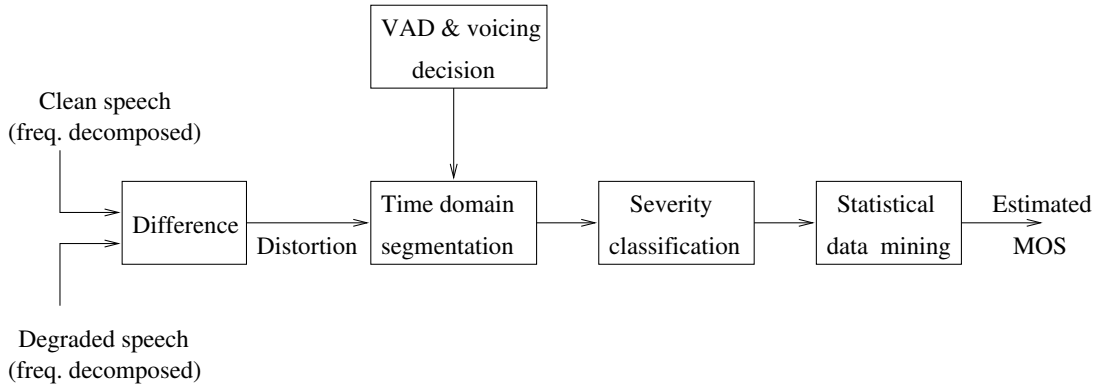


Figure 2.7: Cognitive mapping

Additional contexts are defined in order to create a "rich" pool of candidate features for mining. Besides labelling each frequency subband with its natural subband index, each subband is also labelled with the rank order obtained by ranking the seven distortions in a frame in order of decreasing magnitude. Rank ordering the subband distortions, as well as classifying frame-level distortions based on severity, create contexts that capture distortions independent of specific time-frequency locations, but dependent on the absolute or relative level of distortion severity.

Additional contexts are also created by omitting some labels such as the severity level. These contexts are the seven subbands, in natural or ordered index, for each of the three time-segmented frame classes, without severity classification; altogether there are 7 x 3 = 21 such contexts. Weighted mean and root-mean distortions, probability of each frame type, and the lowest-frequency-band and highest-frequency-band energy of the clean speech frames are also created to produce a pool of 209 candidate features. The 209 feature variables are listed in Appendix A.

SDMQA resorts to data mining techniques to sift a smaller subset of features

($\ll 209$) that are sufficient for the speech quality task at hand. Note that the statistical data mining block in Figure 2.7 is for the design phase only. Once feature mining is performed, the block is replaced by a simple mapping block in the testing phase. SDMQA utilizes classification and regression trees (CART) [29] and multivariate adaptive regression splines (MARS) [30] to map selected features to a MOS score.

2.3.2 Non-Intrusive Speech Quality Estimation

Unlike intrusive methods described in Section 2.3.1, non-intrusive methods do not require the injection of a reference signal and are appropriate for monitoring live traffic. Non-intrusive quality measurement deviates from the scope of this thesis and is briefly introduced here for the sake of completeness. The references given here should supply the reader with sufficient guidance to material pertinent to the topic.

As shown in Figure 2.1, non-intrusive methods fall into two categories: signal-based or parameter-based quality assessment. Signal-based approaches predict voice quality by directly analyzing the degraded speech signal. Parameter-based methods predict subjective MOS directly from IP network impairment parameters (e.g., packet loss, jitter) and non-IP network parameters (e.g, codec, echo). The purpose is to establish a relation between perceived voice quality and network related "distortions."

Signal-based models have been proposed in [11–13], and only recently has ITU-T released P.563 as its non-intrusive objective quality measurement standard algorithm [14]. P.563 resulted from a collaboration of Psytechnics' NiQA algorithm [31], SwissQual's NiNA [32], and Opticom's P3SQM.

The signal parameterization in P.563 is divided in three independent functional

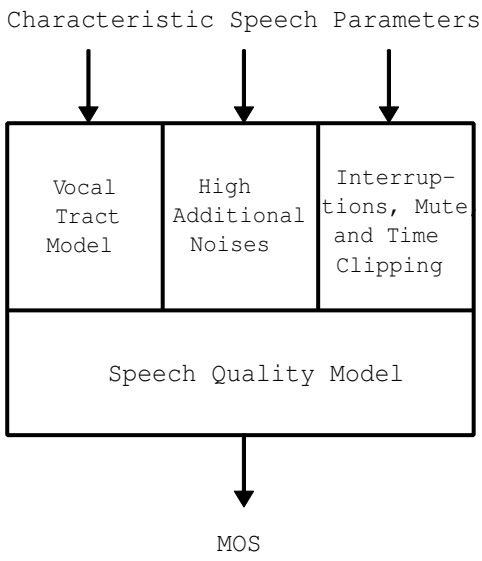


Figure 2.8: ITU-T P.563 functional blocks

blocks, as shown in Figure 2.8, corresponding to the main classes of distortion. These blocks are: vocal tract analysis, high additional noise, and speech interruptions, muting and time clippings. A total of 51 characteristic signal parameters are calculated. Based on a restricted set of 8 key parameters, a dominant distortion class is selected. The key parameters and the selected distortion class are used for adjusting the speech quality model. Furthermore, for each distortion class, a linear combination of parameters is used to generate an intermediate quality rating that, together with other additional signal features are combined to calculate the (raw) objective quality score. Unlike PESQ, the mapping from P.563's raw score to the 5-point ACR scale is achieved by means of a 3^{rd} order monotonic mapping function with coefficients optimized on a per-study basis.

Parameter-based models include the ITU-T E-model [15] and the recently proposed method by Sun [16]. The E-model combines the effect of various transmission parameters into a rating factor, which lies between 0 and 100. This rating factor

is later mapped to a MOS score. The E-model has a number of limitations and is applicable to only a restricted number of codecs and network conditions. In [16], a novel non-intrusive algorithm is proposed based on a combination of PESQ and the E-model. The method is reported to be generic and applicable to a number of multimedia applications. The reader is referred to [16] for a more thorough explanation of the system proposed by Sun.

2.4 Summary

This chapter has presented two methods of speech quality assessment: subjective and objective. Subjective speech quality assessment has been described and its importance has been highlighted. Due to the scope of this thesis, we focus mainly on objective speech quality measurement. State-of-art intrusive algorithms have been introduced and the scheme used to develop the work presented in this thesis (SDMQA) has been described in-depth. Non-intrusive methods are also concisely described.

Chapter 3

Gaussian Mixture Models

3.1 Introduction

Finite Mixture Distributions have been used as models throughout the history of statistics and estimation theory [33]. It has been over 100 years since Newcomb's application of Gaussian Mixture Models (henceforth GMMs) for outliers [34] and Pearson's classic paper on the decomposition of GMMs by the method of moments [35]. The ensuing century has revealed a multitude of fields in which GMMs are applied. Some applications include image compression [36]; speaker identification and pattern recognition [36–41]. GMMs have also been used as classifiers of features based on wavelet transforms for machine monitoring [42] and have found their way into supervised and unsupervised training and in the design of vector quantization [43–48]. Amazingly enough, GMMs have also found their way into finance and economics in the prediction of stocks and exchange rates [49]. This thesis proposes a novel use for GMMs: speech quality measurement [6–10].

In Section 3.2, GMM definition and properties are presented. Section 3.3 focuses

on GMM parameter estimation and the expectation-maximization algorithm is presented. Lastly, Section 3.4 describes GMM-based estimators. GMM-based estimators play an important role in the quality predictor algorithm proposed in this thesis.

3.2 Definition and Properties of GMMs

3.2.1 Definition

As stated previously, finite mixture models have served as important models throughout the history of statistics. The most important class of finite mixture models are the Gaussian mixtures, also known as Normal mixtures. The reasons behind this widespread use are not coincidental: (1) univariate Gaussian densities have a simple and concise representation, depending uniquely on two parameters, mean and variance, and (2) the Gaussian mixture distribution is universally studied and its behaviors are widely-known [41]. At a cost of extra parameters, GMMs improve on Gaussians by allowing asymmetry and multimodality.

Let $\mathbf{u}$ be an K-dimensional vector, a Gaussian mixture density is a weighted sum of M component densities

$$p(\mathbf{u}|\boldsymbol{\lambda}) = \sum_{i=1}^{M} \alpha_i . b_i(\mathbf{u}) \tag{3.1}$$

where $\alpha_i \geq 0, i = 1, ..., M$ are the mixture weights, with $\sum_{i=1}^{M} \alpha_i = 1$, and $b_i(\mathbf{u})$, $i = 1, ..., M$ are the K-variate Gaussian densities with mean vector $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_i$ given by

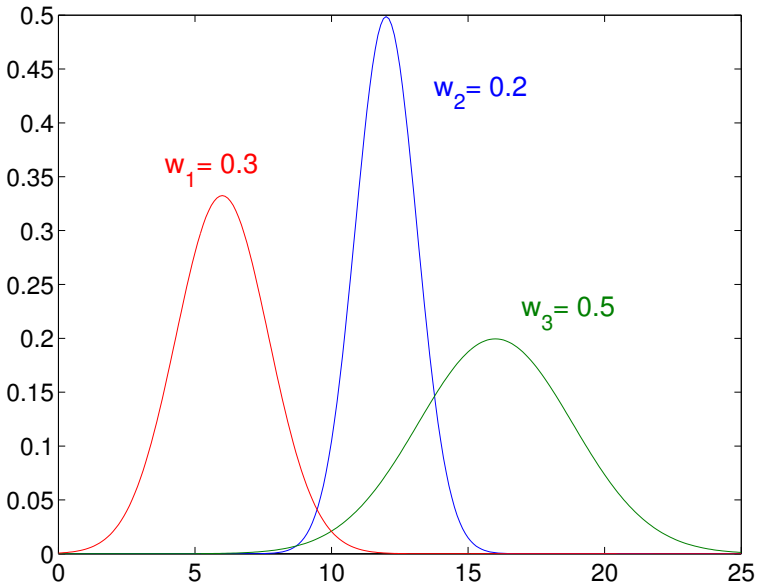
$$b_i(\mathbf{u}) = \frac{1}{(2\pi)^{K/2}|\boldsymbol{\Sigma}_i|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{u} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{u} - \boldsymbol{\mu}_i)\right) \tag{3.2}$$

where the superscript $[\cdot]^T$ indicates the transpose of a matrix or vector. The parameter list, $\boldsymbol{\lambda}$=$\{\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_M\}$, defines a particular Gaussian mixture density; each $\boldsymbol{\lambda}_i = \{\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i, \boldsymbol{\alpha}_i\}$ represents the three elements: mean vectors, covariance matrices and mixture weights of each Gaussian component. Figure 3.1 depicts a Gaussian mixture, consisting of three single Gaussians. Figure 3.1 (a) illustrates the three individual Gaussian distributions with their respective mixture weights. When combined as a GMM, the result is depicted in Figure 3.1 (b).

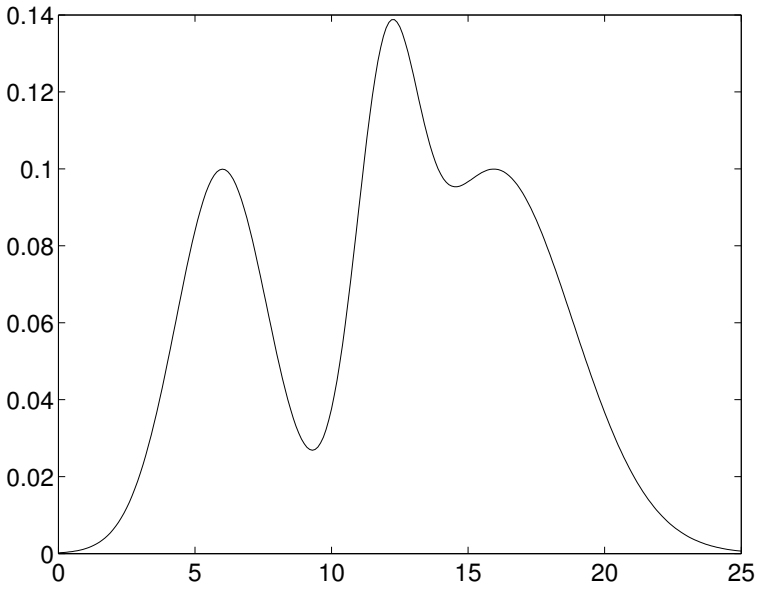
3.2.2 Properties

An important property of GMMs is that, by varying M and $\boldsymbol{\lambda}$, one can approximate any complex probability density function, to an arbitrary accuracy, given that the model uses a sufficiently large M and that the parameters are chosen wisely. In Figure 3.2 a 16-component GMM is used to fit a "spiral" distribution. Figures 3.2 (a) and (b) illustrate a "birds-eye-view" and a normal view of the spiral distribution and the fitted GMM contours, respectively. As more components are used, more accurate is the fit.

Furthermore, GMMs have the ability of forming smooth approximations of arbitrary densities. In [37], comparisons are performed between the classical unimodal Gaussian model, the Gaussian mixture model and the VQ (vector quantizer) model. The classical unimodal Gaussian model represents the feature distribution by a position (mean vector) and a spread (covariance matrix). The VQ model represents the feature vector by a discrete set of characteristic templates. The GMM, by using discrete sets of Gaussian functions, each with their own mean and variance, acts as a hybrid between these two models. An experiment consisting of modelling a single

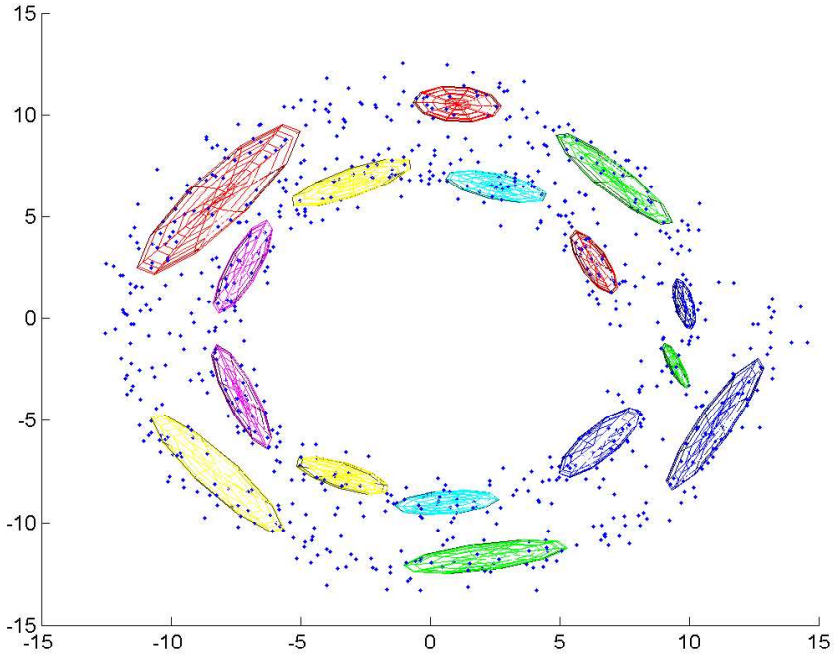


(a)

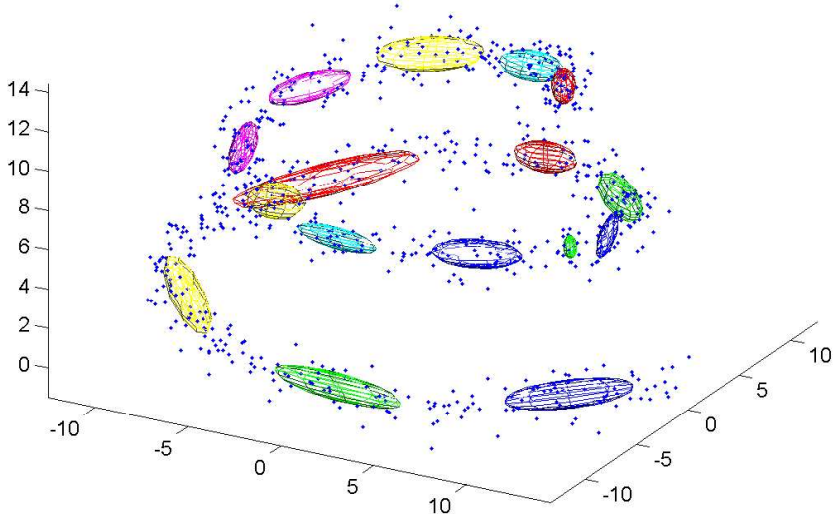


(b)

Figure 3.1: (a) three single Gaussians with their respective mixture weights, and (b) three single Gaussians combined as a GMM [50]



(a)



(b)

Figure 3.2: (a) birds-eye-view and (b) normal illustration of a spiral distribution and the fitted 16-component GMM contours

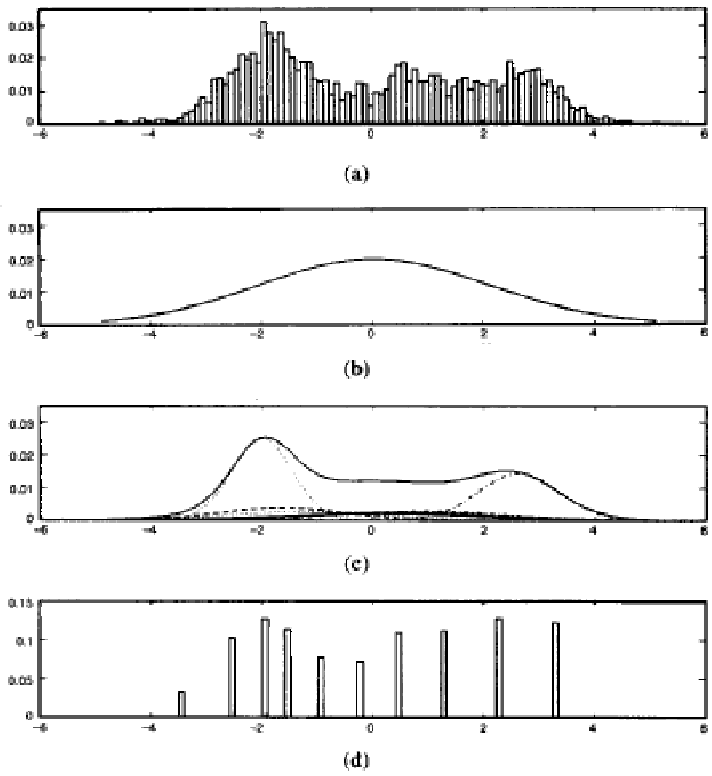


Figure 3.3: Comparison of distribution modelling: (a) histogram of a single cepstral coefficient from a 25 second utterance by a male speaker; (b) unimodal Gaussian model; (c) 10-component GMM; (d) histogram of the data assigned to the VQ centroid locations of a the 10-element codebook [37]

cepstral coefficient from a 25 second utterance by a male speaker is carried out in [37]. Figure 3.3, taken from [37], clearly illustrates the aforementioned fact that the GMM provides a more smooth overall distribution fit and its components detail the multimodal nature of the density.

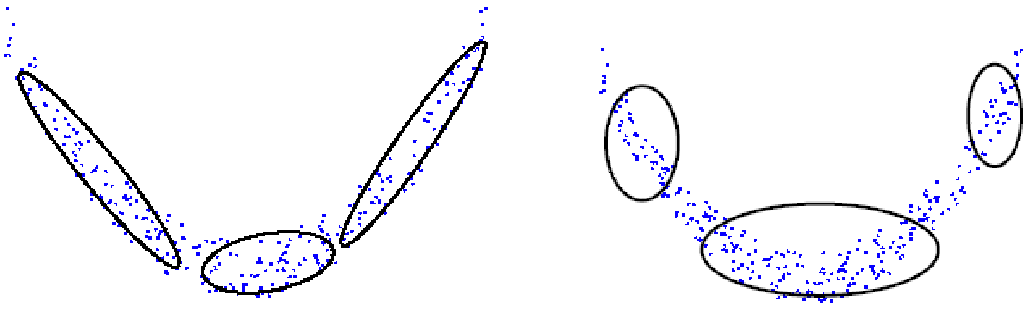
Due to the fact that GMMs use a discrete set of Gaussian functions, each with their own mean and variance, it is expected that the GMM have several different "shapes", depending on the form of their covariance matrices. The two most widely used forms

are full and diagonal covariance matrices. The full covariance matrix models the data as an ellipsoid-shaped cloud, at an angle, to allow correlation between feature components. This type is the most powerful Gaussian model, as it fits the data best. The drawback is the fact that it needs lots of data to properly estimate the parameters and often depends on regularization schemes to yield accurate estimates. For full covariance GMMs, the number of parameters that have to be estimated during training is given by $\frac{M}{2}(K^2+3K+2)$. Note that the optimal value for M is not known a priori. As will be later discussed in Chapter 4, the approach used in this thesis is to vary M over a range of values. The value that results in best performance on the validation data set, whilst maintaining a satisfactory training ratio, is chosen. [1]

On the other hand, diagonal covariance matrices are a good compromise between quality and model size. They model the data as ellipsoid-shaped clouds, aligned with the axes because correlations between feature components are ignored. This type is widely used in practice and mainly due to the fact that, since the Gaussian components are acting together to model the overall probability density function, a linear combination of diagonal covariance Gaussians is capable of modelling the correlations between the feature vector components. In summary, the effect of using M full covariance matrices can be equally obtained by using a larger set of diagonal covariance Gaussians [37]. For diagonal GMMs, $M(2K+1)$ parameters need to be estimated during training. Figure 3.4 illustrates the "clouds" that model the data. Note the clouds at an angle in (a) and the clouds aligned with the axes in (b).

When using GMMs to model data, estimates of $\boldsymbol{\lambda}$ have to be calculated efficiently in order to attain an accurate model. For full covariance GMMs, the number of parameters that need to be estimated scales quadratically with the feature space

[1] The reader is referred to Section 4.2 for a definition of training ratio



(a) (b)

Figure 3.4: Different forms of GMMs (a) full and (b) diagonal covariance matrices

dimension and due to the curse of dimensionality [51], these estimates may quickly become erroneous. The next section focuses on GMM parameter estimation based on the well-known expectation-maximization algorithm [52].

3.3 GMM Parameter Estimation

In statistics, a distinction is made between parametric estimators (those that make strong assumptions about the distribution of the sampled data) and non-parametric estimators (those that make weak distributional assumptions). The parametric approach assumes that the unknown *pdf* ($f(x)$) belongs to a family of parametric densities ($f(x|\theta)$). Once the specific functional form is chosen the problem reduces to finding the value of θ that best models the data. This technique falls in the field of maximum likelihood (ML) parameter estimation. The disadvantage is that an *a priori* structure of the distribution is enforced on the observed data and in most cases this structure is unknown.

The nonparametric approach is data-driven, i.e., it doesn't make any assumptions on the form of the unknown *pdf*. The advantage of this technique is that is gives consistent estimates irrespective of the original form of the unknown *pdf* [53]. These techniques, however, require substantially large numbers of observations to make their estimates and they do not allow easy learning. Some examples of nonparametric methods are the K-nearest-neighbor approach, the method of histograms and the kernel estimator, among others.

Both aforementioned approaches have their own merits and limitations. If a specific form is chosen for the density function, it may differ drastically from the true density and poor estimates are given. Using the nonparametric approach one can approximate any density function but the number of variables in the model grows directly with the number of data points [39]. Clearly an intermediate or semi-parametric approach would offer a more practical solution. Density estimation using mixture models falls into the class of semi-parametric estimation techniques, combining much of the flexibility and consistent estimates of nonparametric methods with certain analytic advantages of parametric methods [45]. Like a parametric model it has structure and parameters that control the behavior of the density in known ways, but without constraints that the data must be of a specific distribution type. Like a nonparametric model, the GMM has many degrees of freedom to allow arbitrary density modelling, without undue computation and storage demands [54].

When using GMMs, the goal in training is to estimate the parameters $\boldsymbol{\lambda}$ for a given training data set. There are several techniques available for estimating the parameters of a GMM but by far the most used and well-established method is the maximum likelihood estimation. The aim in ML estimation is to find the model parameters

that maximize the likelihood (or log likelihood) of the GMM, given the training data. Unfortunately, the expression for the GMM likelihood is a nonlinear function of the parameters, as will be shown in the sequel, and direct maximization is not possible. The use of the expectation-maximization (EM) algorithm [52] is required in order to obtain, iteratively, the ML parameter estimates. The EM algorithm is discussed next.

3.3.1 The Expectation-Maximization (EM) Algorithm

As was stated before, the aim in ML estimation is to find the model parameters that maximize the likelihood of the GMM, given the training data. Assuming independence between the T training vectors, $\mathbf{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_T\}$, the GMM likelihood can be written as

$$p(\mathbf{U}|\boldsymbol{\lambda}) = \prod_{i=1}^{T} p(\mathbf{u}_i|\boldsymbol{\lambda}) \tag{3.3}$$

where $p(\mathbf{u}_i|\boldsymbol{\lambda})$ is given by Equation 3.1.

Clearly this expression is a nonlinear function of the parameters $\boldsymbol{\lambda}$ and a direct maximization cannot be found. The log likelihood of the parameters, given the data set becomes

$$l(\boldsymbol{\lambda}|\mathbf{U}) = \sum_{i=1}^{T} \log p(\mathbf{u}_i|\boldsymbol{\lambda}) = \sum_{i=1}^{T} \log \sum_{j=1}^{M} \alpha_j b_j(\mathbf{u}_i). \tag{3.4}$$

This function is not easily maximized because it involves the log of a sum.

Intuitively, there is a "credit-assignment" [45] problem: which component of the mixture generated a given data point and thus which parameters to adjust to fit the data point? The EM algorithm for GMMs is an iterative method that tries to solve this "credit-assignment" problem [52]. A mathematical trick is often used and consists of assuming a "hidden" binary indicator variable $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_T\}$, such that

$\mathbf{z}_i = (z_{i1}, \ldots, z_{iM})$ and $z_{ij} = 1$ if and only if data point $\mathbf{u}_i$ was generated by Gaussian component j. With this, the maximization problem decouples into a set of simple maximizations, and the log likelihood function becomes [45]

$$l_c(\boldsymbol{\lambda}|\mathbf{U}, \mathbf{Z}) = \sum_{i=1}^{T} \sum_{j=1}^{M} z_{ij} \log[p(\mathbf{u}_i|\mathbf{z}_i; \boldsymbol{\lambda}) p(\mathbf{z}_i; \boldsymbol{\lambda})]. \tag{3.5}$$

Since $\mathbf{z}_i$ is unknown l_c cannot be utilized directly; the expectation of l_c, here denoted as $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(k)})$, is used instead. $Q(\cdot)$ is the so called Q function [52] and $\boldsymbol{\lambda}^{(k)}$ are the estimated parameter values after k iterations. The EM algorithm maximizes the likelihood function by iterating the following two steps, the E-step and the M-step.

1. *E Step:* $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(k)}) = E_Z[l_c(\boldsymbol{\lambda}|\mathbf{U}, \mathbf{Z})|\mathbf{U}, \boldsymbol{\lambda}^{(k)}]$
2. *M-Step:* $\boldsymbol{\lambda}^{(k+1)} = \arg\max_{\boldsymbol{\lambda}} Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(k)})$

The E-Step computes the expected data log likelihood and the M-Step finds the parameters that maximizes this likelihood. These two steps form the basis of the EM algorithm for mixture modelling. Each EM iteration guarantees a monotonic increase in the model's likelihood (log likelihood) value. Furthermore, in [52, 55] the EM algorithm is shown to always converge, given that measures are taken to avert ill-conditioning.

3.3.2 EM algorithm for GMMs

For GMMs, the E-Step simplifies to computing $h_{ij} \equiv E[z_{ij}|\mathbf{u}_i, \boldsymbol{\lambda}^{(k)}]$, i.e., the probability that Gaussian component j with parameters defined by iteration k generated the data point $\mathbf{u}_i$, given by

$$h_{ij} = \frac{|\hat{\boldsymbol{\Sigma}}_j^{(k)}|^{-1/2} \exp\left(-\frac{1}{2}(\mathbf{u}_i - \hat{\boldsymbol{\mu}}_j^{(k)})^T \hat{\boldsymbol{\Sigma}}_j^{(k)^{-1}} (\mathbf{u}_i - \hat{\boldsymbol{\mu}}_j^{(k)})\right)}{\sum_{l=1}^{M} |\hat{\boldsymbol{\Sigma}}_l^{(k)}|^{-1/2} \exp\left(-\frac{1}{2}(\mathbf{u}_i - \hat{\boldsymbol{\mu}}_l^{(k)})^T \hat{\boldsymbol{\Sigma}}_l^{(k)^{-1}} (\mathbf{u}_i - \hat{\boldsymbol{\mu}}_l^{(k)})\right)}. \tag{3.6}$$

The M-Step re-estimates the mixing proportions, means and covariances of the Gaussians using the data weighted by h_{ij}. The new estimates (at iteration $k+1$) become

$$\hat{\boldsymbol{\alpha}}_j^{(k+1)} = \frac{1}{T} \sum_{i=1}^{T} h_{ij}, \tag{3.7}$$

$$\hat{\boldsymbol{\mu}}_j^{(k+1)} = \frac{\sum_{i=1}^{T} h_{ij} \mathbf{u}_i}{\sum_{i=1}^{T} h_{ij}}, \tag{3.8}$$

$$\hat{\boldsymbol{\Sigma}}_j^{(k+1)} = \frac{\sum_{i=1}^{T} h_{ij} (\mathbf{u}_i - \hat{\boldsymbol{\mu}}_j^{(k+1)})(\mathbf{u}_i - \hat{\boldsymbol{\mu}}_j^{(k+1)})^T}{\sum_{i=1}^{T} h_{ij}}. \tag{3.9}$$

Note that the training of a GMM requires the selection of two initial and critical factors: the order M of the mixture model and the initial parameters of the algorithm. A few strategies have been taken into consideration to alleviate the dependence on initialization, e.g., multiple random starts and choosing the one leading to the highest likelihood, and modified EM algorithms using split and merge operations to escape from local maxima [44]. The most common approach, however, is to use the *k-means* algorithm [56]. The algorithm partitions the data into M subsets, each subset populating a region in the feature space. The empirical probability of each subset becomes the initial mixture weights. The mean of the data in each subset becomes

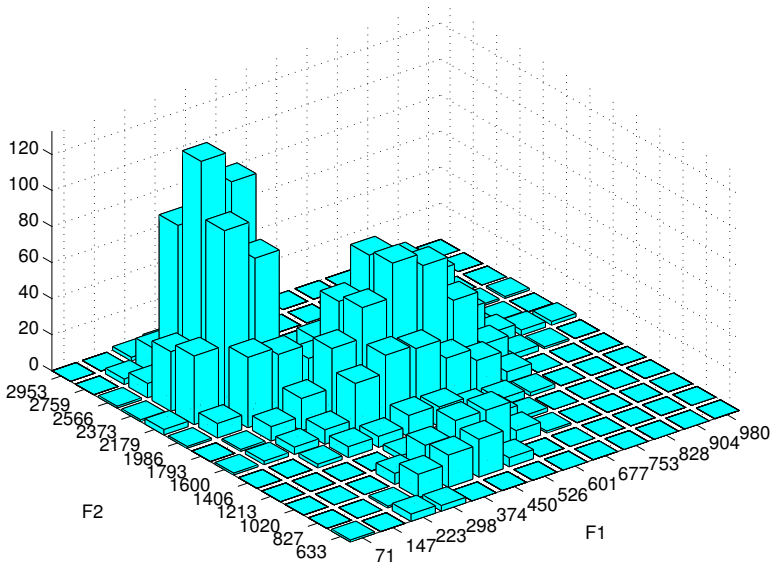


Figure 3.5: Histogram of pairs of the first two formant frequencies for vowels /a/ and /o/

the initial mean of the corresponding mixture component and the covariance of the data of each subset determines the initial covariance of the respective component.

The following experiment illustrates the effectiveness of the EM algorithm on finding the parameters of a GMM. Figure 3.5 depicts a 2-D histogram corresponding to simulated artificial pairs of the first two formant frequencies for the vowels /a/ and /o/. Figures 3.6–3.8 illustrate three GM models (with $M = 3$) estimated at three different EM iterations. Figure 3.6 depicts a GMM with initial parameters estimated via the *k-means* algorithm. Figures 3.7 and 3.8 exhibit the GM model with parameters estimated at iteration 1 and 18, respectively. Convergence is shown to have been achieved by the 18^{th} iteration, as shown in Figure 3.9. Visual comparisons between Figures 3.5 and 3.8 display the accuracy of the GM model, fitted to the data by means of the EM algorithm.

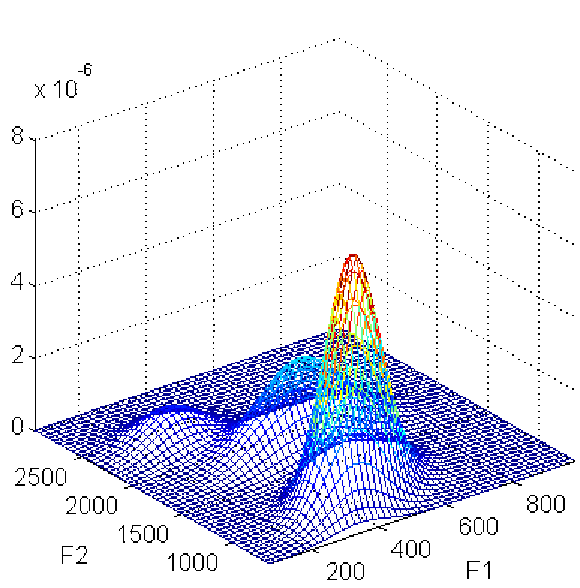


Figure 3.6: GM model with initial parameters found by *k-means* algorithm

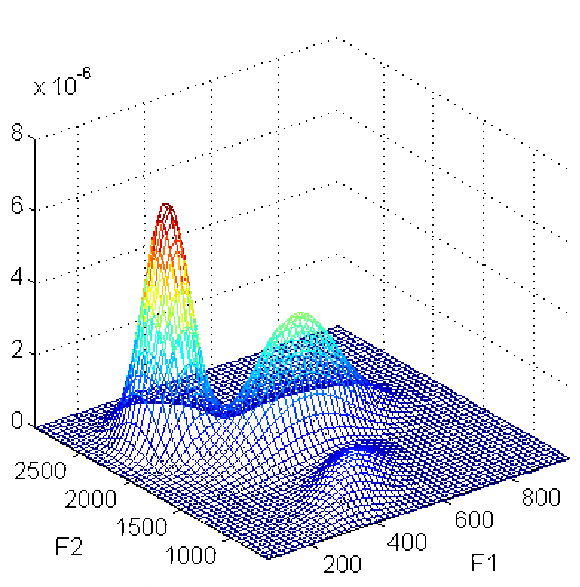


Figure 3.7: GM model after one EM iteration

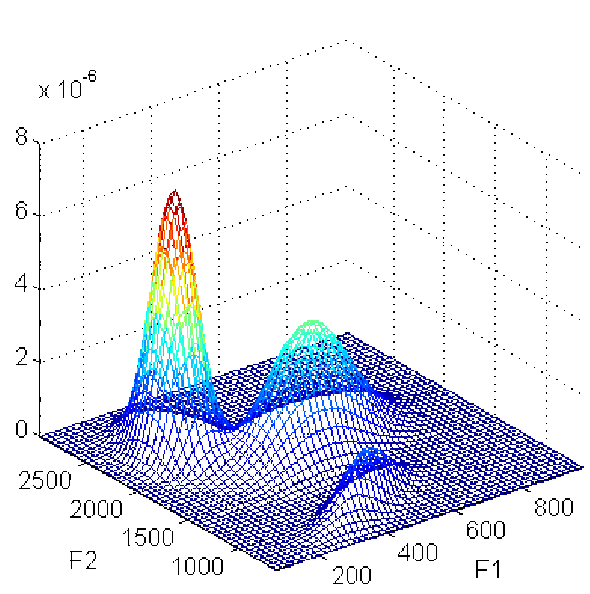


Figure 3.8: GM model after 18 EM iterations

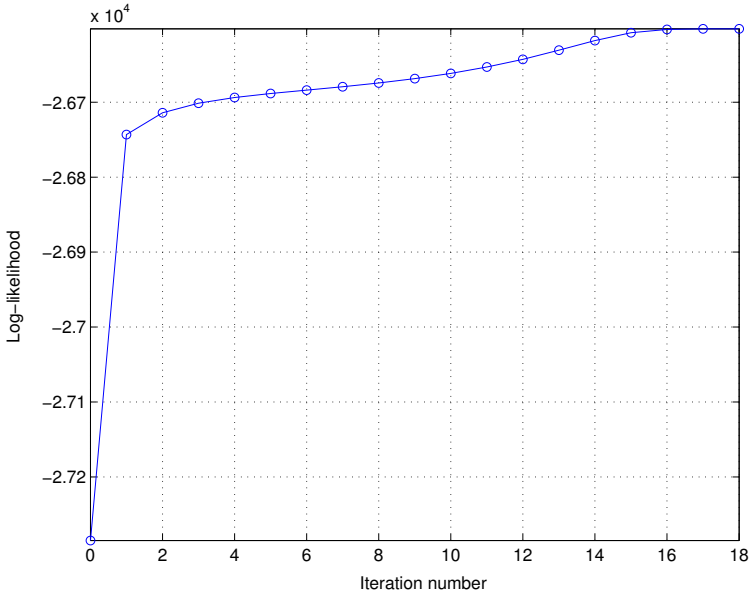


Figure 3.9: Log-likelihood *versus* iteration number; convergence is achieved by the 18^{th} iteration

It is also important to note that with full covariance matrices the number of parameters that need to be estimated scales quadratically with the feature space dimension. When dealing with limited data, severe problems may arise due to singularities. Many regularization schemes have been proposed to avert ill-conditioning. The modified EM algorithm with Tikhonov regularization, proposed by Koshizen *et al*, is an example [57]. Ormoneit and Trest propose in [48] two regularization methods: the first uses a Bayesian prior on the parameter space, the second applies ensemble averaging to density estimation. In [47], a method of pruning the eigen-directions of each covariance matrix is proposed. In [58, 59] constraints are introduced in the covariance matrices enabling stable EM algorithms and deterministic annealing.

Another simple, yet effective approach to avert ill-conditioning is to add a small diagonal matrix, namely $\epsilon I_{n\times n}$, to each covariance matrix in each M-step iteration of the EM algorithm. Typically, the optimal value for ϵ is not known a priori. It is common to vary ϵ over a range of values and choose the value that leads to the best performance on the validation data set. This last approach is the one used throughout the schemes proposed in this thesis.

In [45], an extension to the EM algorithm is given and allows for the incorporation of missing values in the learning phase. [47] builds on the idea and devises estimators based on GMMs. In [40], GMM estimators are used to adjust the magnitude spectrum of a speech signal when the fundamental frequency of the signal is altered. In [60], GMMs are used to estimate missing line spectral frequencies. GMM-based estimators are the heart of this thesis and are presented next.

3.4 GMM-Based Estimators

The goal in GMM-based minimum mean squared error (MMSE) estimation is to find a mapping or regression function, $\hat{f}(\mathbf{x})$, that minimizes the mean squared error, ε_{MSE}, between predictor variables ($\mathbf{x}$) and the target variable (y), where

$$\varepsilon_{MSE} = E[(y - \hat{f}(\mathbf{x}))^2]. \tag{3.10}$$

It is known that the mean squared error (3.10) is minimized when $\hat{f}(\mathbf{x}) = E[y|\mathbf{x}]$, the conditional expectation of the target variable, given the predictor vector.

GMM-based estimators rely on modelling the joint density of the K-dimensional predictor variables with the target variable using (3.1) with $\mathbf{u} = [y, \mathbf{x}]^T$. The mean vector and the covariance matrix of the i^{th} GMM component become, respectively

$$\boldsymbol{\mu}_i = (\mu_i^y \ \boldsymbol{\mu}_i^x),$$

$$\boldsymbol{\Sigma}_i = \begin{pmatrix} \Sigma_i^{yy} & \boldsymbol{\Sigma}_i^{yx} \\ \boldsymbol{\Sigma}_i^{xy} & \boldsymbol{\Sigma}_i^{xx} \end{pmatrix}.$$

Given the GMM parameters, the MMSE regression function is given by [47]

$$\hat{f}(\mathbf{x}) = E[y|\mathbf{x}] = \sum_{i=1}^{M} h_i(\mathbf{x})[\mu_i^y + \boldsymbol{\Sigma}_i^{yx}(\boldsymbol{\Sigma}_i^{xx})^{-1}(\mathbf{x} - \boldsymbol{\mu}_j^x)]. \tag{3.11}$$

The above GMM estimator or GMM regressor function is a weighted sum of linear models, where the weight $h_i(\mathbf{x})$ is the probability that the i^{th} Gaussian component generated the vector $\mathbf{x}$ and given by

$$h_i(\mathbf{x}) = \frac{\frac{\alpha_i}{|\boldsymbol{\Sigma}_i^{xx}|^{1/2}} \ \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i^x)^T(\boldsymbol{\Sigma}_i^{xx})^{-1}(\mathbf{x} - \boldsymbol{\mu}_i^x)\right)}{\sum_{k=1}^{M} \frac{\alpha_k}{|\boldsymbol{\Sigma}_k^{xx}|^{1/2}} \ \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k^x)^T(\boldsymbol{\Sigma}_k^{xx})^{-1}(\mathbf{x} - \boldsymbol{\mu}_k^x)\right)}. \tag{3.12}$$

Note the similarities of (3.12) with (3.6). Moreover, if the covariance matrices are restricted to be diagonal, (3.11) simplifies to

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{M} h_i(\mathbf{x})\mu_i^y. \tag{3.13}$$

This restriction has to be used with care, as it can result in large estimation errors when there exists a significant amount of correlation between the predictor and the response variables, i.e., $\mathbf{\Sigma}_i^{yx}$ are far from zero.

3.5 Summary

This chapter has presented Gaussian mixture models, their properties and applications. The two most widely used forms of GM models have been addressed: diagonal and full covariance matrix GMMs. The expectation-maximization (EM) algorithm, a stable algorithm for estimating GMM parameters has also been introduced. The effectiveness of the EM algorithm is illustrated by means of an example where simulated speech formant pairs are accurately modelled by a 3-component GMM in less than twenty iterations of the EM algorithm. Lastly, the concept of GMM-based estimators are described. GMM-based estimators are the foundation of the speech quality estimation algorithm proposed in this thesis. The proposed algorithm is presented in the next chapter.

Chapter 4

Towards a Novel Speech Quality Estimator

4.1 Introduction

In this chapter the use of GMM estimators to predict the quality of a speech signal is proposed. First, SDMQA is used to generate a large pool of feature measurements from the distortion surface between the original speech signal and the degraded speech signal. Four statistical data mining methods, multivariate adaptive regression splines (MARS) [30], classification and regression trees (CART) [29], a hybrid CART-MARS technique, and the sequential forward selection (SFS) algorithm [61] are used to sift out salient features. Gaussian mixture densities are used to model the joint distribution of these features ($\mathbf{x}$) with the subjective MOS (y). The MMSE estimate, $E[y|\mathbf{x}]$, of the subjective MOS value is derived by means of (3.11). We use PESQ to benchmark the proposed algorithm.

The remainder of this chapter is organized as follows. Section 4.2 will describe the

architecture of the proposed algorithm. A speech quality estimation task is carried out and performance results for the proposed algorithm are provided in Section 4.3.

4.2 Algorithm Description

The proposed intrusive measurement algorithm is designed based on the architecture depicted in Figure 4.1. The algorithm is built on perceptual feature variables, obtained from mining a large pool of candidate feature variables extracted from the SDMQA algorithm. 90% of the available data is used for training of the GMMs, whilst 10% is kept for testing.

A brief review of the feature extraction block (SDMQA) in Figure 4.1 is as follows. First, the clean and degraded signals are split into 7 frequency bands. The spectral power distortion between the clean and degraded speech signals is then found. Time segmentation labels the speech frames as "active" or "inactive". Active frames are further classified into voiced or unvoiced. The total distortion of each frame is given severity classifications of "low", "medium", or "high" by simple thresholding. Distortion samples in time-frequency bins are thus labelled according to its frequency band, time-segmentation type, and severity level. Additional contexts are created to form a pool of 209 candidate features.

A pool of 209 candidate features is redundant for the quality estimation task at hand. A brute force approach to finding the best subset of features to use would entail examining $2^{209} - 1$ possible subsets, a clearly impossible task. Nonetheless, the initial steps towards proposing a novel speech quality assessment algorithm requires a smaller subset of features that combined have the power of effectively predicting a speech signal's quality. Data mining techniques are used to sift out the most relevant

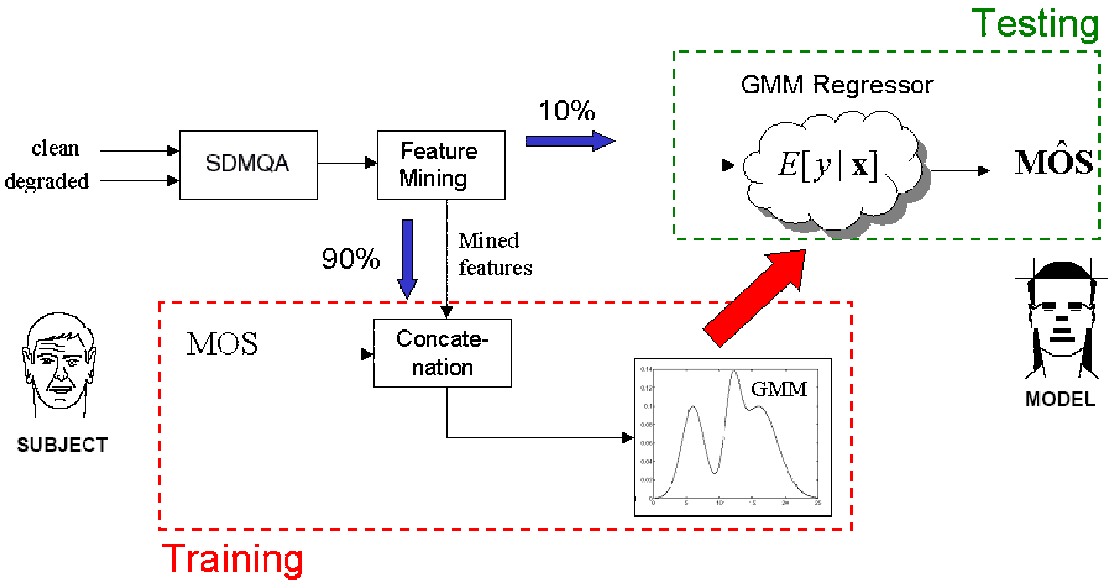


Figure 4.1: Architecture of proposed algorithm

variables from the pool of variables. A brief description of two common data mining tools – CART and MARS – is given in the sequel.

CART

CART (classification and regression trees) is a binary recursive partitioning algorithm. The process is binary because parent nodes are always split into two child nodes by answering a simple "yes" or "no" question on a predictor variable. It is recursive in the sense that the process can be repeated by treating each child node as a parent node. A notion of variable importance is introduced in CART by means of a purity function. A split is selected such that the data in the child node is "purer" than the data in the parent node. A node is recursively split until a decrease in the impurity function reaches a certain threshold.

CART trees are designed in a two-stage process. First, an over-size tree is grown. The tree is then pruned based on performance validation, until the best-size tree is found. At this point, feature importance rankings are determined by summing the decrease in impurity produced in the remaining nodes if the split were attained at that specific feature. Scores reflect the contribution each predictor variable has on estimating the target variable. The feature used to split the root node receives 100 % importance, while features that receive 0% importance play no role in estimation and are discarded.

MARS

Multivariate adaptive regression splines (MARS) are constructed as a sum of basis functions, or more specifically, truncated spline functions. Like CART, the MARS

Table 4.1: Training ratio as a function of M

M	Diagonal	Full
2	68	31
3	45	21
4	34	16
5	27	13

regression model is also built in two stages. First, an over-size model is built by progressively adding more basis functions. In the second stage, basis functions that contribute the least to modelling accuracy are progressively pruned. With MARS, variable importance scores are found by investigating the effects the variable has in fitting the data by dropping it from the model. The most important variable is the one that, when omitted, degrades the model fit the most. As CART, feature variables receive an importance score ranging from 0% to 100%. Features that receive a 0% importance rating are discarded.

It is noted that both CART and MARS assign an importance rating greater than 0% to 20 features (reader is referred to [5] for a description of the 20 features). Training a GMM on 20 features is costly, especially if full covariance matrices are used. Experiments show that by choosing the top-5 most important feature variables, a good compromise between accuracy and complexity is achieved. Table 4.1 shows the training ratio (ratio between the number of parameters that have to be estimated during the training phase and the total number of files in the training set) as a function of M. Results are presented for both diagonal and full covariance GMMs. Note that the training ratio for full covariance GMMs achieves low levels for $M > 3$. Same is true for diagonal GMMs when $M > 5$.

Initially, the top-5 features as ranked by MARS, CART or a CART-MARS hybrid configuration are tested in the design of the proposed GMM-based speech quality estimator. During the training phase, the joint density of the top-5 most important feature variables ($\mathbf{x}$) with the subjective MOS (y) is modelled as a GMM, as in (3.1) with $\mathbf{u} = [y, \mathbf{x}]^T$. The GMM parameters $\boldsymbol{\lambda}$ found in training are used in the testing phase to predict the value of the subjective MOS, y, given the observed values of the 5-dimensional feature vector, $\mathbf{x}$. The MMSE estimate of y given $\mathbf{x}$, namely $E[y|\mathbf{x}]$, is given by (3.11). In the next section, the proposed algorithm is tested and compared to PESQ.

4.3 Speech Quality Prediction Test

The proposed algorithm is compared to PESQ using MOS labelled speech databases. The performance of the algorithms is assessed using the correlation (R) and root-mean-square error ($RMSE$) between subjective MOS w_i and objective MOS y_i. Correlation coefficients are calculated using Pearson's formula

$$R = \frac{\sum_{i=1}^{N}(w_i - \bar{w})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(w_i - \bar{w})^2 \sum_{i=1}^{N}(y_i - \bar{y})^2}} \tag{4.1}$$

where $\bar{w}$ is the average of w_i, and $\bar{y}$ is the average of y_i. $RMSE$ is calculated using

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(w_i - y_i)^2}{N}}. \tag{4.2}$$

Note that in [62] $RMSE$ is shown to be the sum of unexplained variance in the regression model, MOS estimation error due to limited number of listeners (affecting all algorithms equally), and bias error between subjective MOS and objective MOS. The calculation of R does not take into consideration this bias error; therefore, unless

Table 4.2: Description of speech databases used in this thesis

#	Database	Language	no. of files
1	ITU-T Supp23 Exp1A	French	176
2	ITU-T Supp23 Exp1D	Japanese	176
3	ITU-T Supp23 Exp1O	English	176
4	ITU-T Supp23 Exp3A	French	200
5	ITU-T Supp23 Exp3C	Italian	200
6	ITU-T Supp23 Exp3D	Japanese	200
7	ITU-T Supp23 Exp3O	English	200
8	Wireless IS-127 EVRC	English	96
9	Wireless IS-96A	English	96
10	Mixed	English	240
11	G.728	Japanese	1068
12	G.728	English	1068
13	G.728	Italian	1068

the estimates are unbiased or all suffer from the same bias, $RMSE$ can be viewed as a more realistic measure of estimator performance.

The speech databases used in this thesis are listed in Table 4.2. They include the 7 multilingual databases in ITU-T P-series Supplement 23 [63], two wireless databases (IS-96A and IS-127 EVRC), a mixed wireline-wireless database, and three multilingual databases comprised of speech coded using the ITU-T G.728 speech coder.

The three Exp1x databases in ITU-T Supp23 contain speech coded using the G.729 codec, singly or in tandem with one or two other wireline or wireless standard codecs, under clean channel condition. The four Exp3x databases contain single- and multiple-encoded G.729 speech under various channel error conditions and input noise conditions. The wireless IS-96A and IS-127 EVRC (Enhanced Variable Rate Codec) databases contain speech coded using the IS-96A and IS-127 codec, respectively, under various clean and degraded channel conditions.

The mixed database contains speech coded with a variety of wireline and wireless codecs, under a wide range of degradation conditions: tandeming, channel errors, and clipping. The three G.728 databases contain speech coded using the G.728 (16kbps) speech coder subjected to various channel errors, tandeming, and acoustic noise. All databases include reference conditions such as speech degraded by various levels of MNRU (modulated noise reference unit).

The speech databases used in this experiment are databases numbered 1-10. These ten databases are combined into a global database with a total of 1760 speech file pairs and 10-fold cross validation is used to measure performance. The global database is randomly divided into 10 data sets of almost equal size. Training and testing is performed in 10 trials, where, in each trial, one of the data sets serves as a test set and the remaining 9 are combined to serve as a training set. Each data set serves as a test set only once. The ten resulting R's and $RMSE$'s are averaged to obtain the cross-validation R and $RMSE$.

The performance results for the feature variables selected by MARS and CART are shown in Tables 4.3 and 4.4, respectively. GMM-i stands for a Gaussian mixture model with i components. The column "%↑" indicates percentage improvement in R found by using the proposed method. The percentage improvement is given by

$$\% \uparrow R = \frac{R_{GMM} - R_{PESQ}}{1 - R_{PESQ}} \times 100\%. \tag{4.3}$$

The improvement indicates percentage reduction of the gap to perfect correlation. In turn, column "%↓" indicates percentage reduction in $RMSE$ relative to PESQ.

As can be seen, when using 5-component diagonal GMMs, a 13.67% decrease in $RMSE$ can be achieved for MARS selected features. For CART selected features

Table 4.3: Performance for MARS selected variables - diagonal covariance matrices

	R	%↑	$RMSE$	%↓
PESQ	0.8185	N/A	0.460	N/A
GMM-3	0.8086	- 5.45	0.4094	11.01
GMM-4	0.8232	2.58	0.4008	12.86
GMM-5	0.8377	10.58	0.3971	13.67

Table 4.4: Performance for CART selected variables - diagonal covariance matrices

	R	%↑	$RMSE$	%↓
PESQ	0.8185	N/A	0.460	N/A
GMM-3	0.8315	7.16	0.4035	12.27
GMM-4	0.8395	11.57	0.3957	13.97
GMM-5	0.8531	19.06	0.3938	14.38

a 14.38% decrease can be attained. The penalty of using diagonal matrices can be illustrated with the graph of the subjective MOS *versus* objective MOS for one of the cross-validation trials (vide Figure 4.2). The prominent vertical alignment of points suggests poor prediction performance. The alignment disappears and better prediction performance is obtained when full covariance matrices are used, as will be shown in the sequel.

With full covariance matrices the number of parameters that need to be estimated scales quadratically with the feature space dimension. To avert ill-conditioning a small diagonal matrix, namely $\epsilon I_{n\times n}$, is added to each covariance matrix in each M-step iteration of the EM algorithm. In this experiment, the ϵ that led to best performance was $\epsilon = 0.001$.

The performance results for full covariance GMMs are shown in Tables 4.5 and 4.6, for feature variables selected by CART and MARS, respectively. With the correlation between features properly modelled, an improvement of 23.47% and 16.09% in R

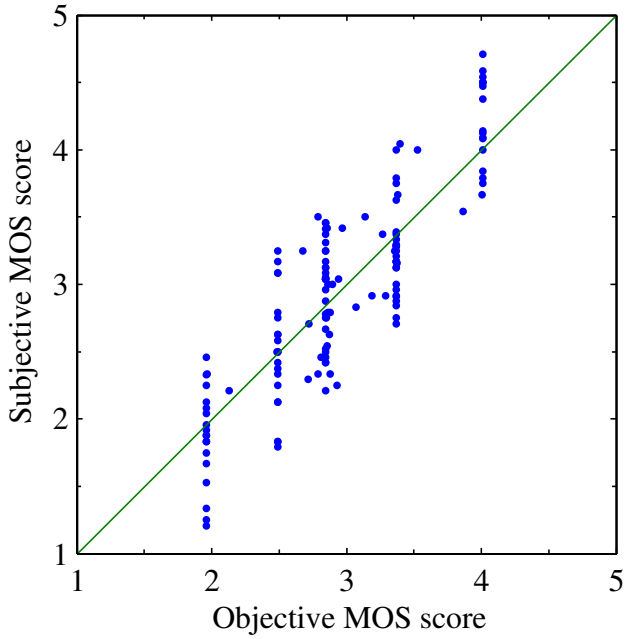


Figure 4.2: Subjective MOS *versus* objective MOS for CART selected features using five diagonal Gaussian components.

Table 4.5: Performance for CART selected variables - full covariance matrices

	R	%↑	$RMSE$	%↓
PESQ	0.8185	N/A	0.460	N/A
GMM-2	0.8569	21.15	0.3892	15.39
GMM-3	0.8611	23.47	0.3860	16.09

and $RMSE$, respectively, is achieved for CART selected features and a 3-component GMM. Further improvement can be seen for MARS selected features where an improvement of 32.78% and 17.98% in R and $RMSE$ is achieved.

It is conjectured that for diagonal covariance GMMs, the improvement in R is more modest because some of the features selected by MARS and CART have significant correlation amongst them. This is illustrated with the use of the correlation color map in Figures 4.3 and 4.4. Figure 4.3 represents the correlation between the predictor

Table 4.6: Performance for MARS selected variables - full covariance matrices

	R	%↑	$RMSE$	%↓
PESQ	0.8185	N/A	0.460	N/A
GMM-2	0.8683	27.43	0.3773	17.52
GMM-3	0.8780	32.78	0.3783	17.98

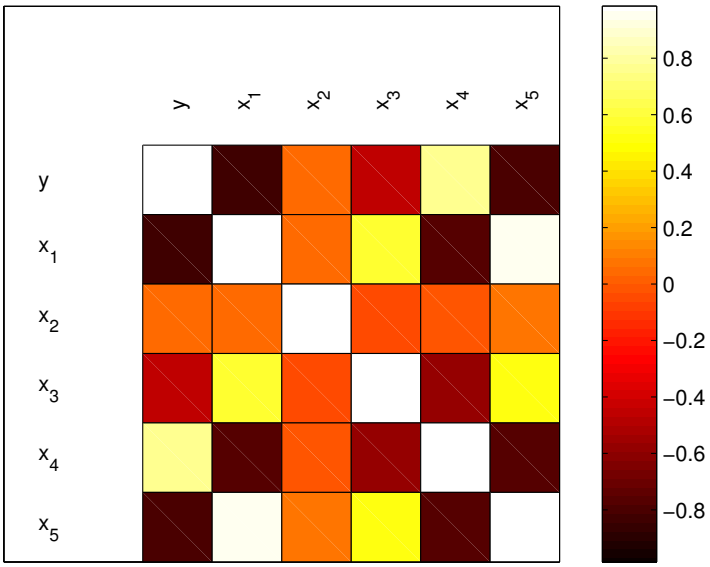


Figure 4.3: Correlation map for MARS selected features.

(y) and the features (x_1 to x_5) selected by MARS. For CART, the color map is shown in Figure 4.4. The use of a small number of diagonal Gaussian components does not compensate for this correlation and full covariance matrices are thus needed in order to predict the residual variation in subjective MOS.

The 5 most salient feature variables are listed in Table 4.7 for all data mining techniques. As can be seen, only voiced frames are captured by the features selected using CART. In whispered speech, all normally voiced phonemes are not vocalized, i.e.

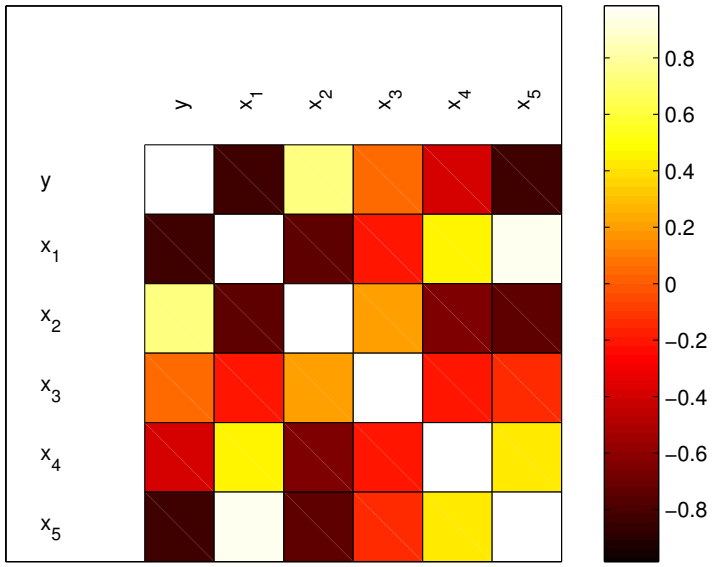


Figure 4.4: Correlation map for CART selected features.

they become unvoiced. Such situation, though rare, would cause problems for these features. It can be inferred that the estimator is not sensitive to degradation in the unvoiced regions and the non-speech regions. The CART-MARS hybrid scheme uses CART to pre-screen features from the feature candidate pool. The features selected by CART are then used as a smaller feature candidate pool for MARS to sort through. By doing this, features can be drawn from voiced and unvoiced frames. An improvement of 30.24% and 19.04% in R and $RMSE$ is achieved with a 3-component GMM. The performance results for the feature variables selected by this hybrid scheme are shown in Table 4.8.

Careful analysis of the features selected by the three aforementioned data mining schemes showed a certain trend within the feature variables as shown in Table 4.9. The table is composed of the subjective MOS, five MARS-selected features ($x1$ to

Table 4.7: Features selected for various data mining algorithms

Rank	MARS	CART	CART-MARS	SFS
1	I_P_VUV	V_WM	I_P_VUV	V_O_1
2	V_B_5	V_O_2	REF_1	I_P_VUV
3	V_B_2	V_O_1	V_WM_2	REF_1
4	V_B_2_2	V_RM	V_B_5	V_B_2
5	U_P_VUV	V_O_0	U_P	V_O_5

Table 4.8: Performance for CART-MARS selected variables - full covariance matrices

	R	$\%\uparrow$	$RMSE$	$\%\downarrow$
PESQ	0.8185	N/A	0.460	N/A
GMM-2	0.8662	26.28	0.3766	18.13
GMM-3	0.8734	30.24	0.3724	19.04

$x5$), and the objective MOS ($\hat{MOS}$) estimated using a 3-component full GMM, for six distinct test speech signals. Note that all six test signals have the same subjective MOS, i.e., they have been rated as having, on average, the same objectionable level of distortion. It can be inferred that these six speech signals belong to the same "distortion class" and their feature values should not vary considerably.

Table 4.9 shows that the features selected by MARS, on the contrary, vary considerably and this variation is reflected on the estimates. For test vector 2 an estimate of 3.598 is obtained, i.e., an error of 2.8%, but for test vector 6 the estimate MOS is 2.308, an error of 34%. It is conjectured that in order to obtain further improvement better features would have to be used, preferably features that do not vary considerably within the same distortion class. To this end, the SFS algorithm is used. The algorithm starts with the variable that is most correlated with the target variable, and at each step adds a new variable that, together with the previous ones, most

Table 4.9: Trends in MARS selected features

File #	MOS	$\hat{MOS}$	$x1$	$x2$	$x3$	$x4$	$x5$
1	3.5	3.843	0.312	0.421	0.662	0.517	1.072
2	3.5	3.598	0.328	0.471	0.703	0.521	1.011
3	3.5	3.037	0.666	0.885	1.194	0.918	1.680
4	3.5	2.309	0.766	1.091	1.538	1.169	2.226
5	3.5	3.038	0.334	0.504	0.895	0.753	1.691
6	3.5	2.308	0.902	1.101	1.448	1.222	2.402

accurately predicts the target. A linear regression mapping is used by the SFS algorithm. Furthermore, a partial F-test is incorporated in the algorithm such that the variables chosen have small variances within each distortion class.

An improvement of 36.08% and 21.7% in R and $RMSE$ is provided by the SFS algorithm. The performance results of this scheme are shown in Table 4.10. Looking back at Table 4.7, it can be seen that the features selected by the SFS algorithm are gleaned from the top three features selected by MARS, CART, and the CART-MARS hybrid scheme.

Figure 4.5 depicts a scatter plot of the subjective MOS *versus* objective MOS for CART-MARS selected features, using a GMM with three Gaussian components and full covariance matrices. The data shown are for one of the cross-validation trials. Note that the points are no longer aligned with the vertical axis as in the case of diagonal covariance matrices. A further method for measuring model performance is to plot the distribution of absolute residual errors between objective and subjective MOS [26]. Figure 4.6 plots the distribution of errors for SFS selected features for one of the trials. As can be seen, almost 78% of the GMM estimates are within 0.50 unit of the subjective MOS.

A last experiment compares GMM estimators to multiple linear regressors. GMMs

Table 4.10: Performance for SFS selected variables - full covariance matrices

	R	%↑	$RMSE$	%↓
PESQ	0.8185	N/A	0.460	N/A
GMM-2	0.8834	35.75	0.3649	20.67
GMM-3	0.8840	36.08	0.3602	21.70

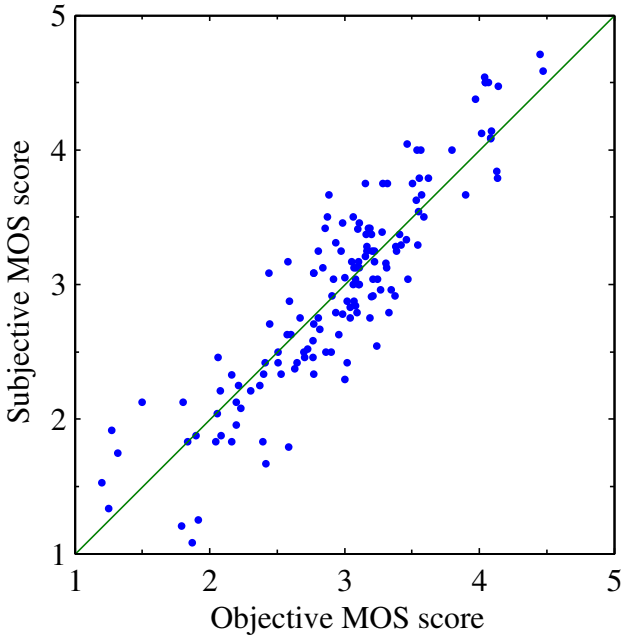


Figure 4.5: Subjective MOS *versus* objective MOS for CART-MARS selected features using three full Gaussian components.

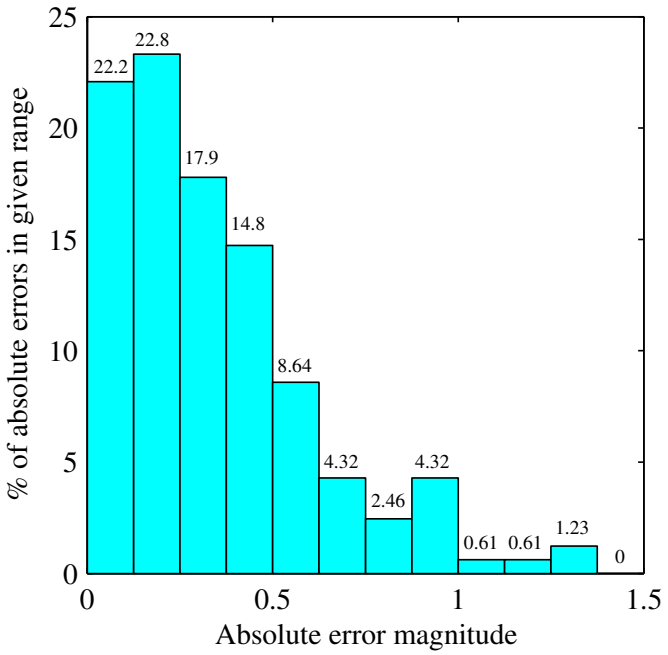


Figure 4.6: Residual error distribution for SFS selected features

are often regarded as being computationally intense; here we show that using such models is indeed worth the cost. We compare GM models with multiple linear regression models, $\mathbf{y} = \mathbf{Xb}$, with coefficients $\mathbf{b}$ estimated using least squares. Note that the first column of the matrix $\mathbf{X}$ is a column vector of all ones. This allows for an intercept term, namely b_0 to be estimated. Various experiments are tested. First we test linear regression models trained on the top-5 features selected by the four data mining algorithms. We also test with linear models trained on the top-20 features selected by CART and MARS. Lastly, we compare with a linear regressor trained on all 209 features. This last model achieves a training ratio of approximately 6, a value three times smaller than when using a 3-component, full covariance GMM. As can be seen in Table 4.11, all models described above result in considerably poorer estimation performance when compared to GMM estimators (vide Tables 4.3 - 4.10).

Table 4.11: Performance of multiple linear regression models

Features	R	$RMSE$
MARS (top-5)	0.7677	0.5126
CART (top-5)	0.7490	0.5299
CART-MARS (top-5)	0.7573	0.5217
SFS (top-5)	0.7972	0.4830
MARS (top-20)	0.8137	0.4659
CART (top-20)	0.7985	0.4824
All 209	0.8220	0.4579

This indicates that GMMs, despite their computational cost, can be used as accurate estimators of speech quality.

4.4 Summary

This chapter has presented the framework behind a novel objective speech quality estimation algorithm based on Gaussian mixture modelling. The usefulness of features selected by CART, MARS, a CART-MARS hybrid scheme, and the SFS algorithm are tested and the SFS algorithm has shown to provide the best performance. The CART-MARS hybrid scheme improved on CART by including features from unvoiced frames.

Diagonal Gaussian components have shown to provide only modest improvement over PESQ and this is attributed to the fact that the five most salient feature variables selected by the data mining techniques were correlated and the use of only five diagonal components was not sufficient to compensate for this. With full Gaussian components the correlation between features is properly modelled and significant improvement over PESQ is reported.

This experiment has shed light on the power of perceptual features for speech quality prediction. It has also highlighted the strengths and weaknesses of a GMM-based quality predictor algorithm. Results show that feature mining in conjunction with GMM modelling can produce simple estimators that outperform PESQ. Currently, CART/MARS selected features are suboptimal for GMM estimation and feature selection that directly optimizes GMM estimation performance would demonstrate if there is a gap relative to best possible performance. The next chapter focuses on a novel feature selection algorithm that directly optimizes GMM prediction performance.

Chapter 5

GMM-Based Feature Selection

5.1 Introduction

In Chapter 4, the first step towards an efficient GMM-based speech quality predictor is given. The choice of feature variables, however, is crucial in the speech quality estimation task, as redundant or noisy features degrade estimation performance. The problem at hand is to pick m feature variables out of $n > m$ variables for the GMM estimator. The best m is often not known *a priori*, and an exhaustive search for an optimal feature subset entails examining $2^n - 1$ possible subsets, a clearly impossible task for large n.

The approach used so far has been to use common feature selection algorithms such as CART [29] and MARS [30]. When designing GMM-based estimators, the features selected by the aforementioned algorithms may not lead to high estimation accuracy, especially when diagonal GMM estimators are used. Consequently, a feature selection algorithm that directly optimizes GMM prediction performance is highly sought.

In [64], the concept of *feature saliencies* in the context of GMMs is proposed.

By adopting a penalty criterion, saliencies of irrelevant features go to zero, thus performing feature selection. This feature selection procedure, however, does not take the GMM estimator into consideration and may still lead to features that are inefficient for the estimation task at hand.

In this chapter a feature mining algorithm targeted to estimation tasks that make use of GMM estimators is proposed. The algorithm is described in Section 5.2. A description of the algorithm's capabilities to perform N-survivor search is also presented in this section. Section 5.3 describes a speech quality prediction experiment where comparisons are carried out between GMM estimators trained on features selected by the proposed algorithm and GMM estimators trained on features selected by CART or MARS. Comparisons with PESQ and a test on unseen data are also shown in this section.

5.2 Algorithm Description

5.2.1 Feature Selection

It is argued in [45] that the GMM estimators have interesting relations to models such as CART and MARS in the sense that the mixture of Gaussians competitively partitions the feature space and learns a linear regression surface on each partition. Thus, it seems evident that one should use the GMM estimator to sift out the most relevant variables. The proposed sequential feature selection algorithm progressively constructs $\hat{f}$ using (3.11) or (3.13) as features are being selected.

The proposed algorithm starts with an empty feature set and features from a candidate feature set are added to the set progressively. To determine which candidate

feature to add, the algorithm tentatively adds to the current feature set one feature that is not already selected to form an augmented feature set. The joint density of the target variable and the augmented feature set is modelled with a GMM, with model parameters $\boldsymbol{\lambda}$ estimated using the EM algorithm. The accuracy of the GMM estimator using $\boldsymbol{\lambda}$ is then calculated. The above is repeated for every candidate feature and corresponding GMM. The candidate feature that produces the least regression error is admitted into the current feature set to form an updated feature set. The algorithm stops when the desired number of features has been selected.

It is worth mentioning that for each candidate feature the best number of Gaussian components in (3.1) can be determined by checking different values of M. Using the notation "EM" to stand for GMM parameter estimation via the EM algorithm, $\hat{f}_k$ for the mapping function with k variables, and D for the desired number of features, the algorithm can be summarized as follows:

Initialization: Let $I = \{1, \ldots, n\},\ S = \emptyset,\ k = 1$;

Step 1: $\boldsymbol{\lambda}_i \leftarrow \text{EM}(y,\ S \cup \{x_i\}), \forall i \in I$;

Step 2: $i_k = \arg\min_{i \in I} \sum_j (y_j - \hat{f}_k(S \cup \{x_i\}|\boldsymbol{\lambda}_i))^2$;

Step 3: $I \leftarrow I - \{i_k\},\ S \leftarrow S \cup \{x_{i_k}\},\ k \leftarrow k + 1$;

Step 4: Go to step 1, stop if $k > D$.

Care has to be taken to avert ill-conditioning of full covariance GMMs. As in Chapter 4, a small diagonal matrix, $\epsilon I_{n \times n}$, is added to each covariance matrix in each M-step iteration of the EM algorithm. By varying ϵ from 10^{-2} to 10^{-10}, tests show that the value that leads to best performance is $\epsilon = 10^{-9}$.

5.2.2 N-Survivor Search

With a corresponding increase in computational complexity, the algorithm can perform sequential multiple-survivor search. So far, the algorithm description has focused on one survivor, i.e., the one feature variable that minimizes estimation error. In N-survivor search, at each iteration, the N features that assume the top-N ranks in minimizing the estimation error are kept as "survivors". A tradeoff between complexity and performance can be adjusted by tuning the parameter N.

If the ultimate goal is to find D features out of n candidate features, then N survivors are kept in iterations $i = 1, 2, \ldots, D-1$. At iteration $i = 1$, the algorithm selects the N best features out of the n available candidates. At iterations $1 < i < D$ the N best ranked features, out of the $N(n-i+1)$ possible feature combinations, are kept. Lastly, at iteration $i = D$, the single best feature is kept. The last best feature and its ancestor features constitute the set of features selected by the search process.

The next section is dedicated to testing the accuracy of the proposed algorithm. Tests are performed with the proposed 1- and N-survivor search algorithm. Comparisons with GMM estimators trained on features selected by CART or MARS are carried out in the first experiment. Comparisons with PESQ are shown in the second, and an estimation test with unseen data is described in the third experiment. Results are presented for both diagonal and full covariance GMMs.

5.3 An Improved Speech Quality Predictor

In the following experiments, databases numbered 1-13 (refer to Table 4.2) are utilized. The combined thirteen databases contain 5864 speech file pairs. 10-fold cross

validation is used to provide some robustness in the performance evaluation. As before, estimation performance is assessed by R, as in (4.1) and by $RMSE$, as in (4.2).

It is worth mentioning that the feature pool used here is slightly different than the one used in Chapter 4. It was noted that the data mining algorithms often selected features that amount to probabilities of certain types of speech frames, e.g., V_P. In theory, these features carry no relevant speech quality information, and serve only as weights for the data mining models. It is also noted that features such as V_WM (weighted mean distortion of voiced frames) and V_RM (root-mean distortion) carry important speech quality information and were seldom chosen to be in the top-5 ranking. It is decided to combine [$\star$]_WM and [$\star$]_RM, where [$\star$] can be either V, U, or I, with their respective frame probabilities, e.g., V_RM $\times$ V_P, and use these new features to replace the frame probability features. A total of 212 $(209 - 3 + 6)$ features are thus used in the following experiments.

5.3.1 Experiment I

The first experiment compares GMM estimators trained on features selected by the proposed feature selection algorithm to estimators trained on features selected by CART or MARS. For this experiment all permissible values of M are checked at each iteration. This is done to obtain a feeling of how the GMM behaves as the algorithm progresses. To allow comparisons with Chapter 4 the top-5 features are chosen, and the number of Gaussian components is restricted to $M \leq 5$. With $M = 5$, an adequate training ratio (ratio between the number of parameters that have to be estimated during the training phase and the total number of files in the training set)

of 37 for full covariance matrices and 81 for diagonal matrices is maintained.

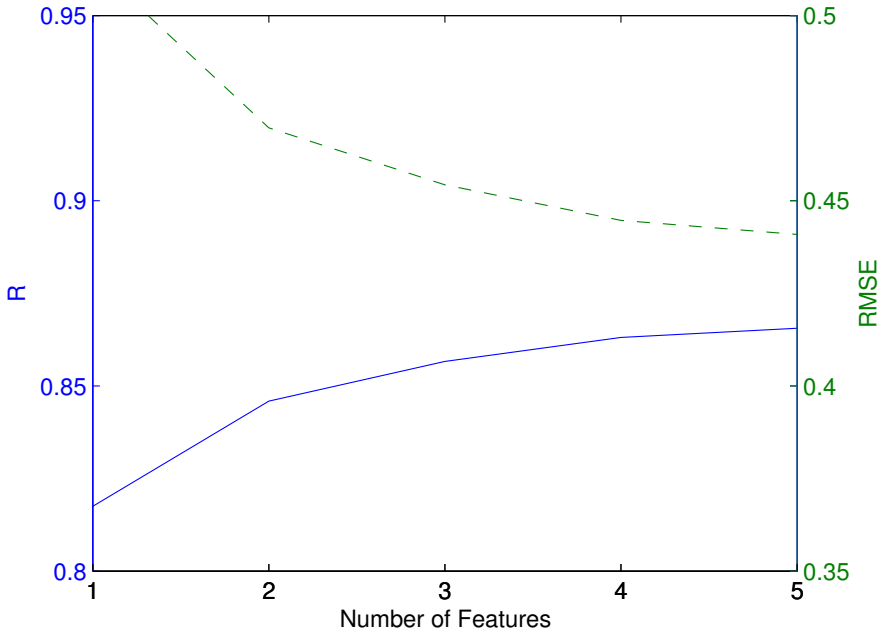
Let M_i be the number of Gaussian components chosen in iteration i of the proposed algorithm, the following combinations were often selected throughout the ten cross validation trials:

- Diagonal: $M_1 = 4,\ M_2 = M_3 = M_4 = M_5 = 5$;
- Full: $M_1 = 2,\ M_2 = 3,\ M_3 = M_4 = 4,\ M_5 = 5$.

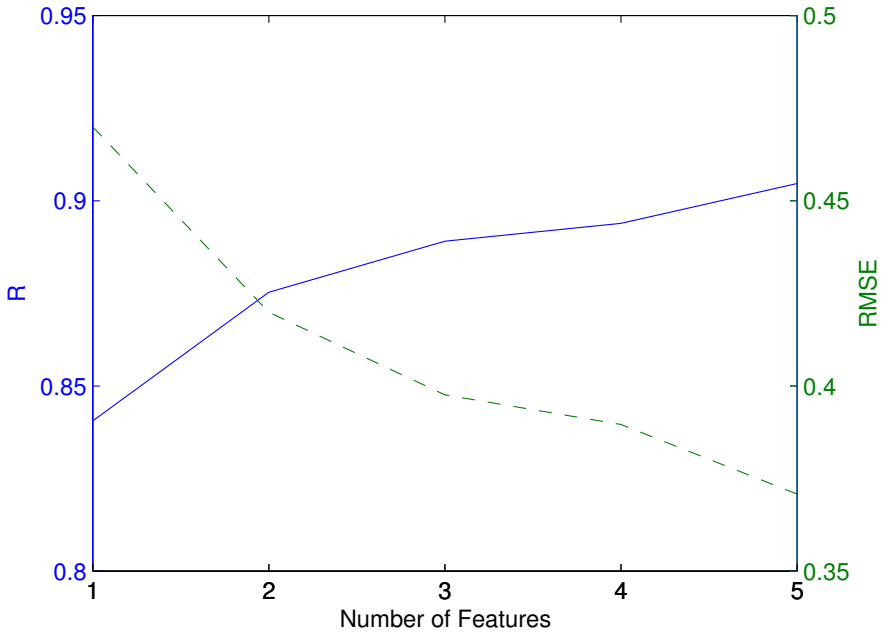
Note that over the five algorithm iterations (D=5) used in this experiment the number of Gaussian components either increases or stays the same as the algorithm progresses. As expected, full covariance GMMs use fewer Gaussian components at the beginning, and the number of components increases with the number of features. Figures 5.1 (a) and (b) show the effects the number of features has on R and $RMSE$, when using Gaussian components with diagonal and full covariance matrices, respectively. Also note that the above values of M_i will be used when N-survivor search is performed.

Tables 5.1 and 5.2 show the features selected using the proposed algorithm for diagonal and full covariance matrices, respectively. Features are presented for each of the ten cross validation trials. Table 5.3 lists the features selected using MARS. CART selected features V_O_1, V_O_2, V_O_3, V_O_4, and V_WM_2 for all trials but the fourth, where V_WM_2 was replaced by V_O_2_2.

Tables 5.4 and 5.5 compare performance figures for a five-component GMM estimator designed using the proposed algorithm to that of an estimator designed using CART or MARS, for diagonal and full covariance matrices, respectively. The column "%↑" indicates percentage improvement in R, as given by (4.3) but with R_{PESQ} replaced by R_{CART} or R_{MARS}. As can be seen, the proposed algorithm outperforms both benchmark algorithms. For diagonal GMM estimators an average improvement



(a)



(b)

Figure 5.1: R and $RMSE$ (dashed lines) as a function of the number of features for (a) diagonal and (b) full GMM estimators

Table 5.1: Feature variables selected by proposed algorithm (diagonal)

Cross Validation	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5
Trial 1	V_O_0	V_B_1_2	I_P_2	U_P_0	V_B_3
Trial 2	V_O_0	V_B_3	V_P_0	V_P_1	U_P_2
Trial 3	V_P_2	V_B_1_2	V_RM×V_P	U_P_1	I_P_2
Trial 4	V_O_0	I_O_0_1	U_B_4_1	REF_1	V_B_1_1
Trial 5	V_O_0	V_B_1	I_P_2	V_P_1	I_B_6
Trial 6	V_O_0	I_O_0_1	U_B_4_1	REF_0	V_B_2_2
Trial 7	V_RM×V_P	I_O_2	I_O_1_1	V_B_4	V_B_0_1
Trial 8	V_O_1	U_P_0	V_B_1_2	I_P_2	V_P_1
Trial 9	V_O_0	U_P_0	I_P_2	I_P_VUV	V_B_1_2
Trial 10	V_O_0	U_P_0	I_P_2	V_P_1	U_B_4

Table 5.2: Feature variables selected by proposed algorithm (full)

Cross Validation	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5
Trial 1	V_WM×V_P	I_B_5	V_P_VUV	U_B_4	I_P_VUV
Trial 2	V_O_2	I_B_6_1	REF_1	I_P_VUV	U_O_3_1
Trial 3	U_O_2	V_O_2	I_B_5	I_WM×I_P	V_O_2_2
Trial 4	V_O_1	I_B_4	V_WM×V_P	V_B_4	I_P_VUV
Trial 5	V_RM×V_P	I_B_5	V_O_0	V_B_4	I_O_0_0
Trial 6	V_WM×V_P	I_B_5	U_RM×U_P	REF_0	I_P_VUV
Trial 7	V_O_2	I_P_2	I_P_VUV	V_P_1	V_B_3
Trial 8	V_O_2	I_B_4	REF_0	I_B_2	REF_1
Trial 9	U_O_2	I_P_VUV	I_O_1	V_O_1	U_B_3
Trial 10	V_O_1	I_B_4	U_O_3	V_B_5_1	I_B_3_1

Table 5.3: Features selected by MARS

Cross Validation	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
Trial 1	I_P_VUV	V_O_2	I_O_5	I_O_4_0	V_B_2
Trial 2	V_O_2	I_P_VUV	I_O_5	U_P_0	I_O_4_0
Trial 3	V_O_2	I_P_VUV	I_O_5	U_P_0	V_O_0
Trial 4	I_P_VUV	V_O_2	I_O_5	V_B_2	I_O_0
Trial 5	I_P_VUV	I_O_5	I_O_0	V_B_2	I_O_4_0
Trial 6	I_P_VUV	V_O_2	I_O_0	I_O_5	V_B_2
Trial 7	I_P_VUV	I_O_5	I_O_0	V_O_2	I_O_4_0
Trial 8	I_P_VUV	I_O_0	I_O_5	I_O_4_0	V_O_0
Trial 9	V_O_2	I_P_VUV	I_O_5	V_O_0	V_B_2
Trial 10	I_P_VUV	I_O_0	I_O_5	I_O_4_0	V_B_2

in R of 26.95% and 38.94 %, and an average decrease in $RMSE$ of 13.93% and 24.16% is attained when compared to CART and MARS, respectively. An average improvement in R of 31.10% and 20.01%, and an average decrease in $RMSE$ of 19.07% and 11.96% is achieved for full GMM estimators.

If multiple survivor search is carried out, performance can be improved. There is, however, a linear increase in design complexity. The 1-survivor algorithm needs to invoke the EM algorithm $M\sum_{i=1}^{D}(n-i+1)$ times, n being the total number of candidate features and D the desired number of features to be selected. Here, $n = 212$ and $D = 5$. By using the N-survivor approach, the number of EM invocations increases to $NM\sum_{i=1}^{D}(n-i+1)$. A simple experiment is carried out with $N = 2$ and the selected features are presented in Table 5.6. The performance increase over single survivor search is shown in Table 5.7. An improvement of up to 7.21% in R and a reduction of 3.12% in $RMSE$ can be attained by using 2-survivor search relative to single-survivor search. Note that for trial 2 and trial 8, both 1- and 2- survivor algorithms select the same five features.

Table 5.4: Performance comparison (diagonal)

Cross Validation	Proposed		CART			
Trials	R	$RMSE$	R	%↑	$RMSE$	%↓
Trial 1	0.8578	0.4390	0.8083	25.82	0.5016	14.25
Trial 2	0.8539	0.4623	0.8216	18.11	0.5036	8.93
Trial 3	0.8530	0.4479	0.7972	27.51	0.5068	13.15
Trial 4	0.8732	0.4448	0.8206	29.32	0.4930	10.83
Trial 5	0.8416	0.4585	0.7937	23.22	0.5126	11.79
Trial 6	0.8694	0.4266	0.8184	28.08	0.4903	14.93
Trial 7	0.8740	0.4305	0.8171	31.11	0.5111	18.72
Trial 8	0.8656	0.4409	0.8171	26.52	0.4996	13.31
Trial 9	0.8521	0.4623	0.7879	30.27	0.5400	16.80
Trial 10	0.8677	0.4341	0.8122	29.55	0.5061	16.58
Average				**26.95**		**13.93**

	MARS			
	R	%↑	$RMSE$	%↓
Trial 1	0.7926	31.44	0.5206	18.58
Trial 2	0.7465	42.37	0.5577	20.63
Trial 3	0.7903	29.90	0.5140	14.75
Trial 4	0.7661	45.79	0.5821	30.86
Trial 5	0.6863	49.51	0.6151	34.15
Trial 6	0.7479	48.20	0.5709	33.82
Trial 7	0.8089	34.07	0.5243	21.78
Trial 8	0.8043	31.32	0.5159	17.01
Trial 9	0.8000	26.05	0.5255	13.67
Trial 10	0.7313	50.76	0.5919	36.35
Average		**38.94**		**24.16**

Table 5.5: Performance comparison (full)

Cross Validation	Proposed		CART			
Trials	R	$RMSE$	R	%↑	$RMSE$	%↓
Trial 1	0.8931	0.3830	0.8404	33.02	0.4627	20.81
Trial 2	0.8917	0.4005	0.8498	27.90	0.4656	16.25
Trial 3	0.8835	0.3930	0.8452	24.74	0.4480	13.99
Trial 4	0.9023	0.3907	0.8648	27.74	0.4557	16.63
Trial 5	0.8852	0.3923	0.8322	31.59	0.4671	19.06
Trial 6	0.8919	0.3889	0.8498	28.03	0.4531	16.50
Trial 7	0.8953	0.3955	0.8538	28.39	0.4626	16.96
Trial 8	0.9075	0.3644	0.8574	35.13	0.4467	22.58
Trial 9	0.8963	0.3889	0.8329	37.94	0.4846	24.61
Trial 10	0.9047	0.3708	0.8498	36.55	0.4572	23.30
Average				**31.10**		**19.07**

	MARS			
	R	%↑	$RMSE$	%↓
Trial 1	0.8694	18.15	0.4209	9.89
Trial 2	0.8816	8.53	0.4168	4.06
Trial 3	0.8651	13.64	0.4218	7.32
Trial 4	0.8923	9.29	0.4094	4.78
Trial 5	0.8336	31.01	0.4657	18.71
Trial 6	0.8742	14.07	0.4173	7.30
Trial 7	0.8900	4.82	0.4060	2.65
Trial 8	0.8749	26.06	0.4196	15.14
Trial 9	0.8553	28.33	0.4541	16.76
Trial 10	0.8229	46.19	0.4931	32.98
Average		**20.01**		**11.96**

Table 5.6: Feature variables selected by 2-survivor search (full)

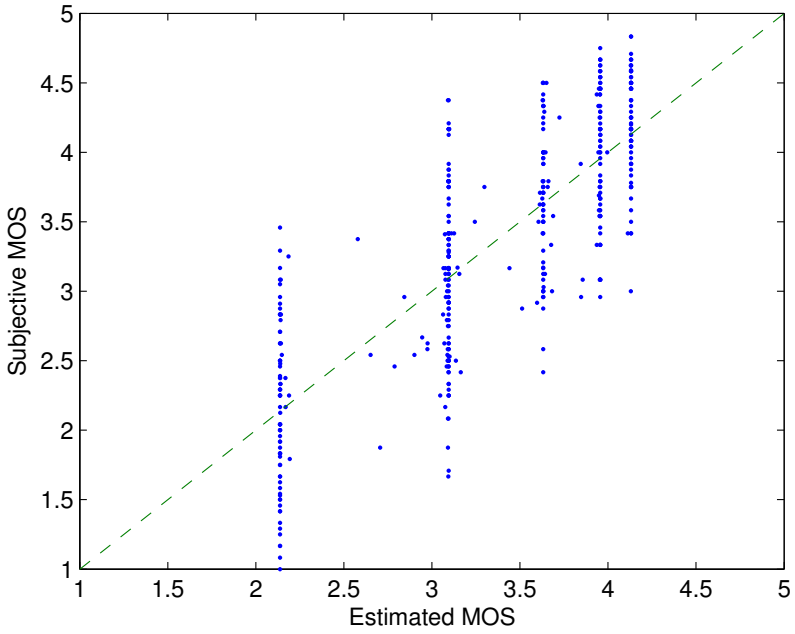
Cross Validation	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5
Trial 1	V_O_2	I_B_6	I_P_VUV	REF_0	V_RM×V_P
Trial 2*	V_O_2	I_B_6_1	REF_1	I_P_VUV	U_O_3_1
Trial 3	U_O_2	V_B_1	I_O_0	I_WM×I_P	V_O_0
Trial 4	V_O_1	I_O_0_1	U_P_0	I_B_5_1	I_P_VUV
Trial 5	V_RM×V_P	I_B_6	U_RM×U_P	I_O_0_1	REF_1
Trial 6	V_O_2	I_B_5	I_P_VUV	U_B_4	I_B_1_2
Trial 7	V_O_2	I_O_0_1	U_P_0	I_B_5	I_P_VUV
Trial 8*	V_O_2	I_B_4	REF_0	I_B_2	REF_1
Trial 9	U_O_2	I_P_VUV	I_RM×I_P	V_O_2	U_B_4_1
Trial 10	V_O_1	I_B_4	U_B_4	V_B_0_1	I_P_VUV

Table 5.7: Performance improvement of 2-survivor search relative to 1-survivor

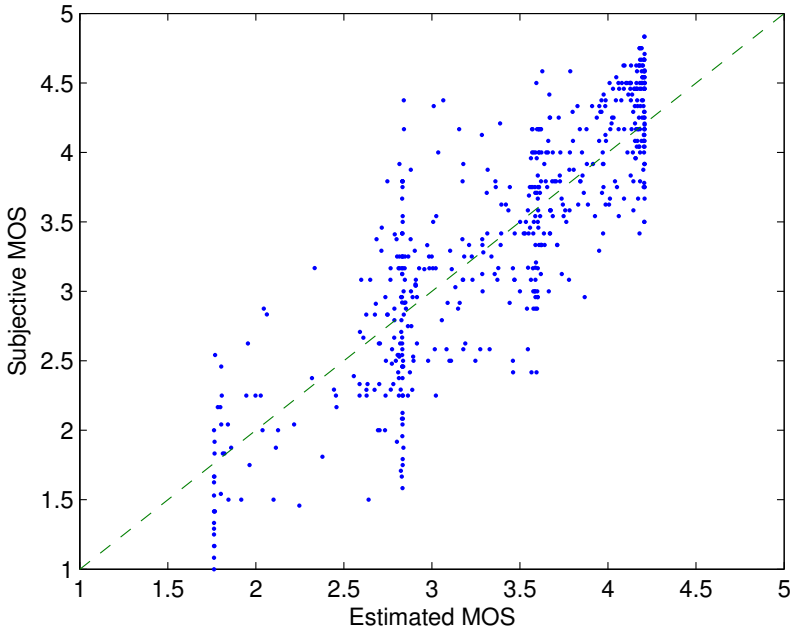
Cross Validation	R	%↑	$RMSE$	%↓
Trial 1	0.8987	5.23	0.3736	2.96
Trial 2*	0.8917	0.00	0.4005	0.00
Trial 3	0.8919	7.21	0.3807	3.13
Trial 4	0.9072	5.02	0.3818	2.27
Trial 5	0.8886	2.96	0.3869	1.38
Trial 6	0.8979	5.55	0.3790	2.55
Trial 7	0.9020	6.39	0.3841	2.88
Trial 8*	0.9075	0.00	0.3644	0.00
Trial 9	0.8991	2.71	0.3839	1.29
Trial 10	0.9085	3.98	0.3638	1.89

Chapter 4 highlights one of the major drawbacks of using CART and MARS for speech quality estimation and consists in the fact that features selected by the data mining algorithms have significant correlation amongst them. Diagonal covariance GMM estimators, consequently, present only modest performance figures. By looking back at Tables 5.4 and 5.5 it can be seen that CART selected features outperform MARS selected features with diagonal GMM estimators. This does not hold true when using full GMM estimators, suggesting that MARS selected features are more correlated.

Furthermore, in Chapter 4, a prominent vertical alignment of points in the subjective MOS *versus* estimated MOS scattered plots was shown for diagonal GMMs (vide Figure 4.2). This suggested that full covariance GMM estimators were needed in order to predict the residual variation in subjective MOS. If one insists on using diagonal GMM estimators, the problem is mitigated by using the proposed feature selection algorithm. For this experiment, Figures 5.2 (a) and (b) illustrate scattered plots for a GMM estimator trained on features selected by MARS and a GMM estimator trained on features selected by the proposed algorithm, respectively. Note that the vertical alignment of points is considerably less accentuated than MARS, reflecting the performance improvements shown in Table 5.4. Observe in Figure 5.2 (a) the five discrete estimated MOS values associated with the five diagonal Gaussian components (see (3.13)) are prominently indicated by the horizontal locations of the vertical clusters. In this case, the weights in (3.13) serve the sole purpose of switching between the five discrete values.



(a)



(b)

Figure 5.2: Subjective MOS *versus* objective MOS for (a) MARS and (b) diagonal GMM selected features

Table 5.8: Performance comparison: PESQ and proposed algorithm

Cross Validation	PESQ		1-Survivor (full)		2-Survivor (full)	
Trials	R	$RMSE$	$\uparrow\%R$	$\downarrow\%RMSE$	$\uparrow\%R$	$\downarrow\%RMSE$
Trial 1	0.8568	0.4643	25.35	17.51	29.26	19.53
Trial 2	0.8535	0.4871	26.08	17.78	26.08	17.78
Trial 3	0.846	0.4809	24.35	18.27	29.81	20.84
Trial 4	0.867	0.467	26.54	16.33	30.23	18.24
Trial 5	0.8449	0.4811	25.98	18.46	28.18	19.58
Trial 6	0.8564	0.4668	24.72	16.69	28.90	18.81
Trial 7	0.8738	0.4633	17.04	14.63	22.35	17.09
Trial 8	0.8581	0.4801	34.81	24.09	34.81	24.09
Trial 9	0.8608	0.4695	25.5	17.17	27.51	18.23
Trial 10	0.8623	0.4604	30.79	19.46	33.55	20.98
Average			**26.12**	**18.04**	**29.07**	**19.52**

5.3.2 Experiment II

In this experiment we compare performance of the GMM-based voice quality predictor to the performance of PESQ with the mapping proposed in [27]. Table 5.8 summarizes the performance figures; the column labelled "$\uparrow\%R$" shows improvement in R by using a 5-component GMM estimator, trained with features selected by the proposed algorithm. Similarly, "$\downarrow\%RMSE$" denotes decrease in $RMSE$ relative to PESQ. Full GMM estimators outperform PESQ by 26.12% and 18.04% in R and $RMSE$, respectively. With 2-survivor search, an average improvement of approximately 29% in R and an average decrease of 19.51% in $RMSE$ is attained. Additionally, it is important to note that, despite lower performance, full GMM estimators trained on features selected by CART or MARS also outperform PESQ, as was shown in [7].

Table 5.9: Performance comparison: PESQ and proposed algorithm – unseen data

GM	Database I		Database II	
Models	$\uparrow\%R$	$\downarrow\%RMSE$	$\uparrow\%R$	$\downarrow\%RMSE$
GMM 1	-5.40	34.54	-6.50	15.31
GMM 2	-9.51	34.72	-11.56	1.25
GMM 3	-4.50	44.91	-5.03	25.79
GMM 4	-4.55	48.50	-5.69	23.17
GMM 5	-2.68	38.78	-4.04	26.15
GMM 6	-8.08	34.52	-10.47	5.37
GMM 7	-9.06	32.47	-10.40	7.67
GMM 8	-4.24	51.37	-4.23	27.97
GMM 9	-2.78	52.24	-3.21	33.17
GMM 10	-3.22	42.02	-4.90	23.14
Average	**-5.40**	**41.41**	**-6.60**	**18.90**

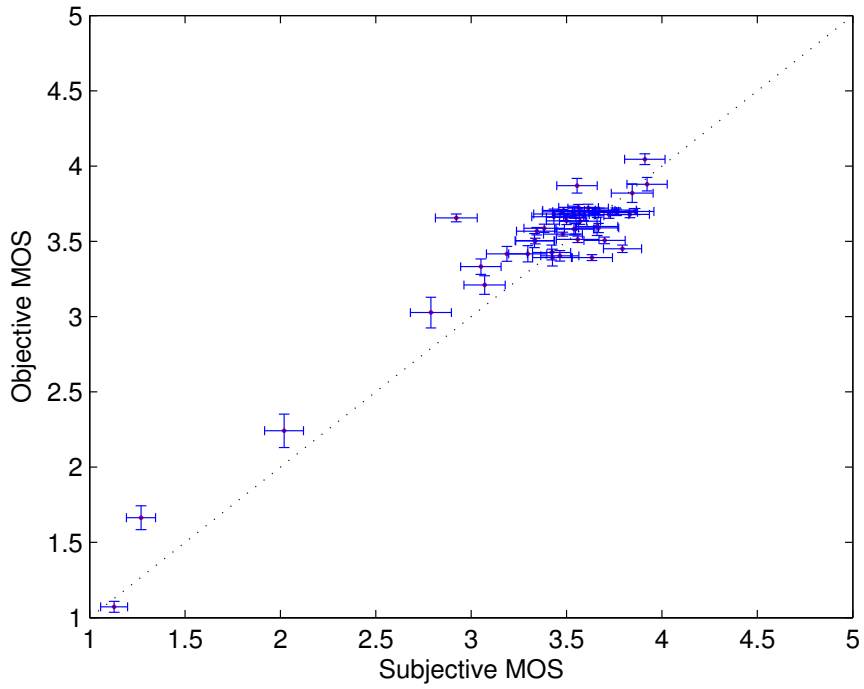
5.3.3 Experiment III

In this last experiment, the proposed algorithm is tested on unseen data, i.e., data that has not been used in the training of the GMM predictors. Two unseen test databases are used, each comprised of approximately 3000 subjectively scored speech file pairs, with speech under various degradation conditions. Table 5.9 shows increase in R and decrease $RMSE$ for the ten GM models found in the ten cross validation trials as per experiment 1. As can be seen, for the first database the proposed algorithm achieves an average 5.4% lower correlation when compared to PESQ. A 6.6% lower R is achieved for the second database. However, for the first database, the proposed algorithm reduces $RMSE$ by an average 41%. For the second database, an average decrease of 19% is attained. Recall from Section 4.3 that $RMSE$ is a more realistic measure of estimator performance.

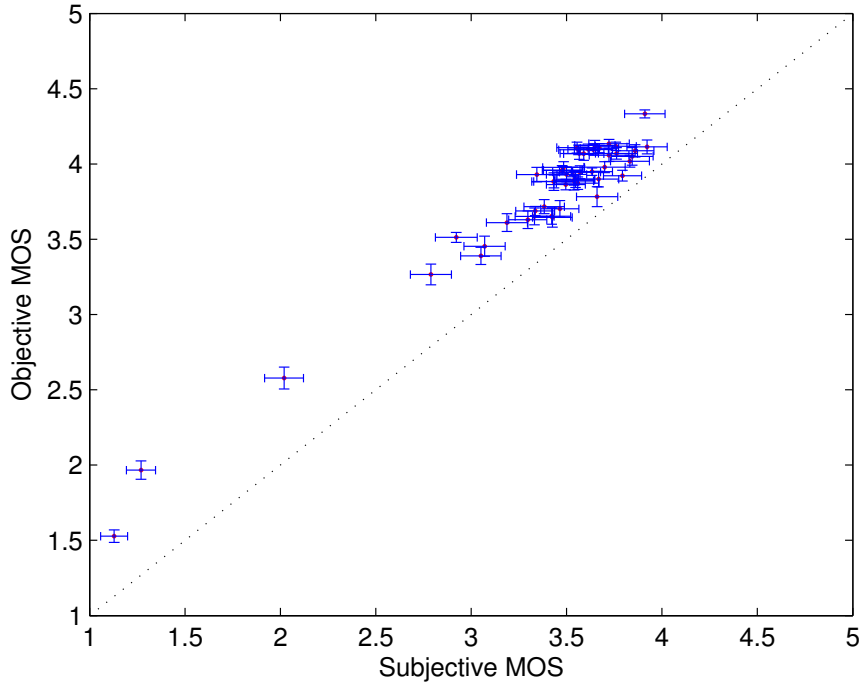
Figures 5.3 and 5.4 depict the objective MOS *versus* subjective MOS for the two unseen databases. Data points in the graphs represent the per-condition MOS rating, i.e., the average for all speech samples under the same degradation condition. Panels (a) and (b) illustrate the cases for the proposed algorithm and PESQ, respectively. The 95% confidence intervals are depicted as error bars, indicating the statistical distribution of the model outputs on different speech samples under the same degradation condition. Note how the proposed algorithm better predicts the quality of noisy speech samples, i.e., samples that have subjective MOSs between 1 and 2.

5.4 Summary

This chapter has proposed a novel feature selection algorithm for speech quality estimation based on Gaussian mixture models. The algorithm is targeted to applications that make use of GMM estimators, as features are selected to minimize squared GMM estimation errors. Moreover, the algorithm is capable of performing N-survivor search, allowing for a tradeoff between complexity and performance to be adjusted via the parameter N. Simulation results show that GMM estimators designed using the proposed algorithm outperform two benchmark selection algorithms, with N-survivor search incurring better performance. Furthermore, it has also been shown that features selected by the proposed algorithm are suitable for diagonal GMM estimators, which incur lower computational complexity. Lastly, a test on unseen data is carried out and the proposed algorithm is capable of reducing $RMSE$ by an average 41% when compared to PESQ.

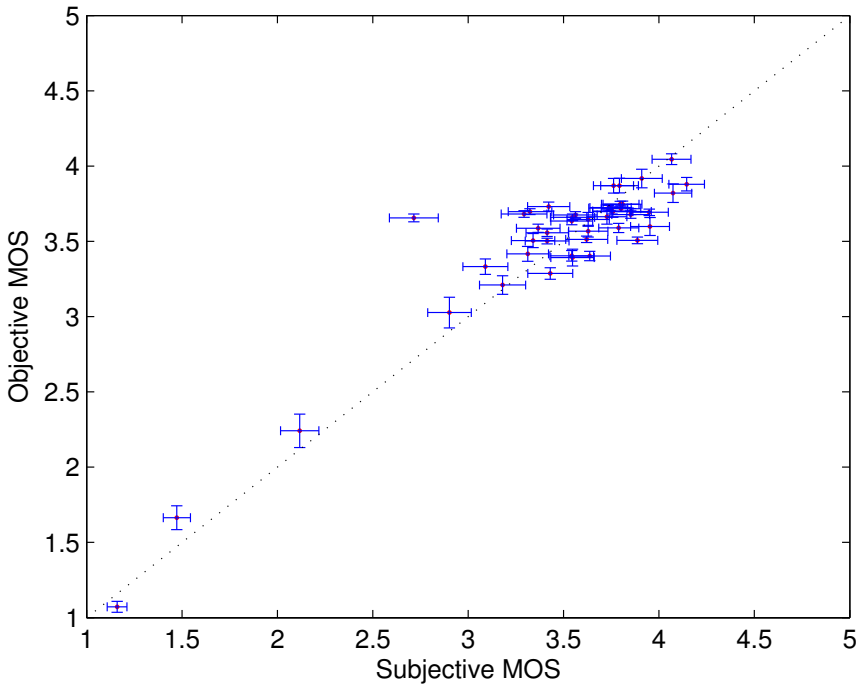


(a)

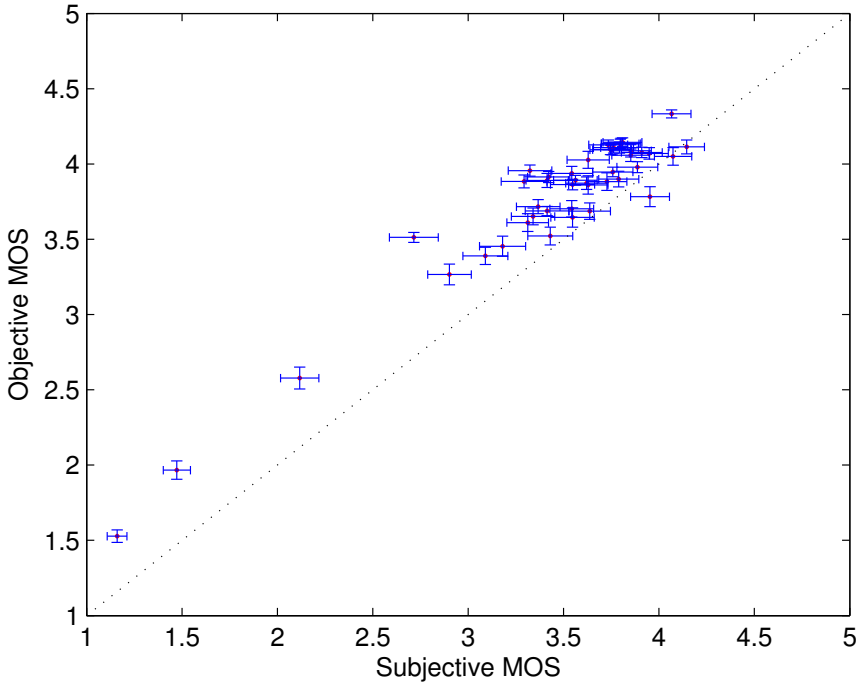


(b)

Figure 5.3: Objective MOS *versus* subjective MOS for (a) proposed algorithm (b) PESQ – Database 1.



(a)



(b)

Figure 5.4: Objective MOS *versus* subjective MOS for (a) proposed algorithm (b) PESQ – Database 2.

Chapter 6

Conclusions and Further Work

6.1 Conclusions

The evaluation of speech quality is of critical importance in today's telephone networks, be it the plain old telephone system, wireless or voice over IP, mainly because quality is a key determinant of customer satisfaction. Objective speech quality assessment algorithms provide low-cost and online monitoring of voice calls, replacing costly and time-consuming subjective listening tests. This thesis exploits the flexibility and simplicity of Gaussian mixture models (GMMs) in designing simple, yet effective predictors of perceived speech quality. A large pool of perceptual distortion features is extracted from speech files. Initially, classification and regression trees (CART), multivariate adaptive regression splines (MARS), and the sequential forward selection (SFS) algorithm are used to sift out the most relevant variables from the pool. The five most salient variables are used to construct good GMM estimators of subjective listening quality.

ITU-T's perceptual evaluation of speech quality (PESQ), the current "state-of-art" standard algorithm, is used to benchmark our proposed GMM-based voice quality predictor. This thesis shows that when using diagonal Gaussian components, the improvement over PESQ is more modest than when full Gaussian components are used. Careful investigation suggests that this is due to the fact that the five most salient feature variables selected by the data mining techniques are correlated and the use of only five diagonal Gaussian components is insufficient to compensate for this correlation. Furthermore, the features selected by the data mining schemes limit the performance of the proposed GMM-based voice quality predictor. GMMs are later used to devise a novel feature selection algorithm that directly optimizes GMM prediction performance.

The GMM-based sequential feature selection algorithm performs N-survivor search, trading complexity and accuracy via the parameter N. It is shown that GMM estimators, trained on features selected by the proposed algorithm outperform GMM estimators trained on features selected by CART or MARS. At the cost of extra computations, N-survivor search (here $N = 2$ was used) can improve performance by as much as 7.21% in correlation. Comparisons with PESQ show that the proposed algorithm incurs, on average, 26.12% higher R and 18.04% lower $RMSE$. Lastly, a test on unseen data is carried out and the proposed algorithm is capable of reducing $RMSE$ by an average 41% when compared to PESQ.

6.2 Further Work

This thesis has introduced a new paradigm in speech quality assessment: the use of simple, yet robust Gaussian mixture models for **intrusive** quality measurement.

As mentioned previously, intrusive measurement involves injecting a clean speech signal into a piece of equipment or a call connection and measuring the quality of the degraded speech signal at the output or receiving end. Non-intrusive measurement is a far more challenging approach to objective measurement – it does not require the injection of a clean reference signal to assist in the prediction of speech quality. Non-intrusive measurement is an emerging research field that once matured will allow for low-cost and continuous voice quality monitoring.

In [65], the first steps towards migrating GMMs to non-intrusive quality assessment are given. The approach compares perceptual features extracted from degraded speech to an artificial reference model. This model employs GMMs trained on features extracted from a dataset of clean speech signals and the degree of mismatch is used as an indicator of speech quality. Initial simulations show an accurate and yet low-complexity speech quality measurement algorithm.

Within the intrusive quality measurement framework, improvements can be found by defining new and improved features. It is known that "a classifier is only as good as its input features." The work presented here strives to demonstrate that simple yet effective voice quality predictors can be found and gain in performance is apparent if better features are proposed. Hopefully this thesis serves as incentive for the research on such improved features. Moreover, the schemes proposed here make use of features that are not dependent on the phase of the speech signal. In [66], phase information is shown to be important in the evaluation of sound quality. A possible expansion of the work presented here could be to experiment the usefulness of phase-dependent features and see if speech quality prediction accuracy can be increased.

Bibliography

[1] Psytechnics Limited, "Clear profit: The business case for managing voice quality – A whitepaper for US wireless service providers," March 2004.

[2] ITU-T Rec. P.830, "Subjective performance assessment of telephone-band and wideband digital codecs," International Telecommunication Union, Geneva, Switzerland, Feb. 1996.

[3] ITU-T Rec. P.800, "Methods for subjective determination of transmission quality," International Telecommunication Union, Geneva, Switzerland, Aug. 1996.

[4] ITU-T Rec. P.862, "Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," International Telecommunication Union, Geneva, Switzerland, Feb. 2001.

[5] W. Zha and W.-Y. Chan, "Objective speech quality measurement using statistical data mining," *EURASIP Journal of Applied Signal Proc.*, to appear in 2005.

[6] T. H. Falk and W.-Y. Chan, "Objective speech quality assessment using Gaussian mixture models," in *Proc. of the 22nd Biennial Symposium on Communications*, June 2004, pp. 169–171.

[7] T. H. Falk, W.-Y. Chan, and P. Kabal, "Speech quality estimation using Gaussian mixture models," in *Proc. of the Int. Conf. on Spoken Language Processing*, Oct. 2004, pp. 2013–2016.

[8] T. H. Falk and W.-Y. Chan, "Feature mining for GMM-based speech quality measurement," in *Proc. of the 38th Asilomar Conf. on Signals, Systems and Computers*, Nov. 2004.

[9] ——, "A sequential feature selection algorithm for GMM-based speech quality estimation," accepted to the *13th European Signal Proc. Conf. – EUSIPCO 2005*.

[10] ——, "An improved GMM-based voice quality predictor," submitted to the *9th European Conf. on Speech Communication and Technology – Interspeech 2005*.

[11] P. Gray, M. P. Hollier, and R. E. Massara, "Non-intrusive speech-quality assessment using vocal-tract models," in *IEE Proceedings - Vision, Image and Signal Processing*, vol. 147, no. 6, Dec. 2000, pp. 493–501.

[12] C. Jin and R. Kubichek, "Vector quantization techniques for output-based objective speech quality," in *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, vol. 1, May 1996, pp. 491–494.

[13] D.-S. Kim and A. Tarraf, "Perceptual model for non-intrusive speech quality assessment," in *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, vol. 3, May 2004, pp. 1060–1063.

[14] ITU-T P.563, "Single ended method for objective speech quality assessment in narrow-band telephony applications," International Telecommunication Union, Geneva, Switzerland, May 2004.

[15] ITU-T Rec. G.107, "The E-model, a computational model for use in transmission planning," International Telecommunication Union, Geneva, Switzerland, May 2000.

[16] L. Sun, "Speech quality prediction for voice over internet protocol networks," Ph.D. dissertation, University of Plymouth, 2004.

[17] R. Kubichek, D. Atkinson, and A. Webster, "Advances in objective voice quality assessment," in *Proc. IEEE Globecom Conf.*, 1991, pp. 1765–1770.

[18] P. Kroon, "Evaluation of speech coders," in *Speech Coding and Synthesis*, W. B. Kleijn and K. K. Paliwal, Eds. Elsevier Science, 1995, ch. 13, pp. 467–494.

[19] ETSI EG 201 377-1, "Speech processing, transmission and quality aspects," European Telecommunications Standard Institute, France, Oct. 2002.

[20] S. Quackenbush, T. Barnwell, and M. Clements, *Objective Measures of Speech Quality*. Englewood Cliffs, New Jersey: Prentice-Hall, 1988.

[21] N. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Englewood Cliffs, New Jersey: Prentice-Hall, 1984.

[22] S. Wang, A. Sekey, and A. Gersho, "An objective measure for predicting subjective quality of speech coders," *IEEE Journal on Selected Areas in Communications*, vol. 10, no. 5, pp. 819–829, June 1992.

[23] S. Voran, "Objective estimation of perceived speech quality – Part I: Development of the measuring normalizing block technique," *IEEE Trans. on Speech and Audio Processing*, vol. 7, no. 4, pp. 371–382, July 1999.

[24] ——, "Objective estimation of perceived speech quality – Part II: Evaluation of the measuring normalizing block technique," *IEEE Trans. on Speech and Audio Processing*, vol. 7, no. 4, pp. 383–390, July 1999.

[25] ITU-T Rec. P.861, "Objective quality measurement of telephone-band (300-3400 Hz) speech codecs," International Telecommunication Union, Geneva, Switzerland, Aug. 1996.

[26] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, "PESQ - the new ITU standard for end-to-end speech quality assessment," in 109^{th} *AES Convention*, Sept., pp. 1–18, pre-print 5260.

[27] ITU-T P.862.1, "Mapping function for transforming P.862 raw result scores to MOS-LQO," International Telecommunication Union, Geneva, Switzerland, Nov. 2003.

[28] E. Zwicker and H. Fastl, *Psychoacoustics - Facts and Models.* Springer-Verlag, 1990.

[29] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees.* Monterey, CA: Wadsworth & Brooks, 1984.

[30] J. H. Friedman, "Multivariate adaptive regression splines," *The Annals of Statistics*, vol. 19, no. 1, pp. 1–141, March 1991.

[31] Psytechnics Limited, "NiQA - Product Description," Tech. Rep., Jan. 2003. [Online]. Available: http://www.psytechnics.com/pages/products/niqa.php

[32] SwissQual Inc., "NiNA - SwissQual's non-intrusive algorithm for estimating the subjective quality of live speech," Tech. Rep., June 2001.

[33] D. Titterington, A. Smith, and U. Makov, *Statistical Analysis of Finite Mixture Distributions*. John Wiley & Sons Ltd., 1985.

[34] S. Newcomb, "A generalized theory of the combination of observations so as to obtain the best result," *Amer. J. Math.*, vol. 8, pp. 343–366, 1886.

[35] K. Pearson, "Contribution to the mathematical theory of evolution," *Phil. Trans. Roy. Soc. A*, vol. 185, pp. 71–110, 1894.

[36] A. K. Aiyer, "Robust image compression using Gauss mixture models," Ph.D. dissertation, Stanford University, Aug. 2001.

[37] D. Reynolds and R. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Trans. on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, Jan. 1995.

[38] S. Kulkarni, G. Lugosi, and S. Venkatesh, "Learning pattern classification – A survey," *IEEE Trans. Inform. Theory*, vol. 44, no. 6, pp. 2178–2206, Oct. 1998.

[39] Y. Rosseell, "Mixture models of categorization," in *Journal of Mathematical Psychology*, vol. 46, April 2002.

[40] A. Kain and Y. Stylianou, "Stochastic modeling of spectral adjustment for high quality pitch modification," in *Proc. of the Int. Conf. on Acoustics, Speech, and Signal Proc.*, vol. 2, June 2000, pp. 949–952.

[41] X. Zhuang, Y. Huang, K. Palaniappan, and Y. Zhao, "Gaussian mixture density modeling, decomposition and applications," *IEEE Trans. on Image Processing*, vol. 5, no. 9, pp. 1293–1302, Sept. 1996.

[42] L. Heck and K. Chou, "Gaussian mixture model classifiers for machine monitoring," in *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, vol. 6, 1994.

[43] P. Hedelin and J. Skoglund, "Vector quantization based on Gaussian mixture models," *IEEE Trans. on Speech and Audio Processing*, vol. 8, no. 4, pp. 385–401, July 2000.

[44] M. Figueiredo and A. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 381–396, March 2002.

[45] Z. Ghahramani and M. I. Jordan, "Supervised learning from incomplete data via an EM approach," in *Advances in Neural Information Processing Systems*, vol. 6. Morgan Kaufmann Publishers, Inc., 1994, pp. 120–127.

[46] M. Jordan and R. Jacobs, "Hierarchical mixtures of experts and the EM algorithm," *Neural Computations*, no. 6, pp. 181–214, 1994.

[47] N. Kambhatla, "Local models and Gaussian mixture models for statistical data processing," Ph.D. dissertation, Oregon Graduate Institute of Science and Technology, Jan. 1996.

[48] D. Ormoneit and V. Tresp, "Improved Gaussian mixture density estimates using Bayesian penalty terms and network averaging," in *Advances in Neural Information Processing Systems*, vol. 8. The MIT Press, 1996, pp. 542–548.

[49] A. Lindemann, C. L. Dunis, and P. Lisboa, "Probability distributions, trading strategies and leverage: An application of Gaussian mixture models," Tech. Rep., 2003.

[50] B. Resch, "Mixtures of Gaussians: A tutorial for the course computational intelligence." [Online]. Available: http://www.igi.tugraz.at/lehre/CI

[51] R. Bellman, *Adaptive Control Processes*. Princeton, NJ: Princeton University Press, 1961.

[52] A. Dempster, N. Lair, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Statistical Society*, vol. 39, pp. 1–38, 1977.

[53] A. Subramaniam and D. Rao, "Pdf optimized parametric vector quatization of speech line spectral frequencies," *IEEE Trans. on Speech and Audio Processing*, vol. 11, no. 2, pp. 130–142, March 2003.

[54] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[55] L. Xu and M. I. Jordan, "On convergence properties of the EM algorithm for Gaussian mixtures," *Neural Computation*, vol. 8, no. 1, pp. 129–151, 1996.

[56] A. Gersho and R. Gray, *Vector Quantization and Signal Compression*. Kluwer Academic Publishers, 1992.

[57] T. Koshizen, Y.Rosseel, and Y. Tonegawa, "A new EM algorithm using Tikhonov regularization," in *International Joint Conference on Neural Networks*, vol. 1, 1999, pp. 413–418.

[58] R. Hathaway, "A constrained formulation of maximum-likelihood estimation for Normal mixture distributions," *Annals of Statistics*, vol. 13, no. 2, pp. 795–800, June 1985.

[59] M. Kloppenburg and P. Tavan, "Deterministic annealing for density estimation by multivariate Normal mixtures," March 1997.

[60] R. Martin, C. Hoelper, and I. Wittke, "Estimation of missing LSF parameters using Gaussian mixture models," in *Proc. of the Int. Conf. on Acoustics, Speech, and Signal Proc.*, vol. 2, May 2001, pp. 729–732.

[61] A. Jain and D. Jongker, "Feature selection: evaluation, application, and small sample performance," in *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 2, Feb. 1997, pp. 153–158.

[62] W. Zha and W.-Y. Chan, "A data mining approach to objective speech quality measurement," in *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, vol. 1, May 2004, pp. 461–464.

[63] ITU-T Rec. P. Supplement 23, "ITU-T coded-speech database," International Telecommunication Union, Geneva, Switzerland, Feb. 1998.

[64] M. Law, M. Figueiredo, and A. Jain, "Simultaneous feature selection and clustering using mixture models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1154–1166, Sept. 2004.

[65] T. H. Falk, Q. Xu, and W.-Y. Chan, "Non-intrusive GMM-based speech quality measurement," in *Proc. of the Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 1, March 2005, pp. 125–128.

[66] M. Karjalainen, "A new auditory model for the evaluation of sound quality of audio systems," in *Proc. of the Int. Conf. on Acoustics, Speech, and Signal Proc.*, vol. 10, April 1985, pp. 608–611.

Appendix A

Feature Description

The perceptual features used in this thesis are defined as follows. The first letter, denoted by T in a variable name, gives the frame type: T=I for "Inactive", T=V for "Voiced", and T=U for "Unvoiced". The subband index is denoted by b, with $b \in \{0, \ldots, 6\}$ indexing from the lowest to the highest frequency band if the index is natural, or from the highest to the lowest distortion if the index is rank-ordered. The frame distortion severity class is denoted by d, with $d \in \{0, 1, 2\}$ indexing from lowest to highest severity. With the above notations, the feature variables are:

- T_P_d : percentage of T frames in severity class d frames
- T_P : percentage of T frames in the speech file
- T_P_VUV : ratio of the number of T frames to the total number of active (V and U) speech frames
- T_B_b : distortion for subband b of T frames, without distortion severity classification, e.g., I_B_1 represents subband 1 distortion for inactive frames

- T_B_*b*_*d* : distortion for severity class *d* of subband *b* of T frames, e.g., V_B_3_2 represents distortion for subband 3, severity class 2, of voiced frames
- T_O_*b* : distortion for ordered subband *b* of T frames, without severity classification, e.g., U_O_3 represents ordered-subband 3 distortion for unvoiced frames, without distortion severity classification
- T_O_*b*_*d* : distortion for distortion class *d* of ordered subband *b* of T frames, e.g., U_O_6_1 represents distortion for severity class 1 of ordered-subband 6 of unvoiced frames
- T_WM_*d* : weighted mean distortion for severity class *d* of T frames
- T_WM : weighted mean distortion for T frames
- T_RM_*d* : root-mean distortion for severity class *d* of T frames
- T_RM : root-mean distortion for T frames
- REF_1 : high-frequency spectral energy of reference signal
- REF_0 : lowest-frequency spectral energy of reference signal

The weighted mean of the 7 subband distortions is calculated using the weights

$$w_i = \begin{cases} 1.0, & \text{for } 0 \leq i \leq 4 \\ 0.8, & \text{for } i = 5 \\ 0.4, & \text{for } i = 6. \end{cases} \tag{A.1}$$

The root-mean distortion is calculated in the following manner. Each frame has a 7-band loudness distortion vector (d_0 d_1 ... d_6) where d_i is the difference between

the loudness of the i^{th} band of the reference signal and the loudness of the i^{th} band of the degraded signal. If the distortion of a given frame j, $j = 1, \ldots, N_f$, is given by

$$D_j = \sum_{i=0}^{6} d_i,$$

then the root-mean distortion (D_{RM}) is given by

$$D_{RM} = \sqrt{\frac{\sum_j D_j}{7 * N_f}}. \tag{A.2}$$