Exploring the Dimensionality of Speech using Manifold Learning and Dimensionality Reduction Methods

Andrew Errity, B.Sc.

A dissertation submitted in fulfilment of the requirements for the award of

Doctor of Philosophy

to the

School of Computing

Faculty of Engineering and Computing

Dublin City University

Supervisor: Dr. John McKenna

January 2010

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy is entirely my own work, that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: _____ ID No.: 99086921

(Andrew Errity)

Date: January 17, 2010

Acknowledgements

Firstly, I would like to express my sincere thanks to John McKenna. Several years ago, as an enthusiastic lecturer, John introduced me to the field of speech processing. Subsequently, as my supervisor, he has provided knowledge, guidance, and advice that was invaluable throughout the course of this research.

I gratefully acknowledge the financial support of the Irish Research Council for Science, Engineering and Technology; grant number RS/2003/114. I also recognise the National Centre for Language Technology for providing access to various corpora that were useful in the development of this work.

My thanks go to Barry Kirkpatrick, my colleague in the Speech Group, for countless enjoyable discussions relating to the field of signal processing and many other topics. Also, to all my friends and colleagues in the School of Electronic Engineering, with particular mention to: Alan, Aubrey, Brendan, Diane, John, Marie, Paddy, Sean, and Trish. Not forgetting all those Friday footballers who joined me in a welcome distraction from work.

I am, as ever, grateful to my parents, Paul and Gertrude, and brother, David, for their encouragement and support. Also, to all my friends outside of DCU.

Finally, thanks to Amanda for her never-ending support during what may have seemed like the never-ending process of thesis writing.

Contents

Abstract

Many previous investigations have indicated that speech data has inherent low-dimensional structure and that it may be possible to efficiently represent speech using only a small number of parameters. This view is motivated by the fact that articulatory movement is limited by physiological constraints and thus the speech production apparatus has only limited degrees of freedom. Also, the set of sounds used in human spoken communication is only a small subset of all producible sounds. A number of dimensionality reduction methods capable of discovering such underlying structure have previously been applied to speech. However, if speech lies on a manifold nonlinearly embedded in high-dimensional space, as has been proposed in the past, classic linear dimensionality reduction methods would be unable to discover this embedding. In this dissertation a number of manifold learning, also referred to as nonlinear dimensionality reduction, methods are applied to speech to explore the possibility of underlying nonlinear manifold structure.

This dissertation describes a number of existing manifold learning methods and details the application of these methods to high-dimensional feature representations of speech data. Representations derived from the conventional magnitude spectrum and less widely used phase spectrum are investigated. The manifold learning methods used in this study are locally linear embedding, Isomap, and Laplacian eigenmaps. The classic linear method, principal component analysis (PCA), is also applied to facilitate the comparison of linear and nonlinear methods. The resulting low-dimensional representations are analysed through visualisation, phone recognition, and speaker recognition experiments. The recognition experiments are used as a means of evaluating how much meaningful discriminatory information is contained in the low-dimensional representations produced by each method. These experiments also serve to display the potential value of these methods in speech processing applications.

The manifold learning methods are shown to be capable of producing meaningful low-dimensional representations of speech data suggesting speech has low-dimensional manifold structure. In general, these methods are found to outperform PCA in low dimensions, indicating that speech may lie on a manifold nonlinearly embedded in high-dimensional space. Phone classification experiments show that Isomap can offer improvements over standard features and PCA-transformed features. Investigation of magnitude and phase spectrum

representations found both to have similar low-dimensional structure and confirm that the phase spectrum contains useful information for phone discrimination. Results indicate that combining magnitude and phase spectrum information yields improvements in phone classification tasks. A method to combine magnitude and phase spectrum features for increased phone classification accuracy without large increases in feature dimensionality is also described.

List of Figures

List of Tables

List of Abbreviations

ASR	Automatic speech recognition
DCT	Discrete cosine transform
EGG	Electroglottographic
EM	Expectation maximisation
EPG	Electropalatographic
Fn	nth formant
FFT	Fast Fourier transform
GDF	Group delay function
GMM	Gaussian mixture model
IPA	International Phonetic Alphabet
Isomap	Isometric feature mapping
K-NN	K-nearest neighbour
KLT	Karhunen-Loève transform
LDA	Linear discriminant analysis
LDF	Linear discriminant function
LEM	Laplacian eigenmaps
L-Isomap	Landmark isometric feature mapping
LLE	Locally linear embedding
LPC	Linear prediction coefficient
MDS	Multidimensional scaling
MFCC	Mel frequency cepstral coefficient
MGDF	Modified group delay function
MIT	Massachusetts Institute of Technology
MODGDF	Modified group delay feature
PCA	Principal component analysis
pdf	Probability density function
PLP	Perceptual linear prediction
RBF	Radial basis function
SNR	Signal-to-noise ratio
SOM	Self-organizing map
STFT	Short-time Fourier transform
SVD	Singular value decomposition
SVM	Support vector machine
TI	Texas Instruments

List of Notation

$\mathcal{B}$	upper case calligraphic letters indicates sets
$\mathbf{b}$	lower case and boldface is used for column vectors
$\mathbf{B}$	upper case and boldface is used for matrices
b_{ij}	ith-row jth-column entry of matrix $\mathbf{B}$
i, j	indices
$\boldsymbol{\mu}_{\mathbf{B}}$	column vector mean of $\mathbf{B}$
$\mathbf{C}_{\mathbf{B}}$	covariance matrix of $\mathbf{B}$
$\mathbf{I}$	identity matrix
$\mathbf{a} \cdot \mathbf{b}$	dot product
$\|\mathbf{b}\|$	Euclidean norm, $\sqrt{\mathbf{b}'\mathbf{b}}$
$\|\mathbf{B}\|$	Determinant of B
$.'$	vector or matrix transposition
Tr[$\mathbf{B}$]	trace, the sum of the diagonal elements, of matrix $\mathbf{B}$
$\log b$	base-10 logarithm of scalar b
$\Gamma(\cdot)$	Gamma function
$H(\cdot)$	Heaviside function
S	MDS stress function
Var[$\cdot$]	variance
$\mathbb{R}^d$	d-dimensional Euclidean space
λ	eigenvalue
$\mathbf{\Lambda}$	diagonal matrix of eigenvalues
$\boldsymbol{\alpha}$	eigenvector
$\mathbf{A}$	matrix of eigenvectors
$\mathbf{X}, \mathbf{x}$	original high-dimensional data matrix, points
$\mathbf{Y}, \mathbf{y}$	low-dimensional data matrix, points
D	dimensionality of original high-dimensional data
d	dimensionality of projected low-dimensional data or inherent dimensionality of data
s	scalar speech signal sample
τ	length of time delay embedding
d_e	embedding dimension
e	Euler's number
$\mathbf{T}$	matrix of inner products
k	number of nearest neighbours used in manifold learning
$O(f(b))$	computational complexity of order $f(b)$

Chapter 1

Introduction

In this dissertation the possibility of an inherent low-dimensional manifold[1] structure to speech signals is explored through the application of a number of manifold learning, also referred to as nonlinear dimensionality reduction,[2] methods. The aim of these methods is to transform high-dimensional data points into a meaningful low-dimensional space, removing redundant information, and maintaining important relationships between the data points.

In this chapter, we introduce the motivations behind the proposal that speech has a low-dimensional manifold structure and outline our approach to discovering this structure and exploiting it for use in speech processing applications. Following this, the principal contributions of this work are summarised and the structure of this dissertation is outlined. This chapter concludes with a list of publications produced during the preparation of this dissertation.

1.1 Preamble

Speech has evolved as the primary form of communication used by humans. As a result, the speech production and perception processes have been the subjects of a large amount of research for many decades. In recent years, the advancement and prevalence of digital computing has both inspired and facilitated the development of speech process-

[1] A manifold is a topological space that is locally Euclidean. The term 'manifold' is discussed further in Section 4.3.2.

[2] The terms 'manifold learning' and 'nonlinear dimensionality reduction' are used interchangeably throughout the remainder of this dissertation.

ing technologies enabling high-quality human-machine spoken communication. However, human-quality artificial speech production and perception have not yet been achieved.

It is therefore important to continue studying the processes involved in human speech communication, furthering existing knowledge and developing new approaches. This dissertation proceeds in this spirit, applying new methods in conjunction with existing knowledge in an effort to examine the underlying structure of speech and develop approaches to exploit this structure.

1.2 Motivation

Speech production is a complex process involving coordinated movement of the respirator muscles, glottis, and articulators. This process produces an acoustic speech signal transmitting a large amount of information. This speech signal[3] can be viewed as a high-dimensional information stream. A common way to represent this signal is to measure the energy in hundreds of different frequency bands, computed over short time frames, sampled every 10–25 ms. Each frequency band can be thought of as a single dimension in multidimensional space, with the number of dimensions equal to the number of frequency bands (Pols, 1971). Thus, every speech sample is represented as a point in this multidimensional space.

However, due to physiological constraints on the movement of the articulators the speech production apparatus has limited degrees of freedom[4]. In addition, only a small subset of sounds from the set of all possible sounds producible by the speech production apparatus are used in all human spoken communication (Nowak and Krakauer, 1999). This motivates the view that speech has an inherent low-dimensional structure and that the underlying variability of the speech data stream can be parametrised by a small number of features. In this case, we can imagine speech data as lying on a low-dimensional manifold embedded in a high-dimensional space; an example of this is shown in Figure 1.1. The presence of low-dimensional structure in speech is supported by previous studies dating back as far as the classic formant plane first described by Peterson and Barney (1952),

[3] All references to the speech 'signal' in this work refer to the acoustic speech waveform as opposed to any other signal, for example that resulting from electropalatography, unless otherwise stated.

[4] Here we use the term 'degrees of freedom' to describe the number of ways in which units of motor control, specifically the articulators, are capable of moving (Rose and Christina, 2005).

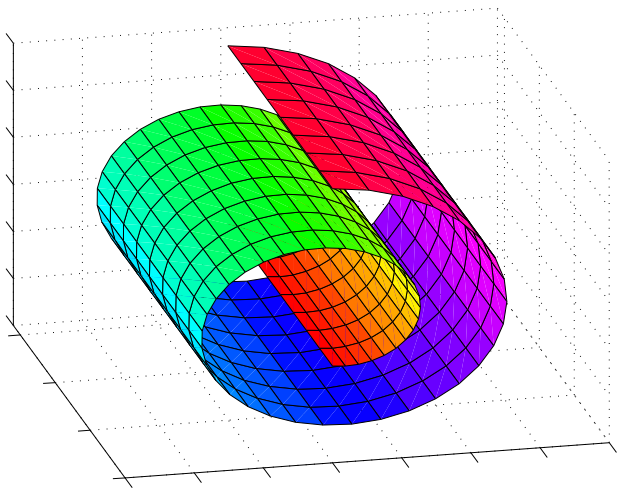


Figure 1.1: A two-dimensional manifold nonlinearly embedded in three-dimensional space.

studies of the perceptual and physical space of vowels conducted by Pols et al. (1969), and the important articulatory parametrisations developed by Fant (1970).

Conventionally, signal processing techniques such as the discrete Fourier transform and linear prediction analysis are applied to speech in order to facilitate the extraction of information that is judged to capture information about the energy and spectral characteristics of the original signal while discarding information that is deemed to be of no interest, often resulting in a reduction in the signal's dimensionality. The extracted information is often transformed according to some perceptually motivated scheme to better model the speech communication system; for example, mel frequency cepstral coefficients (MFCCs) and perceptual linear prediction (PLP) features.

These acoustically and perceptually motivated representations are based on established knowledge and assumptions made of the speech communication apparatus and as such do not attempt to automatically discover the inherent low-dimensional structure of speech. A number of dimensionality reduction methods, driven by the statistics of the data, have been proposed that aim to transform high-dimensional data into a meaningful lower-dimensional space. Applications of these dimensionality reduction methods include data compression, visualisation, noise reduction, and extraction of significant features from high-dimensional data.

Dimensionality reduction methods can be categorised as linear or nonlinear methods.

Linear methods are limited to discovering the structure of data lying on or near a linear subspace of the high-dimensional input space. The most widely used linear dimensionality reduction methods include principal component analysis (PCA) (Jolliffe, 1986), linear discriminant analysis (LDA) (Duda et al., 2000), and multidimensional scaling (MDS) (Cox and Cox, 2001). These methods have previously been applied to a wide range of speech processing problems including: feature transformation for improved automatic speech recognition (ASR) performance (Eisele et al., 1996; Somervuo, 2003a; Wang and O'Shaughnessy, 2003; Schuster et al., 2005), speaker adaptation (Malayath et al., 1997; Kuhn et al., 1998), data compaction (Beyerbach and Nawab, 1991), and speech analysis (Plomp et al., 1967; Pols et al., 1969; Klein et al., 1970; Pols, 1971; Pols et al., 1973; Pijpers et al., 1993).

However, Togneri et al. (1992) and Jansen and Niyogi (2005) have presented evidence suggesting that speech lies on a low-dimensional manifold *nonlinearly* embedded in high-dimensional space; an example of a nonlinear embedding is provided in Figure 1.1. In this case linear methods would be unable to discover the underlying low-dimensional structure. A number of manifold learning (Seung and Lee, 2000) methods have been proposed (Roweis and Saul, 2000; Tenenbaum et al., 2000; Belkin and Niyogi, 2002) which attempt to overcome the limitations of linear dimensionality reduction methods. These methods have been successfully applied to a number of problems in the image processing field, such as: multi-pose (Hadid et al., 2002) and multi-expression (Wang et al., 2004) analysis of face images; synthesis of face images (Zhang et al., 2004; Wang et al., 2003); and inference of 3D body pose based on silhouettes (Elgammal and Lee, 2004). Manifold learning methods may also be useful in speech processing applications; for example, to visualise speech in a low-dimensional space and extract features for use in ASR and speaker recognition tasks. However, there is relatively little preexisting research in this area.

Further motivation for the application of nonlinear methods, such as manifold learning, to speech is provided by the large body of existing work detailing the nonlinear processes at work in the speech communication systems (Kubin, 1995).

1.3 Summary of proposed approach

The aims of this work are to determine if speech has an inherent low-dimensional manifold structure through the application of manifold learning methods and to explore potential uses of these methods in speech processing tasks. Our proposed approach may be summarised in four steps:

1. Preprocessing: Segment speech into appropriate units and apply some preprocessing prior to analysis.

2. Feature extraction: Compute high-dimensional feature vectors parametrising the chosen speech data.

3. Dimensionality reduction: Apply a method to reduce the dimensionality of these input feature vectors.

4. Evaluation: Examine the output low-dimensional feature vectors, evaluate the ability of the dimensionality reduction method to produce meaningful low-dimensional embeddings, and determine if there is inherent low-dimensional structure to the data.

In Step 1, the acoustic speech signal is segmented into appropriate units, for example phones, and a number of preprocessing procedures are performed. In Step 2, two different types of speech parametrisation are computed in order to compare the underlying structure of each. Features derived from the conventional magnitude spectrum, MFCCs, and less widely used phase spectrum features are computed to serve as high-dimensional input vectors for dimensionality reduction. This facilitates the study of the underlying structure and information contained in both the magnitude and phase spectrum.

Next, in Step 3, the following manifold learning methods are applied to the speech parametrisations produced in Step 2: locally linear embedding (LLE) (Roweis and Saul, 2000), isometric feature mapping (Isomap) (Tenenbaum et al., 2000), and Laplacian eigenmaps (LEM) (Belkin and Niyogi, 2002). In order to compare the performance of these methods with linear dimensionality reduction techniques, the classical PCA method is also applied to the speech parametrisations.

Having reduced the dimensionality of each speech parametrisation, the fourth step involves examining the low-dimensional representations output to ascertain if they contain

meaningful information. Meaningful representations retain the structure, sources of variability, and most significant features of the input data. In order to investigate the ability of each dimensionality reduction method to discover meaningful low-dimensional structure in speech we perform a number of experiments on their outputs. We show that the two- and three-dimensional visualisations resulting from these methods are useful in displaying and analysing key characteristics of speech data such as pitch, phone, and speaker variation. As a further means of measuring the performance of these methods the output low-dimensional features are used as feature vectors in a number of phone and speaker classification tasks.

The results of these experiments allow us to compare the performance of the four dimensionality reduction methods on the different input feature representations. Manifold learning methods are found capable of producing meaningful low-dimensional embeddings of speech data. Furthermore, compared with a classic linear dimensionality reduction method, PCA, manifold learning methods have the ability to retain more meaningful structure in very low dimensions. We claim that this shows that the low-dimensional manifold which speech occupies is nonlinearly embedded in high-dimensional space, with manifold learning methods able to exploit this nonlinearity while linear methods are not. The results of this approach to studying the dimensionality of speech support previous proposals such as those made by Togneri et al. (1992) and Jansen and Niyogi (2005).

Also, the low-dimensional structure of features derived from both the magnitude and phase spectrum is explored and both are shown to have similar low-dimensional structure. A framework for combining the complementary information of features derived from the magnitude and phase spectrum, without large increases in feature dimensionality, is proposed.

1.4 Contributions

This dissertation examines the hypothesis that speech has inherent low-dimensional manifold structure and that manifold learning methods are capable of extracting meaningful low-dimensional features representing the information communicated by the acoustic speech signal. The following is a summary of the main contributions of this dissertation:

- The capability of manifold learning methods to produce meaningful low-dimensional

representations of both synthetic and natural speech is demonstrated.

- For the first time, a number of manifold learning methods are applied to speech and their performance is compared. The Isomap algorithm is shown to produce the most meaningful low-dimensional representations of speech data.

- Comparisons of the output of manifold learning methods and PCA indicate that speech has an inherent low-dimensional structure, with a dimensionality of between two and six, nonlinearly embedded in high-dimensional space.

- Manifold learning algorithms are shown to be useful for feature transformation in phone classification systems.

- An approach to combine the complementary information of features derived from the magnitude and phase spectrum, without large increases in feature dimensionality and the associated computational complexity, is demonstrated.

- Dimensionality reduction methods are evaluated in speaker identification tasks. A manifold learning methods is shown to offer the best performance in low dimensions. However, for higher-dimensional feature vectors MFCC and PCA-transformed features are found to yield higher classification accuracy.

1.5 Thesis outline

The remainder of this dissertation is organised as follows:

Chapter 2 provides a discussion of relevant background information relating to the production of speech, the perception of speech, and phonetics.

Chapter 3 reviews previous studies of the dimensionality of speech, highlights established findings, and identifies worthwhile unexplored research topics.

Chapter 4 discusses dimensionality reduction and describes the four dimensionality reduction methods used in this work: PCA, Isomap, LLE, and Laplacian eigenmaps. The methods are compared and example applications of each method are provided. Previous applications of these methods to speech are also reviewed.

Chapter 5 presents our proposed approach to examining the underlying dimensionality of speech and evaluating the performance of the dimensionality reduction methods. The feature extraction process, dimensionality reduction procedure, and means of evaluation used are each discussed.

Chapter 6 details the application of the four dimensionality reduction methods in experiments on synthetic speech data. Results of visualisation and vowel classification experiments are reported and discussed.

Chapter 7 presents experiments carried out on natural speech data. Results of visualisation, phone classification, and speaker identification experiments are reported and discussed. Comparisons of features derived from the magnitude and phase spectrum are also reported.

Chapter 8 concludes and presents possibilities for future work.

1.6 Publications

Some of the work presented in this dissertation has previously been published in conference proceedings. The following publications resulted from research conducted for this dissertation: Errity and McKenna (2006, 2007); Errity et al. (2007a,b); Errity and McKenna (2009). The work presented in this dissertation is not the product of collaborative work, save and to the extent that such work has been cited.

Chapter 2

Overview of Speech Production, Perception, and Phonetics

This chapter provides an overview of background theory regarding the production of speech, the perception of speech, and phonetics.[1] Other aspects of speech theory, particularly with respect to feature extraction, are discussed in Chapter 5.

2.1 Speech production

Figure 2.1 illustrates the speech production apparatus. Speech production begins with the lungs providing a source of air, an excitation, that then flows up the trachea and through the vocal folds within the larynx. The vocal folds are two sets of layers of ligaments, tissue, and muscle that stretch horizontally across the larynx. These vocal folds may be tensed or relaxed. When tensed the flow of air through the gap between them, known as the glottis, causes them to open and close rapidly producing quasi-periodic pulses in the airflow that results in 'voiced' speech sounds. The fundamental frequency, f0,[2] of a spoken utterance is determined by the rate at which the vocal folds are opening and closing. Alternatively, if the vocal folds are relaxed the airflow through them will not be quasi-periodic, resulting in so called 'unvoiced' sounds.

The area from the vocal folds to the lips is called the vocal tract. The shape of this vocal

[1] For a more detailed discussion of the physics and biology of speech refer to Denes and Pinson (1993). Also, the interested reader may wish to refer to Quatieri (2002) for further information relating to speech signal processing.

[2] This fundamental frequency corresponds to the perceived tone of the speech signal which is called the 'pitch'. The two terms are used interchangeably in this work.

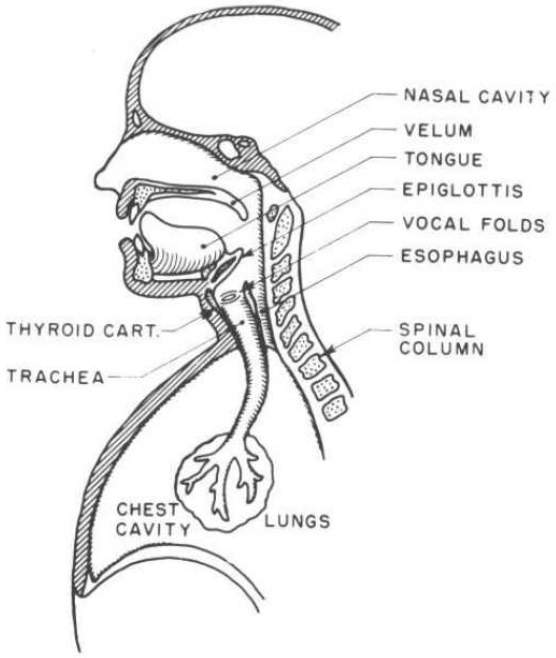


Figure 2.1: Schematic view of the human speech production mechanism, after Flanagan (1972).

tract can be changed to produce different sounds. This change is achieved by movement of the articulators, namely: the jaw, tongue, velum, and lips. The vocal tract can be thought of as an air-filled tube and, as such, has certain natural frequencies of vibration—resonances. These resonant frequencies are known as formants. As the shape of the vocal tract changes to produce different speech sounds, so too do the formant frequencies. Thus, each speech sound can be characterised by its formant frequencies.

2.1.1 Source-filter model

One common approximation of this production apparatus is known as the source-filter model, which is described in relation to synthetic speech generation in Chapter 6. This model is illustrated in Figure 2.2. This is a relatively simple model but it has been successfully applied in a large number of speech processing applications. There are two components of this model:

- The 'source' component represents the airflow from the lungs through the vocal folds.
- The 'filter' then shapes the spectrum of this source signal. In the case of vowel sounds the 'filter' represents the vocal tract. For other types of speech sound, such as

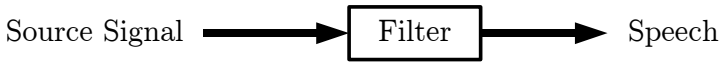


Figure 2.2: Source-filter model of speech production.

fricatives (discussed in Section 2.3), the 'filter' models the resonant cavity extending from a point of constriction in the vocal tract to the lips.

This model makes the assumption that the two components are linearly separable.

Linear prediction (Atal and Hanauer, 1971) is often used to estimate this model. Linear prediction gets its name from the fact that it predicts the current speech sample s_n as a linear combination of p past speech samples,

$$s_n = a_1 s_{n-1} + a_2 s_{n-2} + \ldots + a_p s_{n-p} + e_n \ , \tag{2.1}$$

where e_n is the excitation or glottal source signal, and the values

$$\mathbf{a} = [a_1, \ldots, a_p]' \ , \tag{2.2}$$

are the linear prediction coefficients (LPC) which characterise the filter response. A number of methods exist for estimating these LPCs such as autocorrelation analysis and covariance analysis. Using linear prediction is equivalent to modelling the vocal tract by an all-pole filter $L(z)$ with p poles,

$$L(z) = \frac{G}{1 - \sum_{i=1}^{p} a_i z^{-i}} = \frac{G}{A(z)} \ , \tag{2.3}$$

where G is the gain term which controls the energy in the signal.

Various types of source signal, $\mathbf{e}$, can be used in this model. In basic approaches a periodic pulse train is used to simulate the glottal source for voiced speech and a random noise signal is used as a substitute for the turbulent glottal airflow for unvoiced speech. More complex models of the glottal source signal are also possible, such as the LF-modelled glottal pulse train (Fant et al., 1985) discussed in further detail in Section 6.2.

2.2 Speech perception

When studying speech it is also desirable to have an understanding of the apparatus responsible for receiving the speech signal. As shown in Figure 2.3, the human ear can be viewed as three distinct areas, described briefly below:

- Outer ear: This consists of the external part of the ear located outside the head, known as the pinna, and the external canal. The purpose of the outer ear is to funnel sound into the middle ear.
- Middle ear: This area of the ear converts acoustic sound waves into mechanical vibrations that travel via the oval window to the inner ear.
- Inner ear: This is a fluid-filled chamber containing the cochlea, which is shaped like a snail, and basilar membrane. The vibrations at the oval window create standing-waves in the fluid which in turn vibrate tiny hairs on the basilar membrane. The frequencies of these vibrations are related to the frequencies present in the original acoustic sound wave. The hairs, called stereocilia, are connected to the auditory nerve and essentially convert the mechanical vibrations into electrical nerve impulses. However, the sensitivity of the basilar membrane to frequency is nonlinear; its frequency resolution decreases as frequency increases. A number of approaches have been proposed to account for this nonlinear frequency response; for example, the mel scale described below.

2.2.1 Mel scale

The mel scale is based on perceptual experiments (Stevens and Volkman, 1940) that have shown that the human auditory system is more sensitive to differences between frequencies in low frequency ranges, below 1 kHz, than in higher frequency ranges. Thus, the mel scale is approximately linear below 1 kHz and logarithmic above this. This scale is commonly approximated as,

$$B(f) = 1125 \ln(1 + f/700) \ , \tag{2.4}$$

where f is the frequency in Hz. A plot comparing the mel and Hertz scales is shown in Figure 2.4.

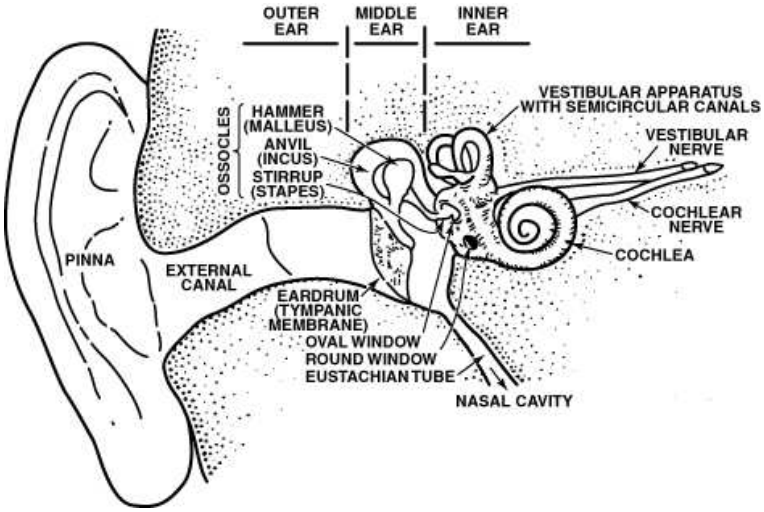


Figure 2.3: Schematic view of the human ear (not to scale), after Flanagan (1972).

2.3 Phonetics

As discussed in Section 2.1, the configuration of the articulators can be modified to produce a range of different vocal tract shapes. These vocal tract configurations are then combined with the source signal—voiced or unvoiced—to produce a particular speech sound. Each word in a language is made up of some sequence of these individual speech sounds. The term phoneme is used to refer to the concept of a distinct speech sound within a language. Phonemes can thus be used to distinguish one word from another. Actual, uttered, speech sounds are individually referred to as phones. Several different phones, physically produced speech sounds, may belong to the same phoneme. Throughout this dissertation the word phone is used to describe a meaningful, distinct sound unit.

The field of phonetics involves the study of human speech sounds and their production. This field has produced various sets of symbols that can be used to refer to phones. Table 2.1 gives a list of two such symbol sets: one from the International Phonetic Alphabet (IPA); and the other from TIMIT (Garofalo et al., 1990), a widely used corpus of speech recordings. Descriptions of phone articulation and English language examples are also provided. Table 2.1 lists only those phones used in the experiments described in this dissertation.

Phones can be grouped into broad categories based on the manner in which they are produced. Phone categories, corresponding to those used in TIMIT, are listed in Table 2.1. A brief description of each of these five phone categories is provided below:

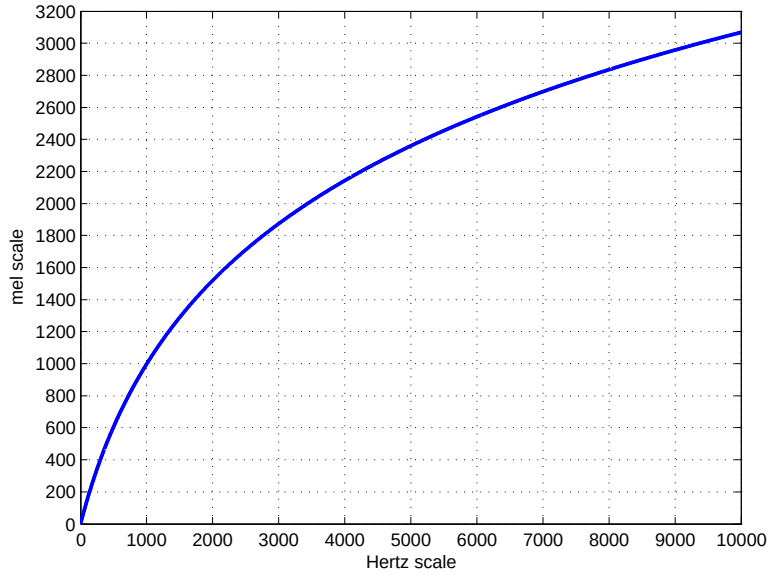


Figure 2.4: Mel scale vs. Hertz scale.

- Vowels: This is the largest category of phones in the English language. Vowels are sounds produced when the vocal folds are opening and closing at some particular fundamental frequency and there is no narrow point of constriction in the vocal tract.

- Fricatives: These are sounds produced when air is forced to flow through some constriction in the vocal tract. For example, a narrow channel formed by placing the lower lip against the upper teeth. Fricatives can be voiced or unvoiced. For voiced fricatives the vocal folds are tensed and hence vibrating. In the case of unvoiced fricatives the vocal folds are relaxed and turbulent airflow results.

- Stops (or plosives): Sounds resulting from a build up of pressure at some point in the vocal tract followed by a sudden release. The build up of pressure may occur at the lips, teeth, or velum. As with fricatives, stops can be voiced or unvoiced.

- Nasals: Describes sounds in which the velum is lowered and air flows through the nasal cavity. As with vowels, the source is quasi-periodic.

- Semi-vowels/glides: Transitional sounds that are difficult to categorise. They are vowel-like as they have a quasi-periodic source. However, the constriction in the vocal tract is greater than in the case of vowels.

TIMIT	IPA	Category	Description	Example
aa	ɑ	vowel	open front unrounded vowel	b***o***b
iy	i	vowel	close front unrounded vowel	b***ee***t
uw	u	vowel	close back rounded vowel	b***oo***t
eh	ɛ	vowel	open-mid front unrounded vowel	b***e***t
ae	æ	vowel	near-open front unrounded vowel	b***a***t
ah	ʌ	vowel	open-mid back unrounded vowel	b***u***t
ih	ɪ	vowel	near-close near-front unrounded vowel	b***i***t
ax	ə	vowel	mid central vowel	***a***bout
ow	o	vowel	close-mid back rounded vowel	b***oa***t
ao	ɔ	vowel	open-mid back rounded vowel	b***ou***ght
s	s	fricative	voiceless alveolar fricative	***s***ea
sh	ʃ	fricative	voiceless postalveolar fricative	***sh***e
p	p	stop	voiceless bilabial plosive	***p***ea
t	t	stop	alveolar plosive	***t***ea
k	k	stop	voiceless velar plosive	***k***ey
m	m	nasal	bilabial nasal	***m***oon
n	n	nasal	dental or alveolar nasal	***n***oo***n***
l	l	semivowel/glide	alveolar lateral approximant	***l***ay
y	y	semivowel/glide	close front rounded vowel	***y***acht

Table 2.1: TIMIT and IPA phonetic symbols. Examples of the corresponding phones are indicated in bold.

Chapter 3

Speech Dimensionality: Literature Review

The underlying dimensionality of speech has long been the subject of research. The constrained movement of the articulators and limited set of sounds employed in human spoken communication have motivated many researchers to investigate the possibility that a small number of variables can be used to describe the speech system. This chapter reviews existing literature concerning the dimensionality of speech.

In the past, investigators have used numerous methods to study the dimensionality of the space occupied by speech. In the first section of this chapter, classical phonetic and acoustic studies that motivated a low-dimensional view of speech are described. Following this, previous work in this area is discussed in four distinct sections: analyses conducted on conventional magnitude spectrum representations, nonlinear dynamical system analysis based studies, manifold learning motivated approaches, and investigations of nonacoustic speech signals. Finally, conclusions based on this literature review are presented.

3.1 Classical studies

Evidence that the speech production system can be modelled by a small number of parameters can be found in numerous studies dating from classic early works in the field of speech analysis to present day research. For example, phoneticians have long been proposing that only a small number of parameters are necessary to describe the speech articulation process. As early as the mid-nineteenth century, Alexander Melville Bell de-

veloped the first universal notation system to describe individual speech sounds. This pioneering work originated the description of vowel sounds in terms of the position of the tongue during articulation. Bell (1867) categorised vowels in terms of two dimensions: the frontness and height of the tongue. The work of Sweet (1877) further advanced and popularised this model.

In the early twentieth century this two-dimensional articulatory phonetic view of vowels was further refined by Daniel Jones. Jones represented vowels on a two-dimensional quadrilateral-shaped plane, with vowels organised between the two most extreme positions of the highest point of the tongue during articulation: high-front and low-back. A number of reference, or cardinal, vowels were defined by Jones in 1918 to constitute reference points on this plane (Jones, 1964). The corners of the vowel quadrilateral equate to the vowels produced when the tongue is positioned at the most extreme points of articulation. Distances between vowels on the quadrilateral are designed to equate auditory and articulatory differences. This model allows vowels to be easily classified and compared based on just two measurements of the articulators.

The widely used IPA uses a version of Jones' model. The IPA vowel chart (IPA, 1999), illustrated in Figure 3.1, clearly depicts the articulation of a number of different vowel sounds with respect to the two dimensions of tongue frontness and height; where vowels appear in pairs, the left and right vowels are produced with rounded and unrounded lips, respectively. The degree of lip rounding during vowel articulation is often considered as a third dimension to the conventional vowel plane (Ladefoged, 1967, p. 140).

The work of these phoneticians resulted in a representation of the physiological processes of vowel articulation in a low-dimensional space, based on the constrained movement of the articulators. These early studies helped motivate the view that speech is inherently low-dimensional. Apart from some references to previous work, this articulatory phonetic approach is not considered further in this dissertation.

In addition to these studies of articulatory phonetics, early investigations of the spectral content of acoustic speech signals have also indicated low-dimensional structure in speech. One such classical study is the analysis of American English vowels conducted by Peterson and Barney (1952). Peterson and Barney measured the first three formant frequencies from the spectrograms of 10 vowel sounds uttered twice by 76 men, women,

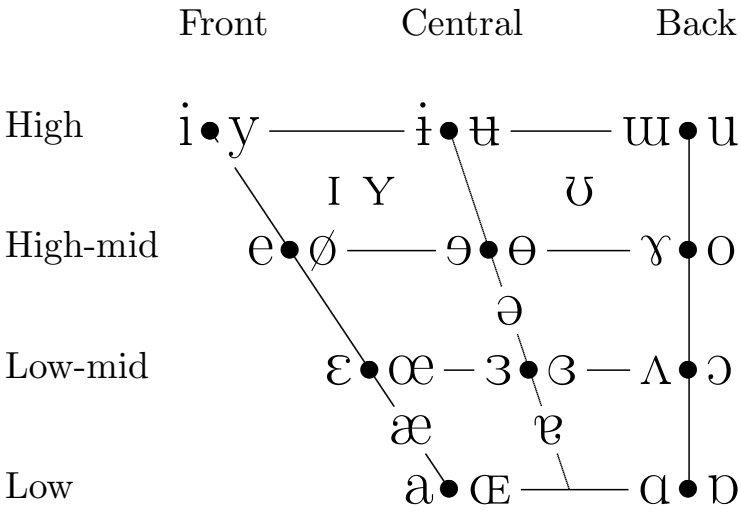


Figure 3.1: IPA vowel chart (IPA, 1999).

and children. They found a correspondence between the vowel type uttered and position in the two-dimensional space formed by plotting the first formant frequencies (F1) versus second formant frequencies (F2). The vowels were found to be clustered in this two-dimensional space with their location dependent on their articulation. This F2/F1 vowel space corresponds closely to Jones' cardinal vowel chart, with F2 and F1 correlated with tongue frontness and lowness, respectively. These results are depicted in Figure 3.2 which shows the mean F1 and F2 measurements of 10 vowels for the male recordings from the Peterson and Barney (1952) vowel data.

The results of these early studies provide clear and logical arguments towards a low-dimensional view of vowel sounds. Following these studies a large number of investigations, utilising a range of methods, have been conducted into the dimensionality of vowels and other classes of speech sounds. In the remainder of this chapter, we present a review of existing literature describing these investigations.

3.2 Analyses of magnitude spectrum based feature representations

Speech is commonly parametrised using a magnitude spectrum based feature representation that describes the frequency content of the speech signal while discarding information from the phase spectrum. This section presents previous studies of speech dimensionality

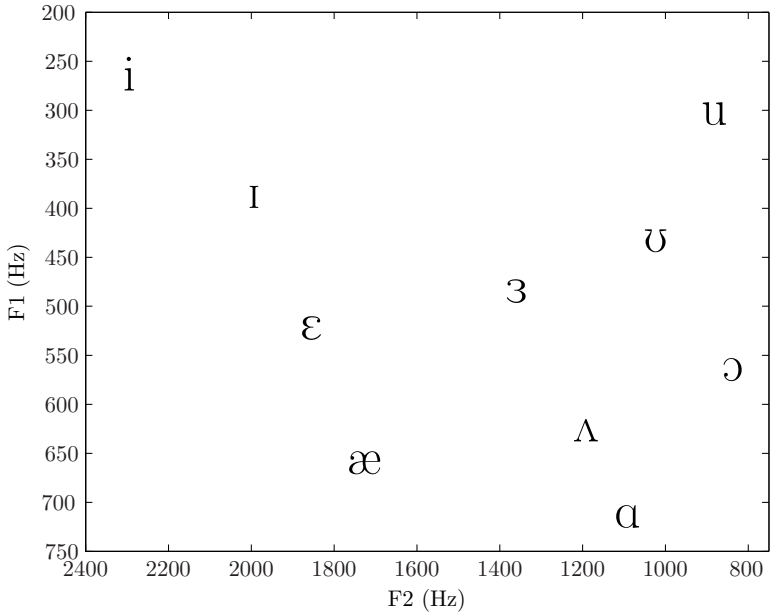


Figure 3.2: Mean first (F1) vs. second (F2) formant frequencies of 10 vowels recorded by 33 male speakers. The data is taken from Peterson and Barney (1952).

conducted on such spectral parametrisations. These previous studies are grouped in terms of the dimensionality analysis approach used. Dimensionality reduction based approaches, studies of inherent dimensionality, and measures of fractal dimension are discussed.

3.2.1 Dimensionality reduction analyses

For many decades researchers have been conducting studies examining the dimensions of the speech space. A large number of these studies share a similar methodology. First, a number of speech recordings are obtained and preprocessed, resulting in the set of sounds the investigators wish to analyse. Second, a high-dimensional feature representation is extracted from short-time frames of the speech signals, based on the conventional assumption that the speech signal is stationary over intervals of 10–40 ms. This high-dimensional feature representation is commonly a parametrisation of the signal's spectral content, often derived from the signal's Fourier transform. Finally, the underlying dimensionality of the data set consisting of these high-dimensional representations is analysed. This is often accomplished by applying methods which attempt to reduce the dimensionality of the data while retaining significant information. Studying the amount of information lost while varying the numbers of dimensions can reveal characteristics of the speech data's underlying structure.

One of the early applications of this analysis methodology was reported by Plomp et al. (1967). In this work, Plomp et al. conducted an analysis of the frequency spectra of 15 Dutch vowels uttered by 10 speakers. The output of 18 bandpass filters provided a high-dimensional representation of the vowels. Applying PCA (Jolliffe, 1986) to this data they found that only four dimensions were needed to describe the vowel data. Of these four dimensions, they found that the first two dimensions accounted for 68.4% of the total variance in the vowel data. Plotting the vowels in this two-dimensional space revealed a configuration similar to the classical F2/F1 vowel plane shown in Figure 3.2. The third and fourth dimensions were found to account for 15.7% of the total variance in the vowel data but Plomp et al. did not find a direct relation between these dimensions and the formant frequencies. Plomp et al. concluded that the first two dimensions are related to the frequencies of the first two formants and the differences between vowel spectra can be described by four independent parameters. Similar studies conducted subsequently by Li et al. (1968), Boehm and Wright (1968), and Favella et al. (1969) supported these conclusions.

Pols et al. (1969) further developed this work by investigating the relationship between the physical characteristics of vowel sounds and the way that they are perceived. Listening tests were performed using 15 subjects to rank the perceptual similarity of 11 vowel sounds. MDS (Cox and Cox, 2001) was applied to the resulting similarity matrix to produce a three-dimensional perceptual space. This perceptual space was compared to a physical space, produced by PCA in a similar fashion to that described above (Plomp et al., 1967), and the two spaces were found to be highly correlated. These results indicated that not only are three or four factors capable of adequately discriminating between vowels, but these factors correspond to both formant frequencies and perceptual evaluations by listeners. Furthermore, Pols et al. (1969) suggested that other classes of speech sounds, namely nasals and liquids, can also be represented in a low-dimensional space.

The approaches of Plomp et al. (1967) and Pols et al. (1969) were combined and a more detailed examination performed in a study by Klein et al. (1970). This article describes results for vowel data from an increased number of speakers: 50 as opposed to 10 previously. This work corroborated the findings of the previous studies and resulted in improved vowel classification results. Using the first four components resulting from

PCA, 98% phone classification accuracy was achieved. This classification experiment used a speaker normalisation procedure first described by Gerstman (1968).

The 50 speaker data used by Klein et al. (1970) was later used in a study by Pols et al. (1973) comparing the merits of a PCA-derived vowel plane and the classical F1/F2 formant plane. They found that vowel classification scores in the two spaces were comparable but the PCA method offered the advantages of greater simplicity and less computational complexity.

Building on these dimensional analyses of speech, Pols (1971) presented a system to perform real-time word recognition based on the methods described in the literature reviewed above. This system was capable of achieving high classification rates, 98.8%, for a 20 speaker, 20 word task using just three-dimensions computed by PCA. This shows that discriminatory information regarding the state of the speech production system can be adequately described using only a small number of dimensions.

3.2.2 Estimation of inherent dimensionality

In addition to the analyses described above, which use dimensionality reduction methods to examine the underlying structure of speech, a number of investigations have applied methods that aim to directly measure the inherent dimensionality of speech data. Inherent dimensionality may be defined as the minimum number of parameters needed to account for the properties of the data (Fukunaga, 1990). The concept of inherent dimensionality is illustrated in Figure 3.3 which gives an example of data with an inherent dimensionality lower than the space in which it is represented. This data is intrinsically one-dimensional but is nonlinearly embedded in two-dimensional space. Inherent, also referred to as intrinsic, dimensionality estimation is a well studied problem in the field of pattern recognition (Camastra, 2003).

In an effort to determine the inherent dimensionality of speech, Tattersall et al. (1990) have applied to speech the self-organizing map (SOM) algorithm—a type of artificial neural network developed by Kohonen (1995). The aim of Kohonen's SOM is to produce a low-dimensional representation of the input data while preserving topological structure. Thus, the algorithm reduces dimensionality similarly to PCA, but preserves a very different kind of structure—namely, the neighbourhood relationships between data points. Tattersall

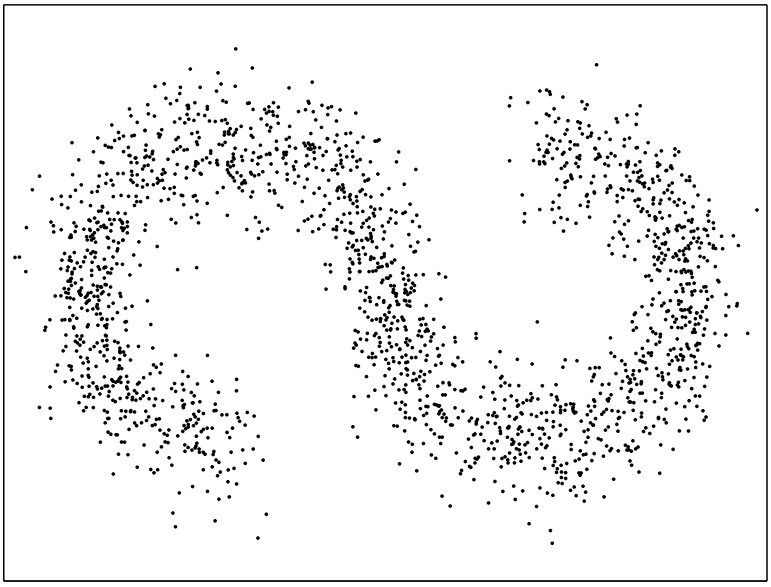


Figure 3.3: Inherently one-dimensional data nonlinearly embedded in two-dimensional space.

et al. proposed that the inherent dimensionality of vowels is two, based on the application of a two-dimensional SOM neural array to filter-bank output representations of vowels.

This proposal has been disputed in a number of studies conducted by Togneri, Alder, and Attikiouzel (Togneri et al., 1990; Alder et al., 1991; Togneri et al., 1992). They argue that the Kohonen algorithm does not produce a measure of the inherent dimensionality of a data set and that, in any case, this measure would be greater than two for speech data. Their view is supported by the dimensional analyses described in Section 3.2.1 which suggest that up to four significant dimensions may be necessary to adequately describe the speech space. They proposed a means of estimating the inherent dimensionality of a data set by using a Kohonen algorithm to fit various d-dimensional grids to the data points and measuring how well these grids fit the data. This measurement is obtained by calculating the mean absolute curvature of the grid. This value is then compared for varying grid dimensionalities, d, and is expected to be close to zero when the grid fits the data well, thus indicating the data's inherent dimensionality. Togneri et al. (1990) and Alder et al. (1991) have applied this approach to both fast Fourier transform (FFT) and LPC representations of frames extracted from continuous speech recordings. Both 12- and 16-dimensional representations were tested. The number of frames and speakers used were also varied. These variations were not found to affect the resulting inherent dimensionality estimates. These studies concluded that the speech space has an inherent

dimensionality of at least four.

Togneri et al. (1992) further refined these findings and concluded that the speech space may be approximated by a grid of three- or four-dimensions. They also found "significant nonlinearities" in the embedding of this low-dimensional manifold in high-dimensional, FFT and LPC, space. It is important to note that these investigations examined the entire speech space rather than a single phone class, i.e. vowels as in previous studies, and found no evidence that different phone classes have different dimensionalities.

In addition to applying the Kohonen algorithm as detailed above, Alder et al. (1991) also applied a number of other intrinsic dimensionality estimation algorithms (Glover, 1989; Judd, 1989) to their speech data. These experiments yielded a mean estimate of 3.4 for the dimension of speech space, consistent with previously reported findings.

A further study worthy of discussion was conducted by Baydal et al. (1989) who applied an inherent dimensionality estimation procedure, developed by Pettis et al. (1979), to speech data. The objective of this study was to investigate the dimensionality of whole vocabularies of utterances, with a view to isolated word recognition tasks. This is in contrast to the objective of this dissertation and the previous studies described in this chapter—to examine the dimensionality of the speech space. While we are interested in the underlying structure of speech, Baydal et al. attempted to measure the inherent complexity, or difficulty, of different vocabularies of utterances. For example, a vocabulary consisting of the words 'cat' and 'dog' would intuitively be expected to be simpler, and have a lower inherent dimensionality, than a large varied vocabulary. Baydal et al. found the vocabularies examined to have inherent dimensionalities ranging from 3–13, with the dimensionality increasing with the size and difficulty of the vocabulary. These estimates are consistent with previous findings regarding the lowest number of parameters capable of representing speech. However, differences in the objectives and methods used prohibit direct comparisons between this study and previously discussed studies.

Somervuo (2003b) investigated the dimensionality of 25 speech utterances represented by both 12-dimensional MFCC features and 26-dimensional log mel-spectrum features. He applied curvilinear component analysis (Demartines and Herault, 1997), an MDS-based dimensionality reduction method, to embed the speech feature vectors into spaces of varying dimension in order to estimate the speech data manifold's intrinsic dimensionality.

This estimate was based on measuring the difference in the distance between points in the original feature space and the distances in the low-dimensional spaces. No clear indication of intrinsic dimensionality could be found. This may be due to measuring distances not along the data manifold, but rather in the feature space, a problem that the manifold learning methods, discussed in the remaining chapters, are designed to overcome.

The main body of Somervuo's work was the application of SOMs to speech data in order to produce a low-dimensional space in which speech feature *trajectories* are well represented. This is in contrast to the static features which have been examined in the studies discussed thus far. The dimensionality of the SOMs was varied between two and six—the maximum indicated by the intrinsic dimensionality analyses. SOMs with dimensions between three and six were found to adequately represent speech trajectories, with five dimensions found to be best.

3.2.3 Fractal dimensions

Thus far, our discussion has been solely in terms of the classic intuitive view of dimensionality, that is, the integer number of parameters required to describe a point on an object in space. Common examples of this are a point having zero-dimensions, lines and curves being one-dimensional, and a cube being three-dimensional. This view can be generalised by the concept of d-dimensional Euclidean space, $\mathbb{R}^d$. The term 'topological dimension' is used to describe this type of dimensionality. However, the notion of dimensionality is not unique and many different definitions have been defined by mathematicians. In fact, it is possible for the dimensionality of a set to have several different numerical values depending on the definitions used.

At this point it is necessary to introduce another such definition of dimensionality that has also been used in relation to speech—'fractal dimension'. The notion of 'fractal dimensions' allows for objects with a noninteger number of dimensions. These complex geometric objects are described by the term 'fractals', coined by Mandelbrot (1983), and have a fractal dimension exceeding their topological dimension. At first, it may be difficult to intuitively understand how an object can have a noninteger dimension and thus lie somewhere between two integer dimensions. An example of a fractal, a Koch curve (Schroeder, 1991, Chapter 1), is shown in Figure 3.4. This geometric object is generated by beginning

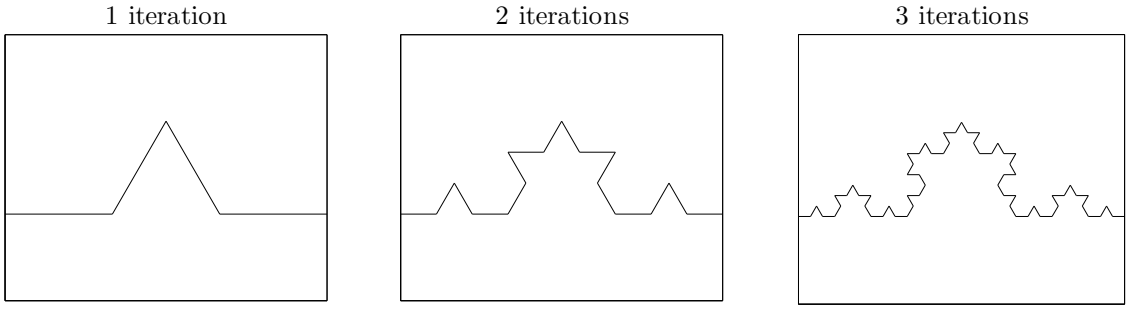


Figure 3.4: Koch curve fractal. The first, second, and third iterations of the Koch curve are shown. The complexity and length of the curve can be seen to increase as the number of self-similar components are increased.

with a single line and recursively performing the following iterative procedure:

1. Divide the line into three equal length segments.

2. Replace the middle segment with two new lines, each equal in length to the original segments, forming an equilateral triangle with the now replaced middle segment.

3. Repeat for each resulting line segment, as shown in Figure 3.4.

This results in a curve which, as stated previously, has a topological dimension of one. However, the Koch curve has infinite length as each iteration of the above procedure increases its length. Also, given infinite iterations, the structure of a Koch curve does not approach a line at any level of magnification. The intuitive topological dimension does not take this complexity into account and this is why fractal dimensions are necessary. The fractal dimension of the Koch curve is approximately 1.26.[1] This is larger than its topological dimension, indicating that the fractal dimension can account for the Koch curve's added complexity.

One of the most popular measures of fractal dimension is the correlation dimension, which is commonly calculated using an approach proposed by Grassberger and Procaccia (1983). The correlation dimension, given a set of N points $\{\mathbf{x}_i,\ \ i = 1, \ldots, N\}$, is defined as

$$D_c = \lim_{r \to 0} \frac{\log C(r)}{\log r} \ , \tag{3.1}$$

[1]The total length of the Koch curve increases by one third after each iteration—the middle of three equal length segments is replaced by two more segments of the same length, thus an increase from three to four segments. The Hausdorff dimension, a standard means of measuring an objects fractal dimension, of a Koch curve is calculated as $\log(4)/\log(3) = 1.26\ldots$ (Schroeder, 1991, Chapter 1).

where

$$C(r) = \lim_{N\to\infty} \frac{2}{N(N-1)} \sum_{j=1}^{N} \sum_{i=j+1}^{N} H(r - \|\mathbf{x}_i - \mathbf{x}_j\|) \ . \tag{3.2}$$

The Heaviside function, $H(p)$, is defined as

$$H(p) = \begin{cases} 0, & p < 0 \\ 1, & p \geq 0 \ . \end{cases} \tag{3.3}$$

Thus, this function counts the number of pairs of points with distance less than some r between them. In practice, using the Grassberger and Procaccia (1983) approach, D_c is estimated by plotting $\log C(r)$ against $\log r$ and measuring the slope of the linear portion of the graph. This is a very popular method, however there exist several practical problems concerning its implementation (Theiler, 1988).

Somervuo (2003b) investigated the correlation dimension of 25 speech utterances represented by both 12-dimensional MFCC features and 26-dimensional log mel-spectrum features. He found the correlation dimension to be between 5.0–5.7 for these utterances. However, as noted by Somervuo, this approach measures distances in feature space, not geodesically along the underlying manifold. Thus, a highly curved or folded manifold may have a lower dimensionality than that indicated by its correlation dimension estimate. Manifold learning methods, such as those applied in this dissertation, allow for such structure and hence may more accurately model the inherent low-dimensionality.

Fractal dimensions are an important concept in the study of chaos[2] and the behaviour of nonlinear dynamical systems. The application of ideas from these fields to speech and the resulting contributions to our knowledge of the dimensionality of the speech system are discussed in the following section.

3.3 Nonlinear dynamical systems approaches

Traditionally in speech signal processing, simplifying assumptions are made of the speech production process in order to describe it using a linear source-filter model. However, there exists a large body of research (Teager and Teager, 1990; Casdagli, 1991; Barney et al.,

[2]There is no agreed upon definition of chaos; however, the following definition effectively describes the characteristics of a chaotic system: "Chaos is aperiodic time-asymptotic behaviour in a deterministic system which exhibits sensitive dependence on initial conditions." (Fitzpatrick, 2006).

1999) suggesting the presence of nonlinear processes, such as source-filter coupling and nonlaminar airflow, at work during speech production. This has lead many researchers to apply methods from the fields of nonlinear dynamical and chaotic systems to speech in an effort to yield a better model of the nonlinear behaviour of the speech system than the traditional linear source-filter approach. This research relates to our work and the previous studies described in this chapter as nonlinear dynamical systems analysis methods can provide information regarding the degrees of freedom and underlying low-dimensional structure of speech.

Before discussing related studies using these approaches we shall provide a brief introduction to the theory and methods of nonlinear dynamical systems analysis for the interested reader.[3] These approaches assume speech is a nonlinear dynamical system and that the system's state can be described by a number of hidden dynamic variables $\mathbf{x}_n$, where $n = 1, \ldots, N$ represents time. The observable scalar speech signal s can then be viewed as a one-dimensional measurement of the systems underlying state. Given this time series signal, the state space of the system—the space of the hidden dynamic variables—can be reconstructed using Takens' (1981) time delay embedding method (Abarbanel, 1996; Sauer et al., 1991). This method can be implemented by sliding a window through the signal. This window is of length d_e, this is referred to as the embedding dimension in contrast to the dimension of the actual system, d. Thus, for a speech signal

$$\mathbf{s} = [s_0,\, s_1,\, s_2,\, \ldots,\, s_i,\, \ldots]' \ , \tag{3.4}$$

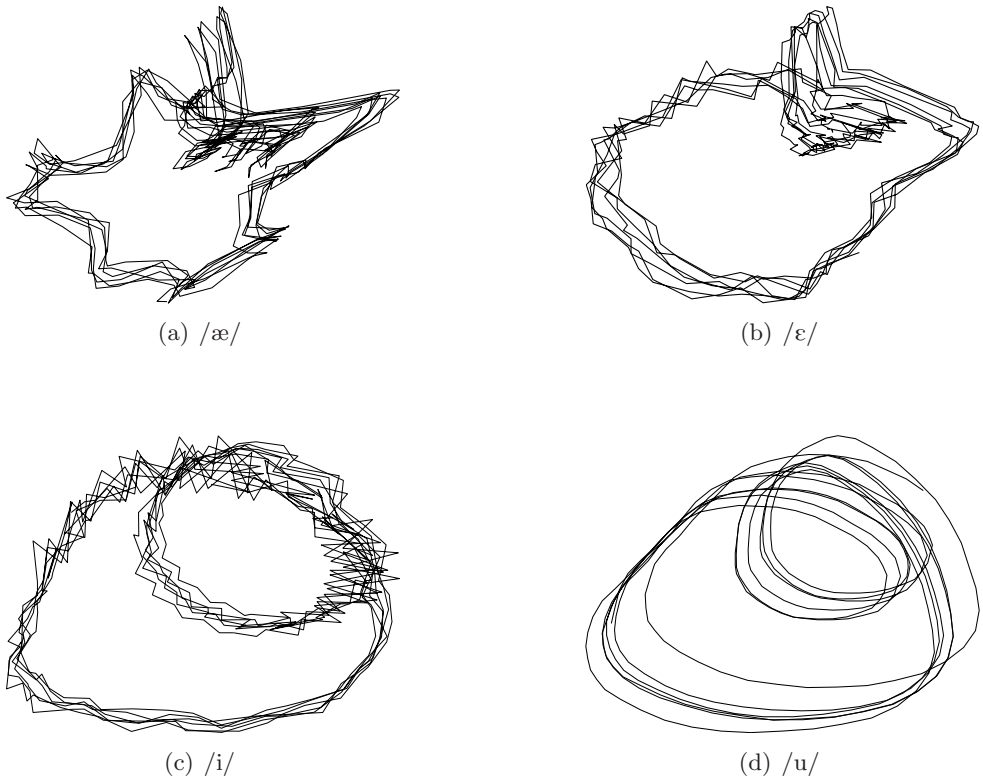
the reconstructed d_e-dimensional state space Y is formed using the window

$$\mathbf{y}_n = [s_n,\, s_{n+\tau},\, s_{n+2\tau},\, \ldots,\, s_{n+(d_e-1)\tau}]' \ , \tag{3.5}$$

where τ is the number of samples of delay.

If τ and d_e are chosen appropriately (Abarbanel, 1996; Pitsikalis et al., 2003) the resulting reconstructed state space embedding of the speech signal will preserve properties of the original state space. This enables analysis of the geometrical structure of the original state space of speech sounds. Examples of time delay embeddings for four different

[3] For a more detailed treatment of nonlinear dynamical systems, related analysis methods, and their application to speech, refer to the work of Banbrook (1996) and Mann (1999).



(a) /æ/ (b) /ɛ/

(c) /i/ (d) /u/

Figure 3.5: Two-dimensional time delay embeddings of the vowels /æ/, /ɛ/, /i/, and /u/; $\tau = 1.25\,\text{ms}$.

sustained vowel utterances are shown in Figure 3.5. These were generated by applying the time delay embedding method (3.5) to the sustained vowel sounds /æ/, /ɛ/, /i/, and /u/ uttered by a male speaker. The time delay, τ, was set equal to 1.25 ms. It can be seen that there is low-dimensional structure present. Analysis of the equivalent three-dimensional time delay embedding of these sounds found that the trajectories do not cross, though they may appear to, given the two-dimensional limits of this presentation medium. Given this low-dimensional structure, measurements can be made as to the degrees of freedom of the system, e.g. the correlation dimension, and how chaotic it is in nature, e.g. Lyapunov exponents (Banbrook, 1996).

Further to the studies described in the previous sections of this chapter, many researchers have investigated the dimensionality of speech using these nonlinear dynamical systems methods. One of the earliest investigations was conducted by Tishby (1990), who computed reconstructed state spaces for 20 segments of voiced speech and found the correlation dimension to be between three and five. He also estimated the correlation dimension of unvoiced speech as between five and eight, however he considered this estimate unreli-

able due to insufficient samples. A further study conducted by Kumar and Mullick (1990) reported similar results, with a correlation dimension estimate of less than three for vowels and stops, and greater than five for most fricatives. These findings are further supported by the work of Townshend (1991) who estimated the correlation dimension using over 30 s of normally spoken speech as 2.9.

McLaughlin and Lowry (1993) analysed three vowel sounds and found the correlation dimension to be higher, between three and five. They concluded that the results were inconclusive due to an insufficient number of sounds and speakers, but suggested that vowels have an underlying two- or three-dimensional manifold structure. A more detailed study analysing a larger number of prolonged vowel utterances was later conducted by Banbrook and McLaughlin (1994). They found a correlation dimension varying between one and three, supporting their previous suggestion. Interestingly, the authors suggest a link between the place of articulation, i.e. position in formant space (Figure 3.2) and on the IPA vowel chart (Figure 3.1), and the correlation dimension. This work was further extended with the application of singular value decomposition (SVD) to account for noisy data and produce smoother, cleaner speech time delay embeddings (Banbrook, 1996; Banbrook et al., 1999). This helped clarify dimension estimates, with the speech system found to be as low as three-dimensional.

Narayanan and Alwan (1995) conducted a study of fricative sounds and found correlation dimension estimates ranging from 3 to 7.2. They compared the results with vowels which they found to be lower-dimensional.

The generalised fractal dimension, an alternative measure of the number of degrees of freedom of a system, of speech was explored by Pitsikalis et al. (2003). They estimated the dimensionality of a number of phonemes taken from the TIMIT corpus. They found the generalised fractal dimension of vowels and stops to vary between approximately one and four, while they found a dimension between one and eight for fricative sounds. Pitsikalis et al. also used these dimension estimates as a component in a feature vector for phoneme recognition and have shown the features possess discriminative ability.

3.4 Manifold learning motivated approaches

A number of manifold learning related studies of speech data have been conducted by other researchers in parallel to, but independently of, the work we have performed and report in this dissertation. One such study was conducted by Jansen and Niyogi (2005, 2006) who recently presented a derivation of a class of approximate vowel sound manifolds, using a traditional concatenated tube model of the articulatory system, and showed that the manifold assumption is true of speech data. Motivated by this, Jansen and Niyogi (2005, 2006) also proposed what they call 'intrinsic Fourier analysis', a modified version of the Laplacian eigenmaps algorithm, described in Section 4.3.2, designed to exploit the manifold structure of speech and produce an 'intrinsic spectrogram', that is, the equivalent projection of the traditional 'extrinsic Fourier spectrogram,' onto the low-dimensional manifold.

The work of Tompkins and Wolfe (2009) builds on that of Jansen and Niyogi, adapting the 'intrinsic Fourier analysis' approach to work on larger data sets by overcoming a number of computational difficulties present in the original approach. Tompkins and Wolfe (2009) also perform phone classification experiments using features derived from the 'intrinsic spectrogram'. These experiments use a feed forward neural network to identify three vowels—'iy', 'ao', and 'ae'—showing that the 'intrinsic spectrogram' is capable of compressing important information relating to phone classification into just a few dimensions.

3.5 Studies of nonacoustic speech signals

All of the studies discussed thus far have been concerned with analysing the acoustic speech signal output at the lips. The dimensionality of speech has also previously been approached from nonacoustic perspectives, for example in the work of Carreira-Perpiñán and Renals (1998). In this study the investigators applied dimensionality reduction methods to electropalatographic[4] (EPG) representations of speech data. The linear methods of factor analysis (Bartholomew, 1987) and PCA (Jolliffe, 1986), and the nonlinear gen-

[4]Electropalatography is a procedure for measuring the timing and position of contact between the tongue and hard palate. The technique is relatively noninvasive, utilising an array of sensors placed on the hard palate.

erative topographic mapping (Bishop et al., 1998) method, an alternative to SOMs, were applied to a corpus of EPG data. The nonlinear method was found to outperform the linear methods. The authors concluded that, given this nonlinear method, the EPG data may be modelled using only a small number of parameters, with representations as low as two dimensions proving useful. While the EPG signal is not a complete description of the speech production system, as it does not account for details such as rounding and nasalisation, these results support the proposal that speech is intrinsically low-dimensional and suggest that a complex nonlinear model may be necessary to discover this underlying structure.

The dimensionality of electroglottographic[5] (EGG) signals has also been the subject of research. Behrman (1999) conducted a comparison of the fractal dimension of the EGG signals of healthy and pathological subjects. Behrman consistently found a dimension of three for the healthy speakers in contrast to inconsistent estimates for the pathological speakers. However, the EGG signal only contains information regarding the voice source and thus does not describe the complete speech system.

3.6 Conclusions

This chapter has provided a review of existing literature concerning the dimensionality of speech. A number of differently motivated approaches have been reviewed, all of which aim to provide information regarding the, possibly nonlinear, low-dimensional structure of speech. Previous results have been somewhat inconsistent; this may be due to factors such as differing motivations, sources of speech data, and feature extraction procedures. However, the general consensus of the studies reviewed in this chapter is that speech has intrinsic low-dimensional structure. A number of these studies have pointed to the possible nonlinear embedding of this structure in high-dimensional space. In the following chapters we describe our work—applying manifold learning methods to speech—which builds upon existing knowledge and attempts to exploit this possibly nonlinear embedding in order to discover the underlying low-dimensional structure in speech.

[5] An electroglottographic signal measures changes in the electrical resistance across the larynx using electrodes placed on the neck. This signal provides information on vocal fold activity.

Chapter 4

Dimensionality Reduction

This chapter provides an introduction to dimensionality reduction, introduces the concept of the curse of dimensionality, and details a number of methods to reduce the dimensionality of a data set.[1] These methods are applied in subsequent chapters.

Dimensionality reduction methods can be categorised as either linear or nonlinear, and we group and discuss the methods under these two categories. In order to demonstrate and compare the capabilities of these methods, we apply them to a number of data sets, including both toy examples and real-world image data, and present the results. Previous applications of these dimensionality reduction methods to speech are also reviewed. In conclusion a summary of these methods and our opinions on dimensionality reduction of speech data is presented.

4.1 Introduction

Contemporary signal processing applications frequently involve dealing with data sets that are high-dimensional; that is, data sets consisting of a large number of measurements, often sampled at a high frequency. The size of these data sets is constantly increasing with advances in sensor and data storage technologies. Examples of such high-dimensional data sets, which are currently the subjects of large and active research areas, include:

- Speech and audio: Hundreds or thousands of measurements describing the spectral

[1] This chapter is intended to provide an introduction to the large and continually expanding field of dimensionality reduction. More detailed surveys of this field and the methods covered in this chapter are given by Carreira-Perpiñán (2001); Verbeek et al. (2004); Burges (2005); Saul et al. (2006); Lin and Zha (2008); van der Maaten et al. (2009).

content of the signal typically made every 10–40 ms.

- Image: Current image capture technology can measure the colour levels of many millions of pixels. For video this is performed many times every second, with 24 frames per second common.

- Genomic and proteomic data: Microarray and mass spectrometry produce a large amount of data describing the characteristics of an organism or disease. A typical genomic data set consists of thousands of microarray gene measurements while a proteomic profile generated by mass spectrometry commonly contains tens of thousands of measurements (Hauskrecht et al., 2007).

- Text: With the advent of the internet, text processing can require working with billions of documents; each of which is typically represented by a vector describing the frequency of occurrence of each of many thousands of words in a dictionary.

It is often desirable to reduce the dimensionality of such high-dimensional data prior to processing as the dimensions are often correlated and may contain a large amount of redundant information which only serves to obscure the significant information within the data. Also, the dimensionality of the original high-dimensional data set—the number of measurements made—may be higher than the number of degrees of freedom of the measured process or system. Thus, the inherent dimensionality of the data may be lower than the dimensionality of the original data space. An example of this is given in Figure 4.1 which shows images of the Newell teapot[2] in various degrees of rotation about one dimension. Each of these images is represented by 1080900 (1201×901) values, with each value representing the grayscale level of a single pixel. This data set is clearly high-dimensional, however the underlying data has just one degree of freedom—the dimension of rotation—and thus could be adequately represented using just one feature. In this case the goal of dimensionality reduction would be to discover this one significant feature.

[2]This teapot model, also known as the Utah teapot, was created by Martin Newell (1975) at the University of Utah and has become a standard three-dimensional reference object used in the computer graphics field. The teapot data set used in this work was developed by Mathworks (2004).

Figure 4.1: Images of the Newell teapot rotated in one dimension.

4.2 Curse of dimensionality

A further motivation for the reduction of high-dimensional data is the 'curse of dimensionality'. This phrase, first used by Bellman (1961) in the field of adaptive control processes, describes the fact that when estimating a function to a given degree of accuracy the amount of data required grows exponentially with the dimensionality of the data. This problem is due to the exponential increase in volume with increasing numbers of dimensions. For a simple example of this—after Steinbach et al. (2004)—consider distributing 100 points randomly in the unit interval $[0, 1]$ and then partitioning the interval into ten equal length, evenly spaced cells . The resulting cells are all likely to contain a number of points. Next, consider distributing 100 points in a similar fashion in two-dimensional space. An equivalent partitioning scheme requires dividing each dimension in 10 and will result in 100 two-dimensional cells, a number of which will likely be empty. The number of cells increases with the number of dimensions. So for a D-dimensional space, 10^D cells would be required. Thus, the number of empty cells increases with dimension. A visual demonstration of this increase in volume is shown in Figure 4.2, which depicts 100 points randomly distributed in one-, two-, and three-dimensional spaces. The volume of the space and data sparsity can be seen to increase with dimension, illustrating the fact that the amount of data required for function estimation to a given degree of accuracy increases exponentially with dimension.

As a further illustration of the inherent sparsity of high-dimensional spaces (Scott, 1992) consider a D-dimensional hypersphere of radius r inscribed within a D-dimensional hypercube with sides of length $2r$. An illustration of this, where $D = 3$, is shown in Figure 4.3. The sphere's volume is defined as

$$V_S(D, r) = \frac{\pi^{\frac{D}{2}} r^D}{\Gamma\left(\frac{D}{2} + 1\right)} \ , \tag{4.1}$$

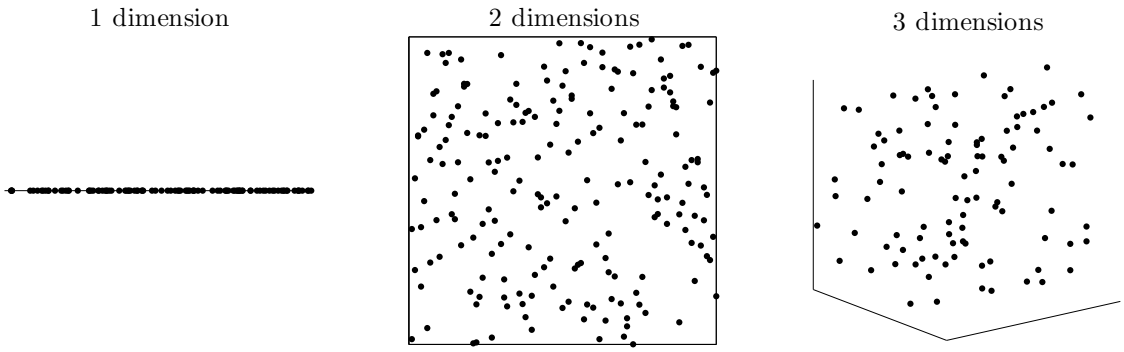


Figure 4.2: Increasing data sparsity with increasing dimensionality, after Wang (2006).

where

$$\Gamma(p) = \int_0^\infty e^{-x} x^{p-1} dx \ . \tag{4.2}$$

While the volume of the hypercube can be calculated as

$$V_C(D, r) = (2r)^D \ . \tag{4.3}$$

The proportion of the volume of the sphere, V_S, to the volume of the cube, V_C, is given by

$$\frac{V_S}{V_C} = \frac{\pi^{\frac{D}{2}}}{2^D \Gamma\left(\frac{D}{2}+1\right)} \ \to 0 \ , \quad \text{as D} \to \infty \ . \tag{4.4}$$

It can been seen that as the dimensionality increases the volume of the sphere becomes *much* less than that of the cube. This means that the vast majority of the volume of the space is located in the 'corners' of the cube. This result illustrates the inherent sparsity and vastness of high-dimensional spaces and the requirement for more data points to accurately model such spaces.

However, in practice the addition of new dimensions often leads to poorer performance. This may be caused by the curse of dimensionality, if the number of training samples is not sufficient relative to the data dimensionality. It may also result from the inclusion of irrelevant information with the new features.

4.3 Dimensionality reduction methods

A large number of methods have been proposed to reduce the dimensionality of a data set by producing a small number of features that describe the key characteristics of the data

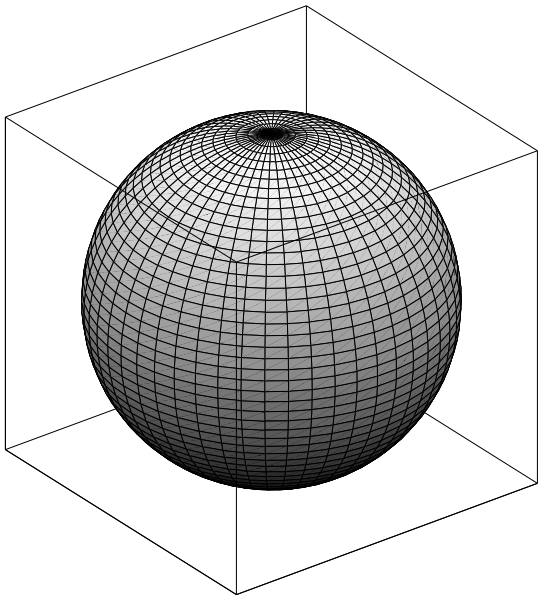


Figure 4.3: A sphere inscribed within a cube in three-dimensional space.

and preserve discriminatory information. These methods can reveal information regarding the true degrees of freedom of the system and help overcome the problems such as the curse of dimensionality, mentioned above. Potential applications of these methods include:

- Visualisation: When dealing with high-dimensional data it may be difficult to determine significant patterns and discover key characteristics. This problem can be overcome by reducing the data down to its two or three most significant dimensions and visually analysing the data to determine any structure, patterns, outliers, and so forth.

- Compression: In situations where transmission bandwidth or data storage resources are limited, it may be desirable to reduce the number of measurements required to adequately represent the data. This can be accomplished by applying dimensionality reduction techniques to achieve the appropriate balance between data dimensionality and loss of information.

- Noise removal: Dimensionality reduction methods can be applied to retain important information while removing redundant information. As noise can be considered redundant information, this strategy can be employed for noise removal.

- Classification: In classification tasks it is necessary to have a feature space in which

different classes are well separated. This separation can be achieved by applying dimensionality reduction methods to preserve discriminative information while removing irrelevant information.

- Improving efficiency: Reduction of dimensionality reduces storage requirements and the computational complexity of any subsequent processing.

Dimensionality reduction methods may be categorised as either feature selection or feature extraction techniques. Feature selection methods aim to select a subset of the original dimensions to represent the data while minimising information loss. These methods offer the advantage of producing features that have a clear meaning attached to them; for example, a particular set of pixels in the case of image processing, or a distinct set of frequency bands in speech or audio processing. They are also advantageous in terms of practical system implementation as once the required features have been identified it is only necessary to compute this small set of features, rather than all the original measurements.

In contrast, feature extraction methods produce an entirely new set of features by forming a combination of the original features, rather than simply choosing a subset of the original features. These new features are formed by performing some operation, such as a projection, to map the original high-dimensional features into a lower-dimensional space. The resulting features are often referred to as hidden or latent variables. Feature extraction techniques are advantageous as they are not limited to a selection of the existing dimensions. The dimensionality reduction techniques used in this dissertation, detailed in Sections 4.3.1 and 4.3.2, may be categorised as feature extraction rather than feature selection.

Dimensionality reduction methods may also be categorised as supervised or unsupervised, linear or nonlinear. Supervised methods require class label information to be provided for each data point, whereas unsupervised methods process unlabelled data. Unsupervised methods are currently the most prevalent as it is often a very time consuming, expensive, error-prone, and thus impractical task for a human, or humans, to manually label an entire data set. As a result we shall primarily focus on unsupervised dimensionality reduction methods. Linear methods are constrained to performing linear transformations of the high-dimensional data whereas nonlinear methods attempt to overcome this con-

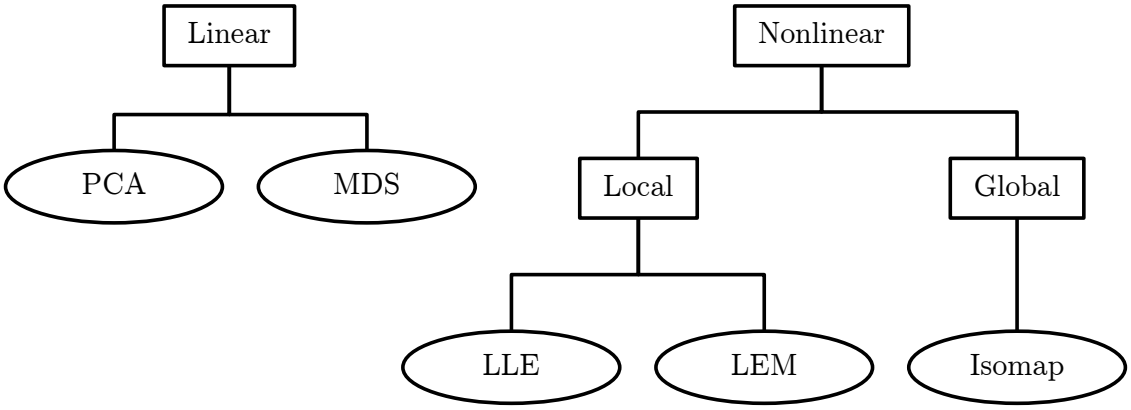


Figure 4.4: Categories of dimensionality reduction methods.

straint. A detailed description of a number of linear and nonlinear methods follows in Section 4.3.1 and 4.3.2, respectively. A diagram illustrating the different categories that the various dimensionality reduction methods used in this dissertation belong to is shown in Figure 4.4.

4.3.1 Linear methods

Linear dimensionality reduction methods are limited to forming linear combinations of the original high-dimensional features. These methods are generally efficient, easy to implement, and often provide a (potentially lossy) bidirectional mapping between high- and low-dimensional space. However, they are constrained to projecting the data onto a linear manifold within the original high-dimensional feature space. The classical PCA and MDS linear dimensionality reduction methods are discussed in detail in the following subsections.

Principal component analysis

PCA (Jolliffe, 1986), also referred to as the Karhunen-Loève transform (KLT), is a well known and widely used linear dimensionality reduction technique. It was originated by Pearson (1901) and developed further in a number of classical papers by Hotelling (1933). The aim of PCA is to produce a low-dimensional representation of high-dimensional data that preserves the greatest sources of variation within the data set. This is achieved by performing a linear transformation of the data, projecting it onto the axes of greatest variance, called the *principal components*. The resulting low-dimensional features are

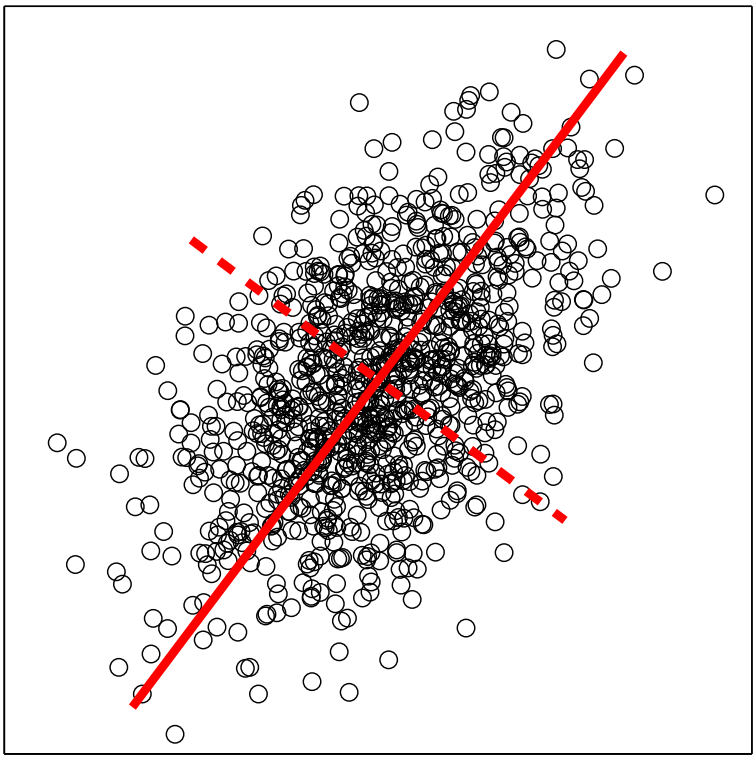


Figure 4.5: The principal components of a two-dimensional data set; the solid line indicates the first principal component, the dashed line represents the second principal component.

uncorrelated and ordered such that the greatest variance by any projection of the data set is accounted for by the first dimension, the second greatest variance by the second dimension, and so on. This is illustrated in Figure 4.5, which shows 1000 points distributed in two-dimensional space with the first and second principal components displayed.

The PCA method can be applied as follows. We begin with an $N \times D$ matrix $\mathbf{X}$ consisting of N D-dimensional points

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]' \quad , \tag{4.5}$$

where $\mathbf{X}$ is assumed to be centered, i.e. have zero mean.[3] PCA finds a linear combination of these D-dimensions resulting in a $N \times d$ matrix

$$\mathbf{Y} = \mathbf{X}\mathbf{A} \quad , \tag{4.6}$$

[3]Nonzero mean data can be centered by simply subtracting the mean $\boldsymbol{\mu}_{\mathbf{x}}$, as follows:

$$\bar{\mathbf{x}}_i = \mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}} \quad \text{for } i = 1, \ldots, N \ .$$

where $d \ll D$. Given the data covariance matrix,

$$\mathbf{C_X} = \frac{1}{N-1}\mathbf{X}'\mathbf{X} \ , \tag{4.7}$$

the $D \times d$ linear transformation matrix $\mathbf{A}$ is composed of the d eigenvectors of $\mathbf{C_X}$ having the largest eigenvalues

$$\mathbf{A} = [\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_d] \ , \tag{4.8}$$

with the eigenvector $\boldsymbol{\alpha}_1$ corresponding to the largest eigenvalue, $\boldsymbol{\alpha}_2$ corresponding to the second largest, and so on.

The above definition can be derived as follows (Jolliffe, 1986). First, consider the case of the first principal component $\boldsymbol{\alpha}_1$; which is required to maximise the variance of

$$\mathrm{Var}(\mathbf{X}\boldsymbol{\alpha}_1) = \boldsymbol{\alpha}_1'\mathbf{C_X}\boldsymbol{\alpha}_1 \ . \tag{4.9}$$

This maximisation problem can be solved by introducing the Lagrange multiplier (Duda et al., 2000, p. 610), λ_1, and the constraint that $\boldsymbol{\alpha}_1'\boldsymbol{\alpha}_1 = 1$. This results in the Lagrangian

$$L(\boldsymbol{\alpha}_1, \lambda_1) = \boldsymbol{\alpha}_1'\mathbf{C_X}\boldsymbol{\alpha}_1 - \lambda_1(\boldsymbol{\alpha}_1'\boldsymbol{\alpha}_1 - 1) \ . \tag{4.10}$$

Differentiating L with respect to $\boldsymbol{\alpha}_1$ gives

$$\frac{\partial L}{\partial \boldsymbol{\alpha}_1} = \mathbf{C_X}\boldsymbol{\alpha}_1 - \lambda_1\boldsymbol{\alpha}_1 = 0 \tag{4.11}$$

and thus

$$\mathbf{C_X}\boldsymbol{\alpha}_1 = \lambda_1\boldsymbol{\alpha}_1 \ , \tag{4.12}$$

therefore, λ_1 is an eigenvalue of $\mathbf{C_X}$ and $\boldsymbol{\alpha}_1$ is the corresponding eigenvector. Using (4.12) we find that the quantity we wish to maximise, previously stated in (4.9), is

$$\begin{aligned} \boldsymbol{\alpha}_1'\mathbf{C_X}\boldsymbol{\alpha}_1 &= \boldsymbol{\alpha}_1'\lambda_1\boldsymbol{\alpha}_1 \\ &= \lambda_1\boldsymbol{\alpha}_1'\boldsymbol{\alpha}_1 \\ &= \lambda_1 \ , \end{aligned} \tag{4.13}$$

thus for the first principal component, which maximises the variance of the projected data, we must set λ_1 equal to the largest eigenvalue and $\boldsymbol{\alpha}_1$ is the corresponding eigenvector. Similarly each successive principal component $\boldsymbol{\alpha}_i$ can be shown to equal the eigenvector corresponding to the ith largest eigenvalue.

The relationship between the eigenvalues and the variance of the data set can be exploited in a number of ways. For example, all of the eigenvalues can be plotted and their relative magnitude compared. If there is a large number of relatively small eigenvalues followed by a series of much larger values, it would suggest that only the eigenvectors corresponding to these larger eigenvalues are necessary to represent most of the data variation after PCA. Also, the eigenvalues can be used to measure the fraction of variance preserved with a chosen number of principal components. The sum of all eigenvalues equals the total data variance, thus comparing this to the sum of the eigenvalues corresponding to the chosen principal components will reveal the fraction of variance preserved, as follows

$$\frac{\sum_{i=1}^{d} \lambda_i}{\sum_{i=1}^{D} \lambda_i} \quad . \tag{4.14}$$

PCA is a popular method, due in part to its simplicity, and has been successfully applied to wide range of data sets. However, PCA is not the best technique for all data sets. In some cases the axes of transformation chosen by PCA may not be optimal for feature extraction. PCA is only concerned with first order correlations and thus will be sub-optimal for data sets where higher order moments are significant.

The effectiveness of PCA is limited by the significant constraint of its global linearity. A number of nonlinear variants of PCA have been proposed in an effort to overcome this constraint. Of these nonlinear variants kernel PCA (Schölkopf et al., 1998) is one of the most widely used.

Classical multidimensional scaling

MDS (Cox and Cox, 2001) is a statistical method for uncovering low-dimensional structure in high-dimensional data; in this respect it is similar to PCA. However, rather than preserving the greatest sources of variance of the data, as is the case in PCA, MDS produces a low-dimensional representation whose interpoint distances preserve as best as possible the pairwise similarities between the original data points. Simply stated, after MDS, sim-

ilar points are located close together and dissimilar points are far apart. The measure of similarity can be achieved in many different ways; for example, simply computing the Euclidean distance between all points in feature space or, asking people to assess the similarity of each pair of objects in a collection. The flexibility afforded by the ability of MDS to work with many different types of distance measures has led to its application in many different fields including: psychometrics, where the technique originated; economics; sociology; anthropology; political science; market research; and speech analysis, as mentioned in Chapter 3.

The term MDS is commonly used to refer to a set of mathematical techniques, rather than a single method. A myriad of variants of MDS have been proposed, including methods which are capable of determining common structure in multiple similarity matrices. In general, MDS methods can be categorised as either metric or nonmetric. Metric MDS refers to case when there is some quantitative objective distance measure indicating the similarity between objects. In contrast, in nonmetric MDS the only requirement is that the objects are ranked according to similarity. In this dissertation we focus on the classical metric MDS, as it is relevant to the Isomap method discussed in Section 4.3.2.

MDS can be mathematically described as follows. Given a set of N high-dimensional points $\mathbf{x}_1, \ldots, \mathbf{x}_N$ MDS seeks to find a low-dimensional representation $\mathbf{y}_1, \ldots, \mathbf{y}_N$. We define the distance between two points $\mathbf{x}_i$ and $\mathbf{x}_j$ as δ_{ij} and let d_{ij} equal the distance between $\mathbf{y}_i$ and $\mathbf{y}_j$. MDS aims to find a low-dimensional representation which minimises the differences between all δ_{ij} and d_{ij}. Ideally this equates to $\delta_{ij} = d_{ij} \ \forall i, j$; however, this is unlikely to be possible. As a result, various different types of error or *stress* function are used in practice, for example

$$S_M = \sum_{i,j=1}^{N} (\delta_{ij} - d_{ij})^2 \ . \tag{4.15}$$

The general MDS algorithm, using a standard gradient-descent method, can be stated as follows:

1. Assign N points $\mathbf{y}_1, \ldots, \mathbf{y}_N$ arbitrarily in low-dimensional space.

2. Compute a measure (e.g. Euclidean distance) of the interpoint distances d_{ij} in this space.

3. Evaluate the stress function, for example (4.15), using these distances and the original dissimilarity measure δ_{ij}.

4. Change the low-dimensional points $\mathbf{y}_i$ in the direction that best decreases the stress function.

5. Repeat the above 3 steps until the stress function ceases to decrease.

As an example of a specific MDS method we describe classical metric MDS, one of the earliest proposed MDS approaches. In classical metric MDS all distances are computed using the squared Euclidean distance function,

$$\delta_{ij}^E = \|\mathbf{x}_i - \mathbf{x}_j\|^2 \ . \tag{4.16}$$

Assume we are given only the interpoint similarity matrix $\mathbf{\Delta}$ consisting of the interpoint distances δ_{ij}^E for all points in the high-dimensional space. The task is then to reconstruct the original high-dimensional points using this matrix. This can be achieved by first constructing a symmetric matrix $\mathbf{T}$ containing the inner products between data points

$$\mathbf{T} = \mathbf{X}\mathbf{X}' \ , \tag{4.17}$$

with elements $t_{ij} = \mathbf{x}_i'\mathbf{x}_j$. As we do not know the original data points we require a means of calculating the inner products from the squared interpoint distances. These squared distances can be expressed in terms of the inner products as

$$\begin{aligned} \delta_{ij}^E &= (\mathbf{x}_i - \mathbf{x}_j)'(\mathbf{x}_i - \mathbf{x}_j) \\ &= \mathbf{x}_i'\mathbf{x}_i + \mathbf{x}_j'\mathbf{x}_j - 2\mathbf{x}_i\mathbf{x}_j \\ &= t_{ii} + t_{jj} - 2t_{ij} \ , \end{aligned} \tag{4.18}$$

and, assuming the data is centered, the inner products can be thus calculated from the squared distances (Verbeek, 2004) as

$$t_{ij} = -\frac{1}{2}\left(\delta_{ij}^E - \frac{1}{N}\sum_{j=1}^{N}\delta_{ij}^E - \frac{1}{N}\sum_{i=1}^{N}\delta_{ij}^E + \frac{1}{N^2}\sum_{i,j=1}^{N}\delta_{ij}^E\right) \ . \tag{4.19}$$

This has the effect of subtracting the row and column average of each entry and then adding the overall matrix average back in. The eigendecomposition of the symmetric matrix consisting of these t_{ij} elements is

$$\mathbf{T} = \mathbf{A}\mathbf{\Lambda}\mathbf{A}' \ , \tag{4.20}$$

where $\mathbf{A}$ is a matrix whose columns are the eigenvectors of $\mathbf{T}$ and $\mathbf{\Lambda}$ is a diagonal matrix made up of the corresponding eigenvalues. The original points can be optimally reconstructed, using (4.17) and (4.20), as

$$\mathbf{X} = \mathbf{A}\mathbf{\Lambda}^{\frac{1}{2}} \ . \tag{4.21}$$

MDS seeks a d-dimensional representation $\mathbf{Y}$ of the original high-dimensional data $\mathbf{X}$. This can be achieved as follows

$$\mathbf{Y} = \mathbf{A}_d\mathbf{\Lambda}_d^{\frac{1}{2}} \ , \tag{4.22}$$

where $\mathbf{\Lambda}_d$ is diagonal and contains the d largest eigenvalues of $\mathbf{T}$ and $\mathbf{A}_d$ contains the corresponding d eigenvectors.

While the PCA and MDS methods both aim to preserve different properties of the data they are effectively equivalent when Euclidean distances are used in MDS, as above. As with PCA, the classical metric MDS approach is limited to discovering underlying linear structure. Various nonlinear variants of MDS have been developed including the popular Sammon's mapping (Sammon Jr., 1969) that is similar to the classical metric MDS described above but uses a different stress function

$$S_S = \frac{1}{\sum_{i,j=1}^{N} \delta_{ij}} \left(\sum_{i,j=1}^{N} \frac{(\delta_{ij} - d_{ij})^2}{\delta_{ij}} \right) \ , \tag{4.23}$$

which has the effect of emphasising the preservation of short distances. Further nonlinear extensions of MDS are discussed in the following section.

4.3.2 Nonlinear methods

As mentioned previously, the linear methods discussed above are constrained to operate in situations where the underlying structure of the data is embedded linearly, or almost

linearly, in the high-dimensional space. As a result, these methods are unable to discover intrinsic structure nonlinearly embedded in high-dimensional space. For example, these methods would fail to reveal the true low-dimensional structure of the Swiss roll and teapot data sets presented in Figures 1.1 and 4.1, respectively. Beyond these toy data sets, the assumption of linearity is also a problem in many real-world tasks. For example images of faces and handwritten digits break the assumption of an underlying linear subspace and require a nonlinear mapping.

In order to overcome the linear limitations of methods such as those discussed in the previous section a number of methods have been proposed based on the assumption that the data has an underlying nonlinear manifold structure. These methods aim to map high-dimensional data onto a nonlinear low-dimensional manifold while retaining the underlying structure of the data; as a result these methods are often referred to as manifold learning methods. Manifold learning methods may be categorised according to the type of structure they aim to preserve. The underlying global geometry of a data set is preserved by methods such as Isomap and its variant, landmark Isomap (L-Isomap), while local geometric structure is preserved by the LLE and Laplacian eigenmaps algorithms. The next section provides a brief description of manifolds and the assumptions made by the manifold learning methods. Following this, each of the manifold learning methods are discussed in detail.

A note on manifolds

The term manifold, as used in differential geometry and topology, may be formally defined as a topological space which is locally homeomorphic to Euclidean n-space, $\mathbb{R}^n$ (Hirsch, 1976). Stated less formally, a manifold is a space that is locally Euclidean—that is, on a small scale in a local neighbourhood such a space resembles the Euclidean space of a specific dimensionality, this is the dimensionality of the manifold. Thus, a curve is an example of a one-dimensional manifold and a sphere is an example of a two-dimensional manifold (Seung and Lee, 2000). Figure 1.1 provides a further example of a two-dimensional manifold.

The manifold learning methods used in this dissertation—Isomap, LLE, and Laplacian eigenmaps—assume that the input data points are sampled from a smooth manifold, i.e. a manifold that contains no discontinuities (van der Maaten et al., 2009). As a result of

this assumption manifold learning methods typically perform poorly under the presence of disconnected, i.e. non-smooth, manifolds in the data. Also, most manifold learning methods are unable to discover the true underlying manifold structure when the manifold contains holes (Tenenbaum, 1998). The interested reader may wish to refer to Lin and Zha (2008) for a more detailed discussion of manifolds in relation to manifold learning methods. Having discussed what a manifold is, each of the three manifold learning methods used in this dissertation are discussed in detail in the following sections.

Isomap

The Isomap algorithm[4] (Tenenbaum et al., 2000) is a nonlinear generalisation of the classical MDS method discussed in Section 4.3.1. Classical MDS is concerned with Euclidean distances in feature space. However, given a situation where the data lies on or near a manifold nonlinearly embedded in the high-dimensional feature space Euclidean distances may be unsuitable. In this case, the use of Euclidean distances may result in points which are located a large distance apart on the manifold being incorrectly classed as close neighbours. This is illustrated in Figure 4.6 which shows data with an inherent dimension of two embedded in three-dimensional space. Both the Euclidean distance and the geodesic[5] distance—that is, the distance on the manifold from which the data is sampled—between two points are shown. It can be seen that the geodesic distance gives a truer reflection of the underlying geometrical structure of the data. The geodesic distance will be large for two points which are located far apart on the manifold and small for two points which are located close together on the manifold. This is in contrast to the Euclidean distance which may be small for two points which are located far apart on the manifold.

Isomap extends the classical MDS methods and aims to preserve geodesic, rather than Euclidean, distances. It seeks a mapping from a D-dimensional data set of size N, $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]'$, to d-dimensional feature space $\mathbf{Y}$, where $d \ll D$, that preserves geodesic distances between pairs of data points. Thus, Isomap aims to produce a new set of features such that the difference between the geodesic distance between points $\mathbf{x}_i$ and $\mathbf{x}_j$ and the Euclidean distance between points $\mathbf{y}_i$ and $\mathbf{y}_j$ is minimised. This has the

[4] Isomap code can be found at `http://web.mit.edu/cocosci/isomap/isomap.html`.

[5] A geodesic is the shortest curve connecting two points on a manifold. For example, a longitudinal line along the Earth's surface connecting the two poles. The length of such a curve is the geodesic distance.

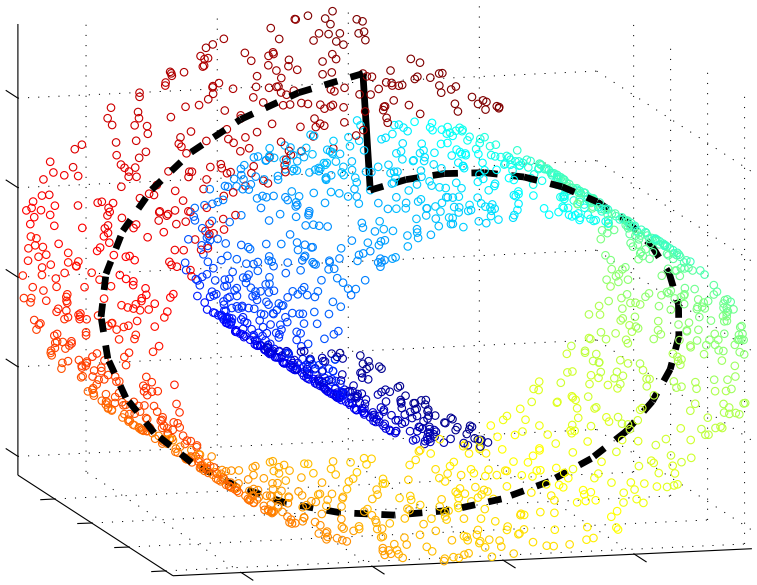


Figure 4.6: Inherently two-dimensional data nonlinearly embedded in three-dimensional space. Point colour corresponds to position on the underlying manifold. The Euclidean distance, solid line, and geodesic distance, dashed line, between two points are illustrated.

effect of preserving the global geometric properties of the manifold while reducing the dimensionality.

The Isomap algorithm consists of three steps:

1. Construct a neighbourhood graph: For each data point $\mathbf{x}_i$ $(i = 1, \ldots, N)$, compute its neighbours. Various approaches exist to compute the neighbours and the user must specify an approach appropriate to the data set. Two simple and popular approaches are the ϵ-radius and k-nearest neighbour approaches. In the case of the k-nearest neighbour approach a definition δ_{ij} of the 'nearness' of points $\mathbf{x}_i$ and $\mathbf{x}_j$, for example the Euclidean distance $\delta_{ij}^{\mathbf{X}} = \|\mathbf{x}_i - \mathbf{x}_j\|$, is used to determine the k data points closest to $\mathbf{x}_i$ in the data space. Alternatively using the ϵ-radius approach the neighbours of $\mathbf{x}_i$ are defined as all points $\mathbf{x}_j$ for which $\delta_{ij}^{\mathbf{X}} < \epsilon$. Having computed the points making up the 'neighbourhood' of each point in the data set, a weighted neighbourhood graph $\mathcal{G}$ is constructed by connecting neighbouring points. An edge $e(i, j)$ is connected between vertices v_i and v_j only if $\mathbf{x}_i$ and $\mathbf{x}_j$ are neighbours. The weight w_{ij} of $e(i, j)$ is set equal to $\delta_{ij}^{\mathbf{X}}$.

2. Estimate geodesic distances: Following construction of the neighbourhood graph it is assumed that the geodesic distance, the shortest path distance between two

points i and j on the manifold, is approximately equal to the length of the shortest path through the neighbourhood graph, $\mathcal{G}$: g_{ij}. The shortest path between all pairs of points on the neighbourhood graph can be computed using a technique such as Floyd's (Floyd, 1962) or Dijkstra's (Dijkstra, 1959) algorithm. The use of Dijkstra's algorithm results in a complexity of $O(kN^2 \log N)$ (de Silva and Tenenbaum, 2003). The resulting geodesic distances g_{ij} form a symmetric matrix $\mathbf{G}$.

3. Apply classical MDS to $\mathbf{G}$: The geodesic distance matrix $\mathbf{G}$ can now be used in place of the Euclidean dissimilarity matrix in the classical MDS method to yield a low-dimensional embedding $\mathbf{Y}$ of the data. First we define $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]'$, $\mathbf{y}_i \in \mathbb{R}^d$. Let $\mathbf{T}$ be a symmetric matrix containing the target inner products between all $\mathbf{y}_i$

$$\mathbf{T} = \mathbf{Y}\mathbf{Y}' \ . \tag{4.24}$$

Assume that the low-dimensional embeddings are translation invariant, $\sum_{i=1}^{N} y_i = 0$. The target inner product matrix can then be computed from the known squared geodesic distances $\hat{\mathbf{G}} = \left\{ g_{ij}^2 \right\} \quad i, j = 1, \ldots, N$ as

$$\mathbf{T} = -\frac{\mathbf{H}\hat{\mathbf{G}}\mathbf{H}}{2} \ , \tag{4.25}$$

where $\mathbf{H}$ is simply a centering matrix with elements

$$h_{ij} = \begin{cases} 1 - \frac{1}{N}, & i = j \ , \\ -\frac{1}{N}, & i \neq j \ . \end{cases} \tag{4.26}$$

Isomap aims to produce a low-dimensional embedding such that the difference between the geodesic distances g_{ij}^2 and the Euclidean distance between points $\mathbf{y}_i$ and $\mathbf{y}_j$ is minimised. Thus, we wish to minimise the least square error between $\mathbf{Y}\mathbf{Y}'$ and $\mathbf{T}$. This can be accomplished by performing an eigendecomposition of $\mathbf{T}$. The low-dimensional embeddings can then be computed as

$$\mathbf{Y} = \mathbf{A}_d \mathbf{\Lambda}_d^{\frac{1}{2}} \ , \tag{4.27}$$

where $\mathbf{\Lambda}_d$ is diagonal and contains the d largest eigenvalues of $\mathbf{T}$ and $\mathbf{A}_d$ con-

tains the corresponding d eigenvectors. These computations are analogous to Equations (4.17)–(4.22) of classical MDS.

Landmark-Isomap

The Isomap algorithm requires the performance of a number of computationally expensive operations and as a result it may be inefficient and impractical to run it on large data sets. There are two particularly demanding computations (de Silva and Tenenbaum, 2003). First, the computation of the geodesic distance matrix $\mathbf{G}$, Step 2 above, which has a complexity of $O(kN^2 \log N)$ using a Dijkstra's algorithm based approach or, worse still, $O(N^3)$ using Floyd's algorithm. Second, the eigendecomposition used in MDS, Step 3 above, involves a full $N \times N$ matrix and has complexity $O(N^3)$.

De Silva and Tenenbaum (2003) proposed the L-Isomap method which reduces the computational requirements of Isomap that prohibit its application to large data sets. In L-Isomap n data points are chosen as landmark points on the manifold, where $n \ll N$. L-Isomap operates in a similar fashion to Isomap, however rather than preserving the distances between all pairs of points, only distances between all points and the n landmark points are preserved. This is achieved using a landmark MDS procedure (de Silva and Tenenbaum, 2003, 2004), as follows:

1. Choose n landmark points from the data set $\mathbf{X}$. The number of landmark points, n, must be larger than the minimum $d+1$. Based on the assumption that the manifold is well-sampled these points may be randomly selected from the input data set.

2. Apply classical MDS to the $n \times N$ matrix $\mathbf{\Delta}^L$ of distances between each point and the landmark points. This produces d-dimensional embeddings of the n landmark points.

3. To embed the remaining points in $\mathbb{R}^d$, first compute a matrix $\mathbf{\Delta}^X$ whose columns $\boldsymbol{\delta}_i^X$ contain the squared distances between each data point $\mathbf{x}_i$ and each of the n landmark points. The ith element of the low-dimensional representation y_{ij} of point $\mathbf{x}_{\bullet j}$ can then be computed as

$$y_{ij} = -\frac{1}{2}\frac{\boldsymbol{\alpha}_i^{'}}{\sqrt{\lambda_i}}(\boldsymbol{\delta}_j^X - \boldsymbol{\delta}_\mu^L) \ , \tag{4.28}$$

where $\boldsymbol{\delta}_\mu^L$ is the column mean of $\boldsymbol{\Delta}^L$. The term λ_i equals the ith largest eigenvalue of the inner-product matrix constructed from $\boldsymbol{\Delta}^L$, as in (4.25), with $\boldsymbol{\alpha}_i$ denoting the corresponding eigenvector.

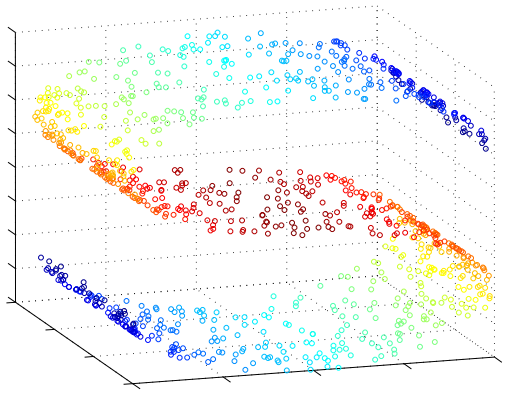
For a small n, relative to N, this approach greatly reduces the computational bottlenecks of Isomap. The $n \times N$ matrix $\boldsymbol{\Delta}^L$ can be computed using Dijkstra's algorithm with complexity $O(knN \log N)$ as opposed to $O(kN^2 \log N)$ and the landmark MDS procedure above has complexity $O(n^2 N)$, improving on classical MDS which runs in $O(N^3)$.

A comparison of the performance of L-Isomap using various numbers of landmark points is shown in Figure 4.7. In this example L-Isomap is applied to a data set of points sampled from a two-dimensional plane nonlinearly embedded in three-dimensional space. L-Isomap successfully discovers the underlying manifold using both large and small numbers of landmark points. The performance of L-Isomap with a relatively small number of landmark points is shown to be comparable to standard Isomap, Figure 4.7(b), which uses all points in the data set. It is interesting to note the small amount of rotation visible in the low-dimensional embeddings of Figure 4.7(g) and Figure 4.7(i). This minor rotation is due to a bias caused by the location, in high-dimensional space, of the particular landmark points selected in both these cases. To verify that this was the cause of the rotation, this experiment was repeated several times with different sets of landmark points and the rotation was not found to reoccur. Such a bias is most likely to occur with a small number of landmark points as the potential for these points to insufficiently sample the embedded manifold is greatest in this case.

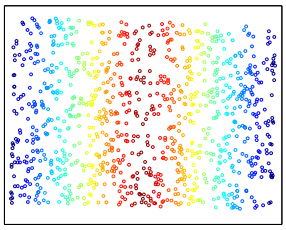
Locally linear embedding

LLE[6] (Roweis and Saul, 2000) is an unsupervised learning algorithm that computes low-dimensional embeddings of high-dimensional data. The principle of LLE is to compute a low-dimensional embedding with the property that nearby points in the high-dimensional space remain nearby and similarly co-located with respect to one another in the low-dimensional space. In other words, the embedding is optimised to preserve local neighbourhoods, as illustrated in Figure 4.8; this is in contrast to Isomap which aims to preserve the underlying global structure of the data set.

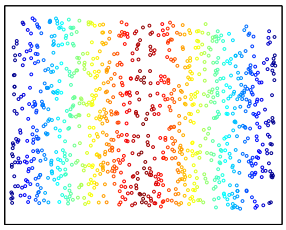
[6] Additional details, examples, and code relating to the LLE algorithm are available at the following website: `http://www.cs.toronto.edu/~roweis/lle/`.



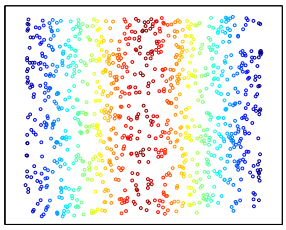
(a) Original S–curve data set



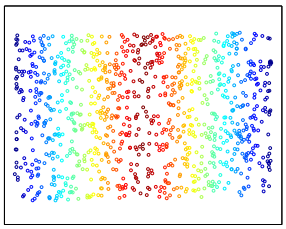
(b) $n = N = 2000$



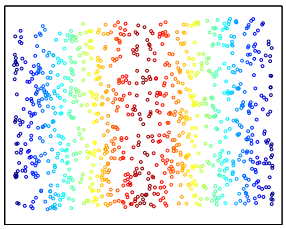
(c) $n = 1000$



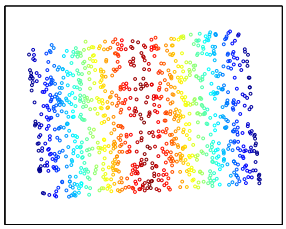
(d) $n = 500$



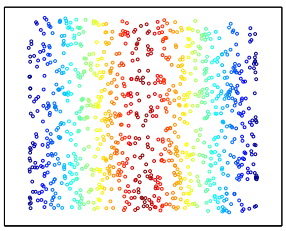
(e) $n = 250$



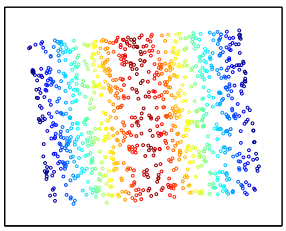
(f) $n = 125$



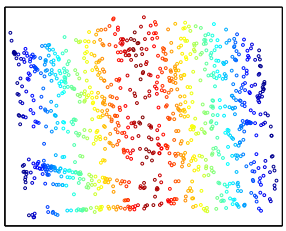
(g) $n = 62$



(h) $n = 31$



(i) $n = 7$



(j) $n = 3$

Figure 4.7: Effect of the number of landmark points n on L-Isomap. Two-dimensional embeddings resulting from the application of L-Isomap on (a) $N = 2000$ data points sampled from a three-dimensional S-curve manifold using different numbers of randomly sampled landmark points, n. L-Isomap successfully discovers the underlying two-dimensional structure for a wide range of n values, from $n = N$ (all the points) to $n = 3$.

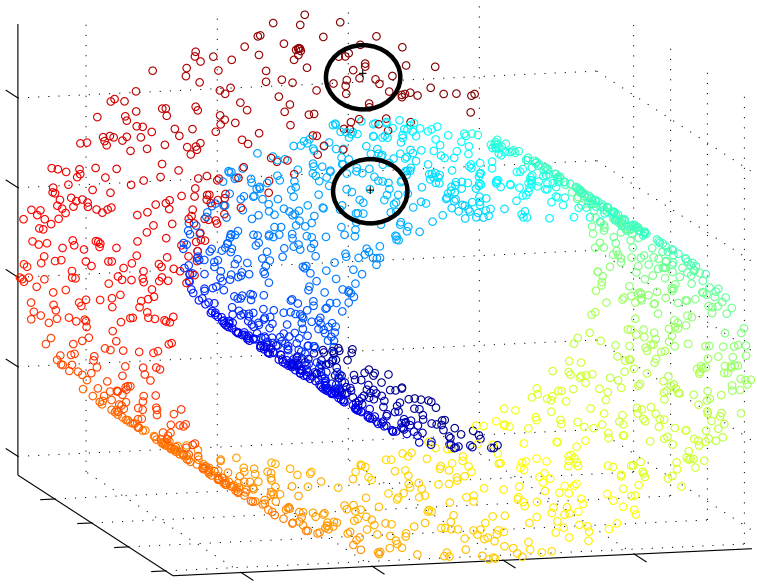


Figure 4.8: Inherently two-dimensional data nonlinearly embedded in three-dimensional space. Point colour corresponds to position on the underlying manifold. Examples of the locally linear neighbourhoods whose structure is preserved by LLE are outlined in black.

As with Isomap, the LLE algorithm can be summarised in three steps (Saul and Roweis, 2003):

1. For each data point $\mathbf{x}_i$ compute its k neighbours. This can be accomplished using the k-nearest neighbours or ϵ-radius schemes described in Step 1 of the standard Isomap algorithm.

2. Compute weights $\mathbf{W} = \{w_{ij}\}$ that best reconstruct each data point $\mathbf{x}_i$ from its neighbours, minimising the reconstruction error E:

$$E(\mathbf{W}) = \sum_{i=1}^{N} \left\| \mathbf{x}_i - \sum_{j=1}^{N} w_{ij} \mathbf{x}_j \right\|^2 \tag{4.29}$$

   The weight w_{ij} scales the amount by which the data point $\mathbf{x}_j$ contributes to the reconstruction of point $\mathbf{x}_i$.

3. Compute the low-dimensional embeddings $\mathbf{y}_i$, best reconstructed by the weights w_{ij}, minimising the cost function Ω:

$$\Omega(\mathbf{Y}) = \sum_{i=1}^{N} \left\| \mathbf{y}_i - \sum_{j=1}^{N} w_{ij} \mathbf{y}_j \right\|^2 \tag{4.30}$$

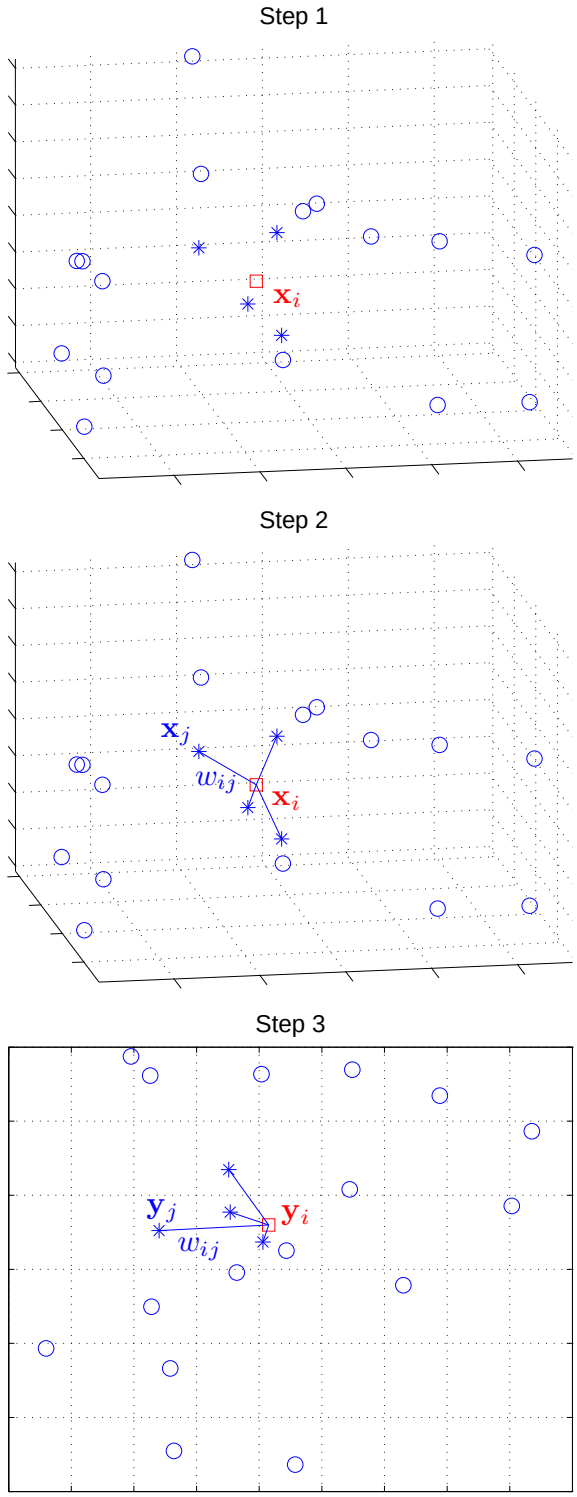


Figure 4.9: The LLE algorithm. In Step 1 the nearest neighbours, plotted as stars, of the data point $\mathbf{x}_i$, represented as a square, are computed (using a $k = 4$ nearest neighbour scheme). In Step 2 the reconstruction weights w_{ij} are calculated, minimising the reconstruction error (4.29). In Step 3 the low-dimensional embeddings $\mathbf{y}_i$ are constructed based on the weights w_{ij}, minimizing the cost function (4.30).

This three step process is illustrated graphically in Figure 4.9.

In Step 2, the reconstruction error is minimised subject to two constraints: first, that each input is reconstructed only from its nearest neighbours, or $w_{ij} = 0$ for any $\mathbf{x}_j$ that is not a neighbour of $\mathbf{x}_i$; second, that the reconstruction weights for each data point sum to one, or $\sum_{j=1}^{N} w_{ij} = 1 \quad \forall\, i$. The optimum weights for each input can be computed efficiently by solving a constrained least squares problem.

The cost function in Step 3 is also based on locally linear reconstruction errors, but here the weights w_{ij} are kept fixed while optimising the outputs $\mathbf{y}_i$. The minimisation is performed subject to constraints that the outputs are centered,

$$\sum_{i=1}^{N} \mathbf{y}_i = 0 \in \mathbb{R}^d \ , \tag{4.31}$$

and have unit covariance,

$$\frac{1}{N} \sum_{i=1}^{N} \mathbf{y}_i \mathbf{y}_i' = \mathbf{I} \ . \tag{4.32}$$

The cost function has a unique global minimum solution for the outputs $\mathbf{y}_i$. This is the result returned by LLE as the low-dimensional embedding of the high-dimensional data points $\mathbf{x}_i$. The embedding cost function can be minimised by first defining a sparse, symmetric, positive semidefinite $N \times N$ matrix $\mathbf{M} = \{m_{ij}\}$ as

$$\mathbf{M} = (\mathbf{I} - \mathbf{W})'(\mathbf{I} - \mathbf{W}) \ . \tag{4.33}$$

This allows (4.30) to be expressed as the quadratic form

$$\Omega(\mathbf{Y}) = \sum_{i,j=1}^{N} m_{ij}(\mathbf{y}_i' \mathbf{y}_j) \ , \tag{4.34}$$

which can be minimised, according to the Rayleigh-Ritz theorem (Horn and Johnson, 1990), by finding the $d+1$ eigenvectors of $\mathbf{M}$ with the smallest nonzero eigenvalues. The unit eigenvector, with the smallest eigenvector, is discarded fulfilling the constraint in (4.31). The remaining d eigenvectors are the low-dimensional embeddings $\mathbf{Y}$ output by LLE.

Many variants and extensions of the LLE algorithm described above have been developed subsequent to its proposal by Roweis and Saul (2000) including kernelised LLE

(DeCoste, 2001), locally linear coordination (Roweis et al., 2002), Hessian LLE (Donoho and Grimes, 2003), supervised LLE (de Ridder et al., 2003; Kayo, 2006), and robust LLE (Chang and Yeung, 2006).

Laplacian eigenmaps

The Laplacian eigenmaps (Belkin and Niyogi, 2002, 2003) algorithm has a similar principle to that of LLE, to compute a low-dimensional representation of high-dimensional data that faithfully preserves proximity relations. It was originally motivated by the way that heat transmits from one point to another. The algorithm is based on a neighbourhood graph based approach, as in Isomap, and is structured as follows:

1. Construct a neighbourhood graph $\mathcal{G}$ as in Step 1 of Isomap.

2. Assign weights w_{ij} to the edges of the graph. These weights are typically constant, e.g. $w_{ij} = 1$, or exponentially decaying, e.g.

$$w_{ij} = e^{-\left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma}\right)} \ , \tag{4.35}$$

   where $\sigma \in \mathbb{R}$ is a scaling parameter.

3. The embeddings $\mathbf{y}_i \ \ (i = 1, \ldots, N)$ are computed by minimising the cost function:

$$\mathfrak{E}(\mathbf{Y}) = \sum_{i,j=1}^{N} w_{ij} \|\mathbf{y}_i - \mathbf{y}_j\|^2 \ . \tag{4.36}$$

This cost function measures the squared distances between the embedded points, with distance measured by the weights in the matrix $\mathbf{W}$, and incurs a heavy penalty if neighbouring high-dimensional points are mapped far apart in embedding space. This cost function can be minimised by first introducing the Laplacian $\mathbf{L}$:

$$\mathbf{L} = \mathbf{\Theta} - \mathbf{W} \ , \tag{4.37}$$

where $\mathbf{\Theta}$ denotes the diagonal weight matrix with elements $\theta_{ii} = \sum_j^N w_{ij}$. The Laplacian is a symmetric, positive definite matrix that represents the graph $\mathcal{G}$. Equation (4.36) can

then be written as:

$$\mathfrak{E}(\mathbf{Y}) = \mathrm{Tr}[\mathbf{Y}'\mathbf{L}\mathbf{Y}] \ . \tag{4.38}$$

Subject to the constraints, as in LLE, that the embedded points are centered (4.31) with unit covariance (4.32), the cost function (4.36) is minimised by solving the eigenvector problem

$$\mathbf{L}\mathbf{y} = \lambda\mathbf{D}\mathbf{y} \ . \tag{4.39}$$

As in LLE, the smallest eigenvector—having the smallest eigenvalue—is discarded and the remaining d smallest eigenvectors are the low-dimensional embeddings $\mathbf{Y}$ output.

4.3.3 Example applications

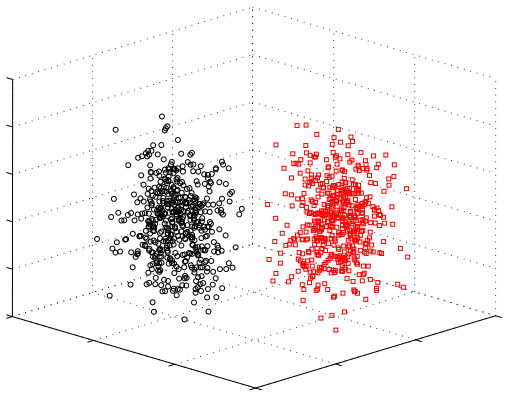
The methods discussed in Sections 4.3.1 and 4.3.2 offer a means of overcoming the inherent problems faced when dealing with high-dimensional data, as introduced and discussed in Sections 4.1 and 4.2. In this section, dimensionality reduction methods are applied to a number of example data sets in order to provide a clear demonstration of the properties and potential applications of these methods.

Two linearly separable classes

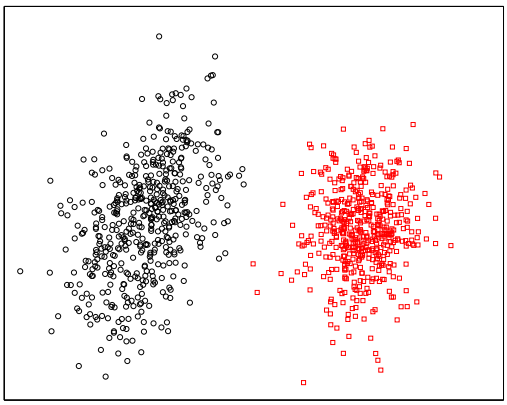
The first data set used to demonstrate the dimensionality reduction methods consists of two sets of $N = 1000$ points distributed in three-dimensional space, as illustrated in Figure 4.10(a). The means of the two sets of data were chosen to ensure the two 'classes' are linearly separable. The dimensionality reduction methods PCA, Isomap, LLE, and Laplacian eigenmaps were each individually applied to reduce the dimensionality of the original 2000 data points from three-dimensions to two-dimensions. The two-dimensional outputs of each of these methods are shown in Figure 4.10(b)–(e). The two classes can be seen to be well separated in the two-dimensional space output by each of the four dimensionality reduction methods.

Swiss roll

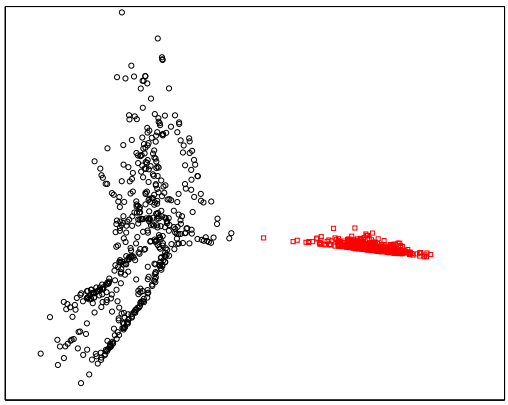
The Swiss roll data set has been used in a number of previous sections as an illustration of a low-dimensional manifold embedded in high-dimensional space. This data set is a two-



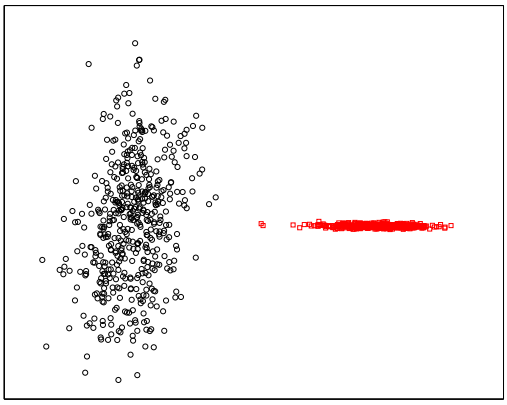
(a) Original linearly separable two class data



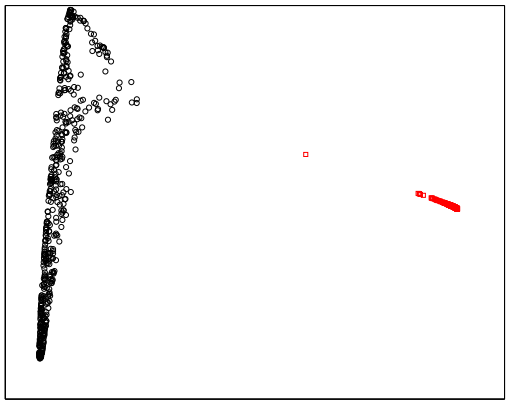
(b) PCA



(c) Isomap



(d) LLE



(e) Laplacian eigenmaps

Figure 4.10: Examples of the performance of four dimensionality reduction methods on $N = 1000$ data points sampled from two linearly separable classes of data, represented by circles and squares, in three-dimensions. The two-dimensional representations resulting from the application of PCA, Isomap, LLE, and Laplacian eigenmaps are shown.

dimensional plane which is nonlinearly embedded in three-dimensional space, as shown in Figure 1.1. The Swiss roll data set is generated using the function:

$$f(\varphi, h) = [\varphi\cos(\varphi),\ h,\ \varphi\sin(\varphi)] \quad , \tag{4.40}$$

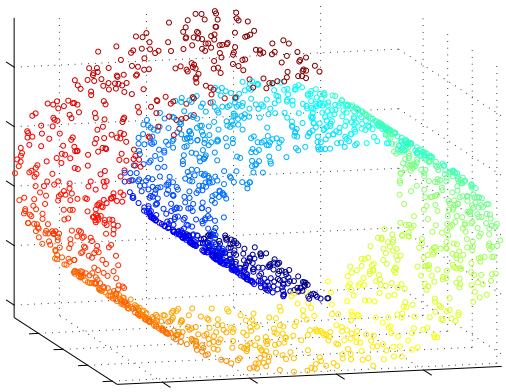
where φ and h are numbers selected randomly from the intervals $\left[\frac{3\pi}{2}, \frac{9\pi}{2}\right]$ and $[0, 21]$, respectively. This data set is frequently used to demonstrate the capabilities of manifold learning methods. The above implementation is based on a data set used by Roweis and Saul (2000) to test the LLE algorithm.

In order to further demonstrate the abilities of the dimensionality reduction methods $N = 2000$ data points were sampled from the Swiss roll data set, computed as detailed in (4.40). PCA, Isomap, LLE, and Laplacian eigenmaps were then each applied to reduce the original three-dimensional data to two dimensions. The two-dimensional outputs resulting from the application of each of these four methods are shown in Figure 4.11. PCA projects the data set into a new coordinate space so as to preserve the principal sources of variation. However, in the case of the Swiss roll data this does not account for the intrinsic geometric structure of the data set and results in an embedding which does not 'unroll' the Swiss roll and discover the underlying two-dimensional manifold. In contrast, all of the manifold learning methods successfully 'unroll' the Swiss roll and uncover its underlying structure. The Isomap algorithm best preserves the global structure of the manifold. This is due to its ability to preserve metric distances.

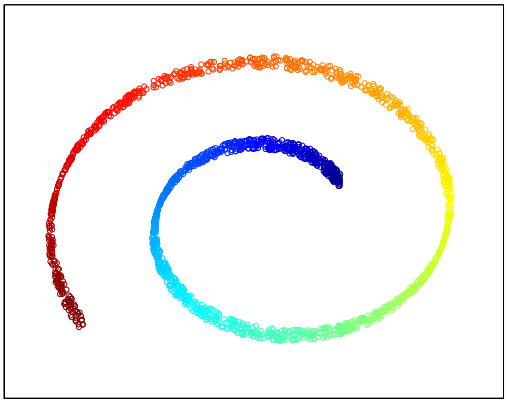
Teapot images

The previously presented examples serve to demonstrate the limitations of linear dimensionality reduction methods and motivate the use of manifold learning as a means to overcome these constraints. However, these examples have dealt with data sets of relatively low dimensionality. The following example uses a higher-dimensional data set and displays the ability of manifold learning methods to reveal information regarding the underlying degrees of freedom of a system.

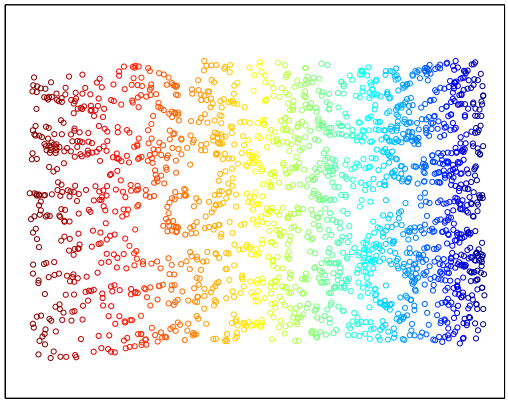
Consider a teapot rotated through 360 degrees in a single dimension, as previously discussed in Section 4.1 this system has one degree of freedom. However, if we were to take measurements of this system in the form of images from a fixed viewpoint the resulting



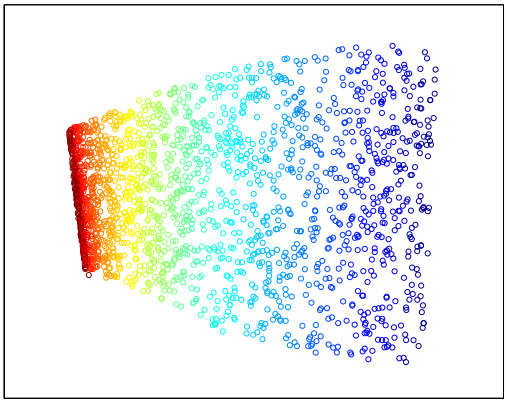
(a) Original Swiss roll data set



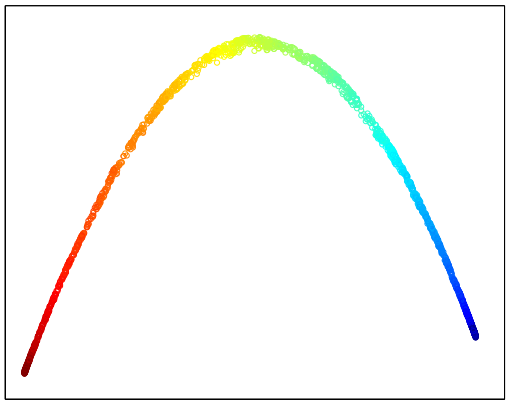
(b) PCA



(c) Isomap



(d) LLE



(e) Laplacian eigenmaps

Figure 4.11: Examples of the performance of four dimensionality reduction methods on $N = 2000$ data points sampled from a three-dimensional Swiss roll structure. The two-dimensional representations resulting from the application of PCA, Isomap, LLE, and Laplacian eigenmaps are shown.

measurements may be high-dimensional. For example, if the images consisted of 32 pixels horizontally and 32 pixels vertically an image could be represented as a feature vector by simply concatenating all of the rows together to yield a 1024-dimensional feature vector—that is, 32×32 pixels. $N = 719$ such images were created by rotating a graphical model of a Newell teapot in half degree increments from 0–360 degrees in one-dimension, about the vertical axis. Example images from this data set are given in Figure 4.12(a). PCA, Isomap, LLE, and Laplacian eigenmaps were then each applied to produce two-dimensional representations of the original 1024-dimensional images. The resulting representations are shown in Figure 4.12(b)–(e). Again, the manifold learning methods produce a feature space in which the underlying low-dimensional structure of the data set is clearly evident. Conversely, PCA is unable to successfully discover the intrinsic manifold.

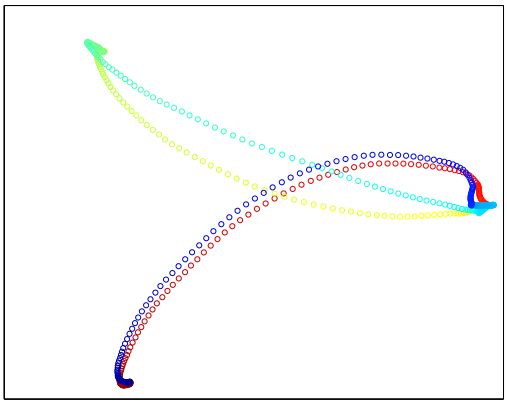
4.3.4 Comparison of dimensionality reduction methods

This section presents a comparison of some of the general properties of the methods discussed above. For a thorough comparison of a wider range of dimensionality reduction algorithms refer to van der Maaten et al. (2009).

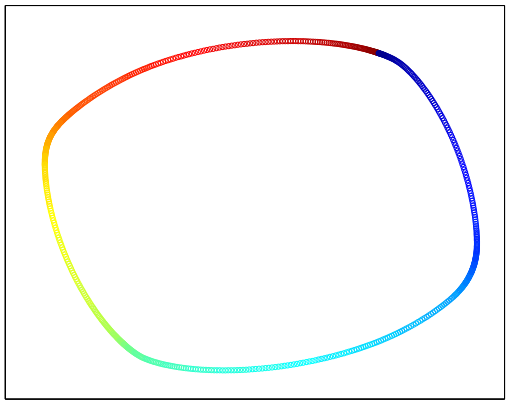
Parameters

One free parameter to be chosen when using all of the dimensionality reduction methods discussed above is simply the dimensionality, d, of the embedding space to be output by these methods. In many cases the choice of d may be specified by the desired application, for example if data visualisation is required the target dimensionality must be one, two, or three. However, in many cases the correct choice of d or the inherent dimensionality of the data is unknown. Assumptions and knowledge of the system responsible for generating the data set may then be used to provide an indication of the underlying dimensionality of the data, for example in the case of the rotated teapot of Figure 4.1 where the underlying dimensionality is known to be one. Inherent dimensionality estimation methods, previously discussed in Chapter 3, can also be used to determine the correct choice of d. Furthermore, when using PCA, the value of d can be chosen such that a certain percentage variance is retained, based on the eigenvalues of the principal component eigenvectors as described in Section 4.3.1, in the low-dimensional embedding.

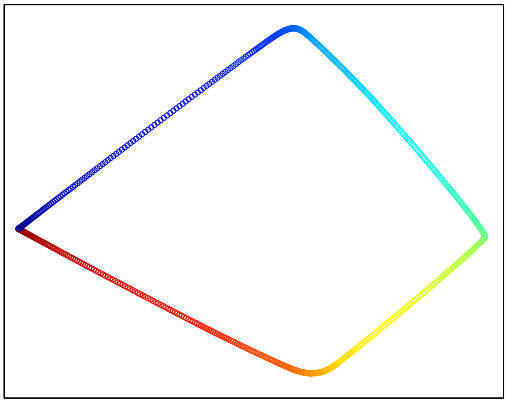
(a) Example images taken from the teapot data set. Images at rotation 0°, 60°, 120°, 180°, 240°, 300°, and 360° are shown above their corresponding colour representation.



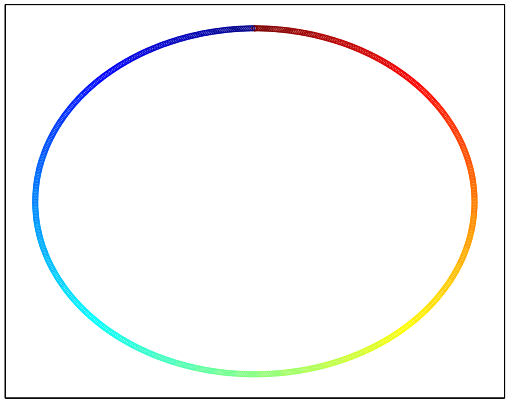
(b) PCA



(c) Isomap



(d) LLE



(e) Laplacian eigenmaps

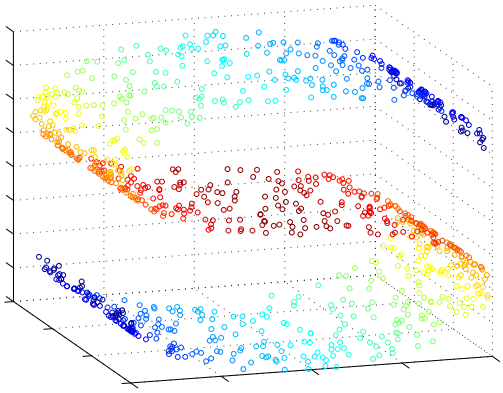
Figure 4.12: Examples of the performance of four dimensionality reduction methods on $N = 719$ images of a teapot rotated through 360 degrees in one-dimension; each original image has 1024 dimensions (32×32 pixels). The two-dimensional representations resulting from the application of PCA, Isomap, LLE, and Laplacian eigenmaps are shown.

When using the nonlinear dimensionality reduction methods described above a second parameter must be considered. This parameter concerns the computation of the nearest neighbours: namely the number of nearest neighbours to compute, k, or the value of ϵ if using an ϵ-radius neighbourhood scheme. A number of approaches have been proposed to estimate the 'optimal' value of this parameter; for example, the work of Kouropteva et al. (2002), Samko et al. (2006), and Shao (2008). These methods use various quantitative measures to attempt to compute the quality of the low-dimensional embedding and select the 'optimal' k value in relation to these measures. However, in practice the number of nearest neighbours used is often chosen empirically. A comparison of the performance of each manifold learning method using various numbers of nearest neighbours is shown in Figure 4.13. In this example each method is applied to a data set of points sampled from a two-dimensional plane nonlinearly embedded in three-dimensional space. If the number of nearest neighbours used is too small or too large the manifold learning methods fail to discover the underlying two-dimensional manifold. However, if the number of nearest neighbours used is in the correct range the methods successfully uncover the latent geometric structure. This issue is discussed further in Section 5.4.2.

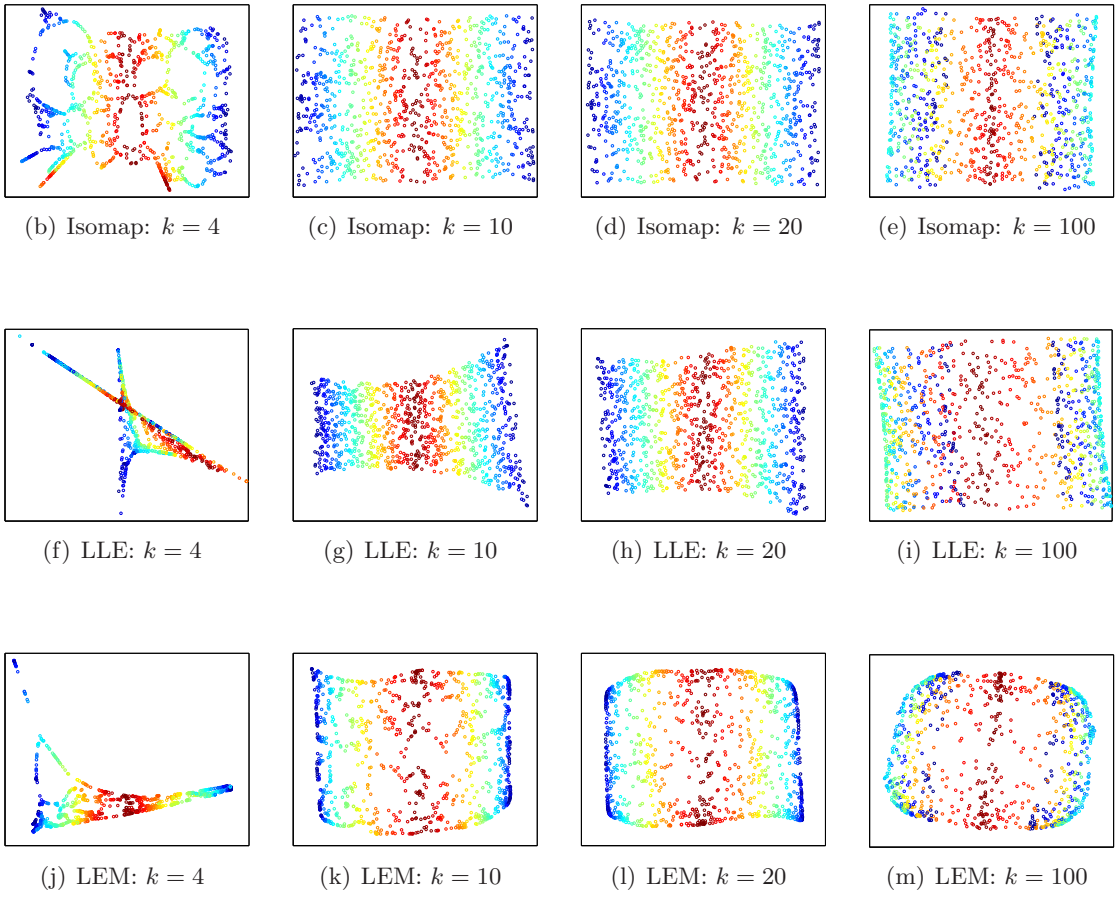
Computational complexity

The practical applicability of a dimensionality reduction method is greatly affected by its computational cost. In Table 4.1, the complexity of the most expensive computational part of each dimensionality reduction method discussed above is shown. The value m in Table 4.1 represents the sparsity of the matrices used in the computation of each method. Specifically, m is the ratio of nonzero elements to the total number of elements in the sparse matrix. Greater sparsity reduces the computational complexity of the required eigenanalysis. The value N denotes the number of data samples. The number of landmark points used in L-Isomap is represented as n.

It can be seen that the nonlinear dimensionality reduction methods are computationally demanding, with a complexity that is the number of data samples, N, squared or cubed. Consequently it may be infeasible to run such algorithms on large data sets. However, approaches have been proposed to overcome this limitation. For example, the L-Isomap algorithm, as described previously, offers a means to reduce the computational complexity



(a) Original S–curve data set



(b) Isomap: $k = 4$ (c) Isomap: $k = 10$ (d) Isomap: $k = 20$ (e) Isomap: $k = 100$

(f) LLE: $k = 4$ (g) LLE: $k = 10$ (h) LLE: $k = 20$ (i) LLE: $k = 100$

(j) LEM: $k = 4$ (k) LEM: $k = 10$ (l) LEM: $k = 20$ (m) LEM: $k = 100$

Figure 4.13: Effect of the number of nearest neighbours k on manifold learning. Two-dimensional embeddings resulting from the application of (b–e) Isomap, (f–i) LLE, and (j–m) Laplacian eigenmaps on $N = 2000$ data points sampled from a three-dimensional (a) S-curve structure with different numbers of nearest neighbours, k, used. If k is too small, leftmost column, or too large, rightmost column, the manifold learning methods fail to discover the underlying two-dimensional manifold.

Method	Computational Complexity
PCA	$O(D^3)$
MDS	$O(N^3)$
Isomap	$O(N^3)$
L-Isomap	$O(n^2N)$
LLE	$O(mN^2)$
LEM	$O(mN^2)$

Table 4.1: Computational complexities of dimensionality reduction methods, adapted from van der Maaten et al. (2009).

of the standard Isomap algorithm.

Out-of-sample embedding

Out-of-sample embedding refers to the embedding of new high-dimensional feature vectors into an existing low-dimensional space. Some dimensionality reduction methods possess this ability, whereas others require so-called out-of-sample extensions in order to embed new high-dimensional data points. In linear methods, such as PCA, out-of-sample embedding can be achieved by simply applying the same transformation that was used to embed the original high-dimensional data. However, for some nonlinear dimensionality reduction methods, including those detailed above, such an approach is not possible. This is due to the fact that these algorithms use a batch processing technique for which a parametric out-of-sample embedding method is unavailable. As a result, a number of non-parametric out-of-sample extensions have been developed (Bengio et al., 2004; Law and Jain, 2006). Non-parametric methods do not provide all the parameters needed to embed new data, in contrast to parametric methods. These non-parametric extensions estimate the transformation of new high-dimensional data to the previously created low-dimensional space.

The experiments performed for this dissertation, discussed in Chapters 6 and 7, did not use out-of-sample extensions. Instead, all of the required data was transformed into the low-dimensional space in one batch, so there was no requirement for out-of-sample data to be embedded. The motivation for this approach is that the main focus of this work is to examine the inherent low-dimensional manifold structure of speech and the performance of the manifold learning techniques applied to speech data; an evaluation of the performance of the out-of-sample extensions is beyond the scope of this work.

However, this batch processing approach is limited in that it may only be used in

applications where all of the data to be embedded is available at the same time. For example, this approach would not be suitable for use as a feature extraction front-end in a typical ASR task. In such a task the ASR system is typically trained on a large training set of speech feature vectors and later tested on previously unseen, out-of-sample, speech feature vectors. Using a dimensionality reduction method it would be possible to produce low-dimensional feature vector representations of the training speech utterances as they are typically available as a single batch. These low-dimensional features could then be used to train the ASR classifier. However, classification of unseen test speech would not be possible without an out-of-sample extension as there would be no means to map the testing speech data to the same low-dimensional feature space as the training speech.

Local vs. global methods

As discussed in brief in Section 4.3.2 above, the LLE and Laplacian eigenmaps methods are local methods. These methods aim to preserve proximity relationships between data points. This is in contrast to the Isomap algorithm that aims to preserve global geodesic structure. LLE and Laplacian eigenmaps are also known as spectral embedding methods, due to the fact that the low-dimensional embedding is produced by solving a sparse eigenvalue problem under the unit covariance constraint (4.32). However, a consequence of this constraint is that the aspect ratio of the underlying manifold is distorted (Bo et al., 2008). This problem is not encountered when using global methods, such as Isomap, which thus result in an embedding closer to the true underlying manifold. This property is clearly illustrated by the Swissroll example shown in Figure 4.11 above.

4.4 Previous applications to speech

Chapter 3 provides a review of existing literature investigating the inherent dimensionality of speech. A number of these studies employed dimensionality reduction methods to explore the dimensionality of the speech space. However, dimensionality reduction methods have also been used, and continue to be used, in a wide range of speech processing applications. We briefly review some of the relevant literature here.

Linear methods

Linear dimensionality reduction methods have been used in a myriad of previous speech processing problems including: feature transformation for improved automatic speech recognition performance (Eisele et al., 1996; Somervuo, 2003a; Wang and O'Shaughnessy, 2003; Schuster et al., 2005), speaker adaptation (Malayath et al., 1997; Kuhn et al., 1998), data compaction (Beyerbach and Nawab, 1991), and speech analysis (Plomp et al., 1967; Pols et al., 1969; Klein et al., 1970; Pols, 1971; Pols et al., 1973; Pijpers et al., 1993). A number of linear dimensionality reduction methods have also been successfully applied to speaker recognition in the past, including: LDA (Jin and Waibel, 2000), independent component analysis (Jang et al., 2001) and PCA (de Lima et al., 2002).

Manifold learning methods

In addition to this interest in applying linear methods to speech, a number of exploratory studies have recently applied manifold learning methods to speech data. The majority of these studies were conducted during the same period of time we were conducting our own investigations.

For example, to demonstrate the performance of the Laplacian eigenmaps algorithm, Belkin and Niyogi (2003) applied the algorithm to $N = 685$ 256-dimensional log Fourier coefficient feature vectors extracted from a single sentence of speech. In the resulting two-dimensional embedding, feature vectors were clustered into broad phone classes, for example: vowels, fricatives and, plosives.

Similarly, LLE has also been applied in an exploratory study to visualise speech data in a low-dimensional space. Jain and Saul (2004) applied both PCA and LLE to produce two-dimensional embeddings of log-power Fourier spectra feature vectors extracted from natural speech phones. The phones compared were limited in number, with the following phones compared: 'aa' and 'ae'; 'ay' and 'ey'; and 'p', 't', and 'k'. In the first two cases the vowels were found, based on visual inspection, to be separated in the two-dimensional embeddings produced by LLE, while PCA failed to separate them. Both methods failed to separate the 'p', 't', and 'k' phones; however, in this case visual inspection of the low-dimensional space resulting from LLE was found to reveal some phone-related structure.

Hegde and Murthy (2004) compared MFCCs and modified group delay features, dis-

cussed in Section 5.3, in a number speaker classification experiments. In order to visualise the speakers the investigators also applied the LLE and Isomap algorithms to the original high-dimensional feature vectors to yield three-dimensional spaces. However, it is difficult to draw concrete conclusions regarding the performance of the algorithms in this case, given the insufficient visualisation results and corresponding discussion presented. In a similar study, Hegde et al. (2005) compared the performance of MFCCs and modified group delay features in experiments aiming to identify the language of an utterance spoken by an unknown speaker. In order to visualise clusters of data from different languages, Hegde et al. employed the Isomap algorithm to produce two-dimensional embeddings of the high-dimensional data. Varying degrees of separability between the different languages investigated were found in the resulting embeddings. It is worth noting that in both of these studies the manifold learning algorithms were only used to facilitate visualisation—they were not used to process the data prior to classification.

Also, You et al. (2006) developed a manifold learning algorithm they term 'enhanced Lipschitz embedding' and applied this in an application to recognise various classes of emotion in spoken utterances. The manifold learning method was used to embed 64-dimensional acoustic features—48 of these dimensions related to prosody and 16 to formant frequencies—into a 6-dimensional embedding space. A support vector machine (SVM) classifier was then used to classify spoken utterances into various emotional states. The manifold learning approach yielded improvements of 5–26% over the traditional linear approaches of PCA, LDA, and feature selection.

4.5 Summary and conclusions

In many fields, ranging from genomics to speech processing, researchers are frequently faced with high-dimensional data sets. The dimensionality of these data sets has generally increased over the years along with technological advances such as increased and faster data storage media and more detailed measuring devices. However, there are a number of inherent difficulties in processing high-dimensional data, such as the oft cited curse of dimensionality.

A number of different methods have been proposed to alleviate these problems by reducing the dimensionality of the data while retaining the, possibly latent, significant

information within it. As discussed in Section 4.3 these methods may be categorised as either linear or nonlinear. The linear PCA and MDS methods and nonlinear Isomap, L-Isomap, LLE, and Laplacian eigenmaps methods are described above. The ability of the nonlinear manifold learning methods to outperform PCA and discover underlying nonlinear manifold structure is demonstrated in the examples shown in Figures 4.11 and 4.12.

Dimensionality reduction methods have successfully been applied to speech in a large number of previous studies, as reviewed briefly in Section 4.4. Also, as discussed in Chapter 3, it has previously been shown that speech may lie on a low-dimensional manifold nonlinearly embedded in high-dimensional space. Thus, manifold learning methods could potentially be useful in speech analysis, to study the underlying structure of speech, and in other speech applications such as producing features containing significant discriminatory information for use in phone classification tasks. The following chapters build upon the theory, methods, and studies described in this dissertation thus far by experimenting with and analysing the potential applications of these manifold learning methods in speech processing.

Chapter 5

A Framework for Reducing the Dimensionality of Speech

This chapter presents our proposed methodology for applying dimensionality reduction methods to high-dimensional speech data and describes the means by which the low-dimensional embeddings produced may be evaluated. We begin by outlining the aims of this approach. Following this, detailed descriptions of the constituent parts of our approach are provided, these include: speech data preprocessing and parametrisation; application of dimensionality reduction methods; and evaluation procedures.

5.1 Proposed approach

As discussed in Chapter 1 we aim to investigate the underlying low-dimensional structure of speech. We are particularly interested in exploring the hypothesis that speech has a low-dimensional structure that is *nonlinearly* embedded in high-dimensional feature space. This hypothesis is motivated by the previous studies reviewed in Chapter 3. Investigation of this hypothesis requires methods capable of discovering intrinsic nonlinear structure in a data set. Thus, we propose applying a number of manifold learning methods, discussed in the previous chapter, to speech data in order to examine the underlying structure.

Our approach involves computing a high-dimensional feature representation of some chosen set of speech, applying both linear and nonlinear dimensionality reduction methods to produce low-dimensional embeddings of these high-dimensional features, and finally evaluating the performance of the dimensionality reduction methods. This approach is de-

signed to enable comparisons of the performance of linear and nonlinear dimensionality reduction methods to determine whether any one method is capable of discovering information which other methods cannot. Also, by varying the dimensionality of the embeddings produced, the amount of meaningful information captured at varying dimensionalities can be evaluated. For example if a large amount of meaningful information was present in the three-dimensional embedding of a data set, produced by some dimensionality reduction method, and an insignificant increase in information content was observed as the target dimensionality was increased from three up to the original dimensionality, D, it would suggest that the data has an intrinsic three-dimensional structure.

One further problem addressed in our approach is how to evaluate the effectiveness of the dimensionality reduction methods, or analogously, how to define the term 'meaningful information' as used above. The problem of evaluation is approached in two ways:

- Visualisation: Two- and three-dimensional embeddings produced by the dimensionality reduction techniques are visually inspected in order to determine if any characteristics of the original speech data are evident in the low-dimensional embeddings. In particular, several characteristics of specific significance to speech are investigated, including: speaker characteristics, prosodic information, and phone specific information such as formant values.

- Classification: In order to perform a more rigorous, objective evaluation of the information contained in any particular low-dimensional feature set a number of phone classification and speaker identification experiments are performed. The aim of these experiments is to evaluate how well phones and speakers, respectively, are separated and clustered within any particular feature space. The classification performance therefore provides a measure of how much 'meaningful information' is retained in the feature space.

In Chapters 6 and 7, we apply our approach to data from two distinct speech corpora—one synthetic, one natural. Using synthetic speech facilitates the analysis of signals with known and controllable characteristics and allows us to determine the degree to which these characteristics are retained after dimensionality reduction. Experiments are conducted on synthetic speech data to provide an insight into the underlying dimensionality of speech and applicability of the manifold learning methods to speech. However, synthetic speech

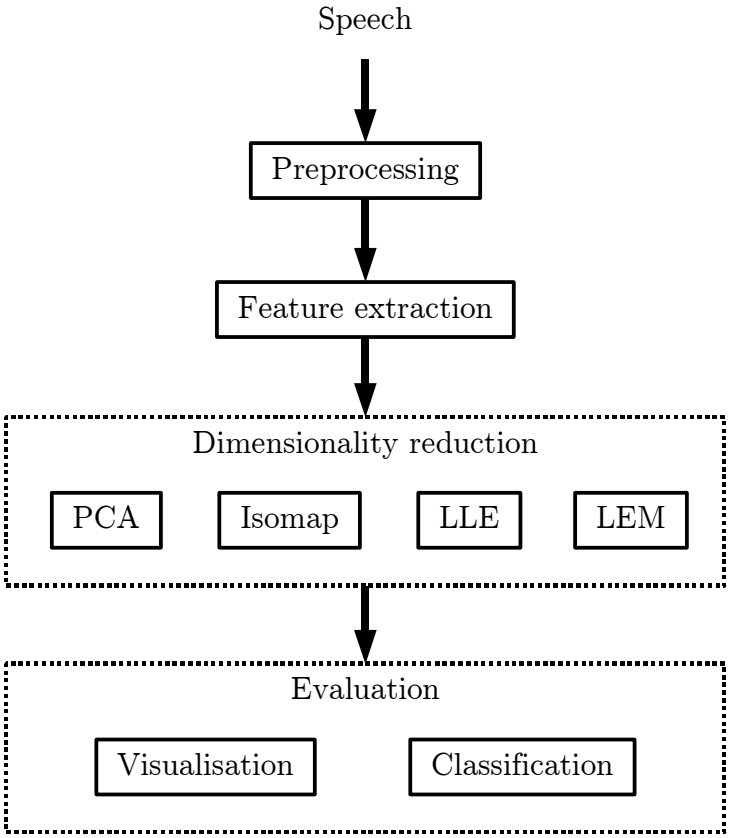


Figure 5.1: Proposed approach to applying and evaluating the dimensionality reduction methods.

is, by definition, only an approximation of the natural speech produced by humans. Thus, we also perform experiments on a corpus of natural speech data containing the types of prosodic and phonetic variation found in the real world. This enables us to draw conclusions about real speech that experiments on synthetic data alone cannot provide.

The framework of our approach is outlined in Figure 5.1. The constituent parts of this approach are discussed in the following sections; implementation details specific to individual corpora and/or experiments are omitted in the following sections and discussed in detail where appropriate in Chapters 6 and 7.

5.2 Preprocessing

The initial step in our approach requires that a particular chosen set of speech utterances are made available. In the case of synthetic speech, a set of speech utterances are generated to meet the requirements of the experiment in question. Our speech synthesis approach

is described in Section 6.2. Alternatively a suitable set of utterances may be taken from a natural speech corpus, as detailed in Section 7.1. Depending on the experiment to be performed or desired application the speech chosen may be: from a single speaker or multiple speakers; from an individual phone or many phones.

The speech signals to be analysed are then pre-emphasised by applying a filter of the form,

$$P(z) = 1 - pz^{-1} \ , \tag{5.1}$$

where typically $0.95 \leq p \leq 1$. This increases the magnitude of the high frequency spectrum with respect to the low frequency spectrum. The purpose of applying such a filter is to reduce the spectral contributions of lip radiation and the larynx (Deller Jr. et al., 2000). Reduction of these contributions facilitates more accurate estimation of the shape of the vocal tract, described in Section 2.1, a key part the speech production system.

Next, a series of short overlapping frames are extracted from the speech signals. This framing is based on the common assumption that the characteristics of the speech production system vary relatively slowly. Thus, for a short frame of speech, typically 20–40 ms (Quatieri, 2002), the speech signal can be assumed to be stationary, e.g. the vocal tract shape, glottal source input, and noise do not vary considerably over such a short-time interval. Over longer intervals speech signals are unlikely to be stationary due to the variation of vocal tract and source signal that are essential properties of speech production. Many analysis techniques, such as the Fourier transform, cannot be applied to nonstationary signals—hence the motivation for this framing. Individual frames of speech are extracted from the original speech signal by moving a sliding window, typically of duration 20–40 ms, through the speech signal extracting all samples within the window. This window slides at a frame rate that is chosen to be of a sufficiently short time interval to capture dynamics within the speech signal, e.g. 10–20 ms (Quatieri, 2002).

The window shape used in our approach is the common Hamming window (Deller Jr. et al., 2000),

$$w_n = \begin{cases} 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), & n = 0, 1, \ldots, N-1 \\ 0, & n \text{ otherwise} \ . \end{cases} \tag{5.2}$$

The reason for using a window function such as this is to reduce abrupt discontinuities

at the edges of the analysis window and thus prevent spectral leakage in any subsequent Fourier analysis. This particular window shape has been chosen for use in our approach as it is suitable for the computation of features derived from the magnitude spectrum and features derived from the phase spectrum. The Hamming window has frequently been shown in the literature to be suitable for preprocessing speech prior to the computation of MFCC features from the short-time magnitude spectrum. This window shape has also successfully been used in the computation of features derived from the short-time phase spectrum based on the modified group delay function[1] (Hegde, 2005; Hegde et al., 2007b).

Figure 5.2 displays each of the above preprocessing steps applied in sequence to a sentence of real speech taken from the TIMIT database[2]. The result of this procedure is a data set of windowed speech frames of equal length from which suitable high-dimensional features may be extracted. This feature extraction procedure is detailed in the following section.

5.3 Feature extraction

In a wide range of speech applications it is desirable to transform the raw speech signal into a form in which the most important signal characteristics are easily accessible. This transformed signal should facilitate further processing and extraction of the key characteristics of the speech signal. Over the decades a large number of such speech signal representations have been proposed. In recent times Fourier transform based representations have seen prevalent use in various speech applications, such as ASR and speech coding.

Conventionally, features are computed based on the magnitude spectrum of the short-time Fourier transform. One such feature set, the psychoacoustically motivated MFCCs, have proven to be one of the most successful and have also been demonstrated to outperform other popular features, including LPCs and PLPs, in ASR tasks (Davis and Mermelstein, 1980; Milner, 2002). Due to the prevalence of MFCC features and the proven

[1] While our choice of Hamming window is based on that used by Hegde et al. (2007b) in the computation of the *modified* group delay function, the interested reader may wish to note that Bozkurt et al. (2004) present results indicating that Hanning-Poisson windows are preferable in group delay function based analysis methods. However, Bozkurt et al. also show that the effect of window shape on the extraction of phase information is comparatively less important than the size of window used.

[2] The TIMIT database is described in detail in Section 7.1. This particular sentence contains the utterance "She had your dark suit in greasy wash water all year," spoken by speaker DR1\FCJF0, and is from the training portion of the corpora.

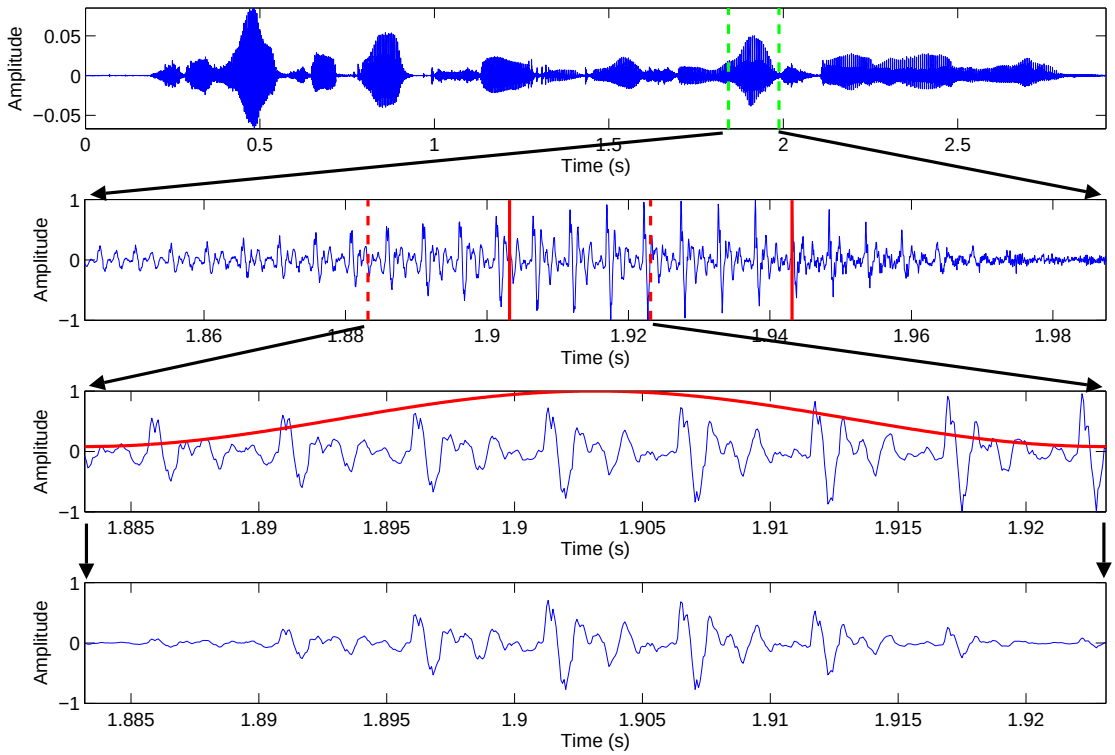


Figure 5.2: Illustration of the preprocessing procedure applied to a sample of speech from TIMIT. The topmost plot shows an entire utterance with a single phone, /ɔ/, delineated by dashed lines. Below this the single phone is shown after extraction from the original utterance and application of preemphasis; two 40 ms frames, with an overlap of 20 ms, are indicated by the dashed and solid lines. The first (dashed lines) frame is shown below this with the Hamming window overlaid. The bottom plot shows the result of applying the Hamming window producing a windowed frame of speech, ready for feature extraction.

ability of these features to effectively parametrise the important characteristics of the speech signal we choose to use MFCC feature vectors as high-dimensional inputs to the dimensionality reduction methods.

The short-time Fourier transform (STFT) can be decomposed into a magnitude spectrum and phase spectrum; the latter is commonly discarded and features are derived solely from the magnitude spectrum as in the case of MFCCs above. This is true of many speech-related tasks including ASR, speech coding, speech enhancement, and voice conversion. This has also been the case in a large number of previous studies of the intrinsic low-dimensional structure of speech sounds, as reviewed in Chapter 3. In contrast to this conventional belief that phase information may be ignored, a number of recent studies have demonstrated the importance of the phase spectrum in human speech perception (Paliwal and Alsteris, 2005) and ASR (Bozkurt and Couvreur, 2005). To ex-

ploit the relevant information present in the phase spectrum Murthy and Gadde (2003) proposed a feature set, modified group delay features (MODGDF), based on a modified version of the group delay function. This feature set has been demonstrated to be useful in several speech-related tasks, including: automatic identification of phoneme, syllable, speaker, and language identification (Hegde et al., 2007b); and join cost calculation in concatenative speech synthesis (Kirkpatrick et al., 2006). We propose applying a number of dimensionality reduction methods to MODGDF representations of speech signals in order to examine if any useful low-dimensional structure exists.

By examining the intrinsic low-dimensional structure of both magnitude- and phase-derived features and evaluating the performance of low-dimensional embeddings of these features we aim to determine if features derived from the phase spectrum are indeed useful, as recently proposed, and if they have a similar intrinsic dimensionality to features derived from the magnitude spectrum. A further motivation for choosing these two, differently derived, feature types is to investigate whether or not they contain complementary information. Such complementary information, if present, could be exploited in speech applications such as ASR. Both of the chosen feature types are described in more detail in the following sections.

Furthermore, it is common in many speech applications to append *delta* features, encoding temporal information, to the *static* features mentioned above. As a result, these features are also studied from a dimensionality reduction perspective. These delta features are discussed in Section 5.3.3.

5.3.1 Mel frequency cepstral coefficients

Proposed by Davis and Mermelstein (1980), MFCCs have seen widespread use in many speech-related tasks, particularly ASR and speaker recognition. One key aspect of this feature representation is its exploitation of psychoacoustics, specifically the human auditory system's ability to perceive frequencies. This is achieved by applying a series of overlapping triangular filters spaced according to the mel scale, discussed in Section 2.2.1, to the STFT magnitude spectrum. These filters, illustrated in Figure 5.3, mimic the human auditory system's ability to distinguish differences in low frequency ranges better than differences at higher frequencies. The MFCC implementation used in this work, after

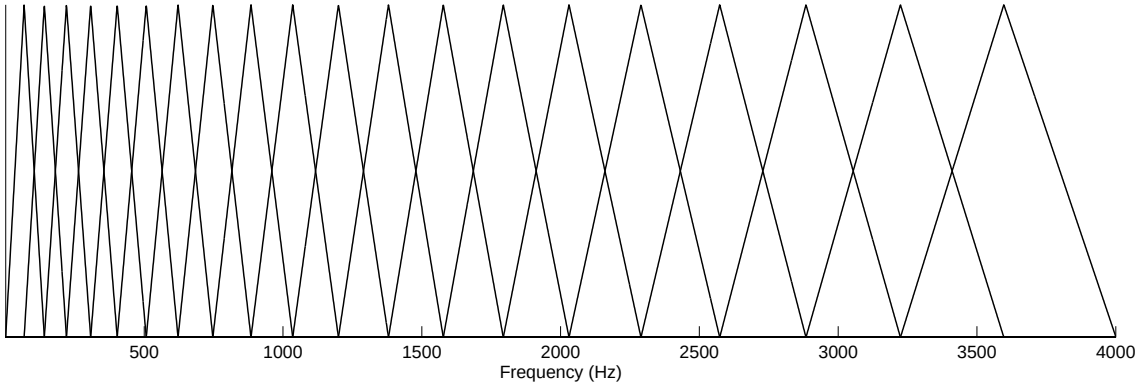


Figure 5.3: A triangular filter bank where each filter is spaced according to the mel scale, linearly for low frequencies and logarithmically for higher frequencies.

Huang et al. (2001), is illustrated in Figure 5.4 and described in detail in the following section[3].

Computation of mel frequency cepstral coefficients

First the discrete STFT, $X(\omega)$, of the windowed speech waveform is computed:

$$X(\omega) = \sum_{n=0}^{N-1} x(n) e^{(-j2\pi\omega n/N)}, \quad \omega = 0, 1, \ldots, N-1 \ , \tag{5.3}$$

where N is the length of the discrete STFT.

A mel scale filter bank is then applied to the square of the magnitude of the STFT, $|X(\omega)|^2$. A mel scale filter bank consists of a number of triangular-shaped filters. The center frequency and band edges of each filter are spaced linearly below 1 kHz and logarithmically above this, to match the mel scale. A mel scale filter bank consisting of 20 filters is displayed in Figure 5.3. Each of these M filters, $m = 1, 2, \ldots, M$ is computed as (Huang et al., 2001):

$$H_m(\omega) = \begin{cases} 0, & \omega < f_{m-1} \\ \dfrac{(\omega - f_{m-1})}{(f_m - f_{m-1})}, & f_{m-1} \leq \omega \leq f_m \\ \dfrac{(f_{m+1} - \omega)}{(f_{m+1} - f_m)}, & f_m \leq \omega \leq f_{m+1} \\ 0, & \omega > f_{m+1} \ , \end{cases} \tag{5.4}$$

[3]Many different MFCC implementations have been developed over the years, for a comparative evaluation of a number of these implementations refer to Ganchev et al. (2005).

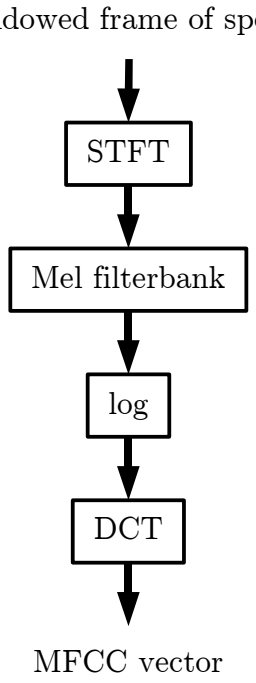


Figure 5.4: The steps involved in the computation of MFCC features.

where $\omega = 0, 1, \ldots, N$. The edges of each triangular filter, f_m, are computed as:

$$f_m = \left(\frac{N}{F_s}\right) B^{-1}\left(B(f_l) + m\frac{B(f_h) - B(f_l)}{M+1}\right) \quad , \tag{5.5}$$

where the lowest, f_l, and highest, f_h, frequencies of the filter bank are defined in Hz, as is the sampling frequency F_s. The function $B(f)$—defined in Equation (2.4)—above converts a frequency value in Hz, f, to the equivalent value on the mel scale. The function $B^{-1}(q)$ performs the inverse operation, converting a mel scale value, q, to Hertz as follows:

$$B^{-1}(q) = 700\left(e^{(q/1125)} - 1\right) \quad . \tag{5.6}$$

Following this the logarithm of the output of the mel filter, $m = 1, 2, \ldots, M$, is computed as

$$S(m) = \log\left(\sum_{\omega=0}^{N-1} |X(\omega)|^2 H_m(\omega)\right) \quad . \tag{5.7}$$

The purpose of this logarithm is to, theoretically, make the vocal tract and glottal excitation components of the speech signal, as discussed in Chapter 2, linearly separable. The

discrete cosine transform (DCT)[4] is then applied to the M log-filter outputs to produce the mel *cepstrum*,

$$\text{MFCC}(\omega) = \sum_{m=0}^{M-1} S(m) \cos\left(\frac{\pi}{M}\left(m + \frac{1}{2}\right)\omega\right) \quad \omega = 0, 1, \ldots, M-1 \ . \tag{5.8}$$

In this representation the slowly varying vocal tract contribution is represented in the low quefrency range, the lower coefficients, whereas the more rapidly varying excitation component occupies the higher quefrency range, the higher coefficients. Generally the first 12–20 coefficients are retained and the remainder discarded, as the higher coefficients contain little relevant information.

The DCT can be viewed as a form of dimensionality reduction[5] as it results in most of the information being contained in the first few mel cepstrum coefficients. In fact, the DCT is an approximation of PCA (or KLT), discussed in Section 4.3.1. Stated informally, a DCT uses a sum of orthogonal cosine basis functions to approximate a series of data points. A DCT differs from PCA in that it uses a set of fixed basis vectors, whereas the basis vectors used in PCA—the eigenvalues of the data covariance matrix—are data dependent. The equation for the DCT basis vectors is provided in Equation (5.8). The DCT is widely used for lossy compression of images and signals as most of the significant information is found in the first few components of the DCT. While PCA is the optimal linear transform for mapping from a high-dimensional space into a lower dimensional space, the DCT has been shown to be equivalent to PCA for certain types of data (Hamidi and Pearl, 1976).

5.3.2 Modified group delay features

Spectral features parametrising speech are conventionally extracted from the magnitude spectrum, derived from the STFT of the speech signal, such as those described in the previous section. The phase spectrum also resulting from the STFT is conventionally ignored due to the common belief that the phase spectrum does not play a significant part

[4]Here the DCT is equivalent to an inverse discrete Fourier transform as $S(m)$ is real and symmetric.

[5]In this case the DCT is in effect reducing the dimensionality of the mel cepstrum of a single frame of speech. This differs greatly to the dimensionality reduction experiments performed in Chapters 6 and 7. The aim of these experiments is to reduce the dimensionality of a 'speech space' containing a large number of frames of speech, individually represented by MFCCs or MODGDFs, in order to study the space's intrinsic dimensionality.

in human auditory perception over the short time frames used in STFT analysis. However, a number of recently performed studies have shown that the short-time phase spectrum is useful in human speech perception (Paliwal and Alsteris, 2005) and ASR (Bozkurt and Couvreur, 2005).

It is, however, difficult to extract useful features from the STFT phase spectrum due to problems with phase unwrapping and zeros of the signal's z-transform close to the unit circle (Yegnanarayana and Murthy, 1992). The group delay function (GDF) (Oppenheim and Schafer, 1975) has been used to represent the phase spectrum in a number of speech processing applications in the past (Yegnanarayana and Murthy, 1992). However, the spiky nature of the GDF and the fact that it can become undefined due to zeros of the z-transform of the signal that are close to the unit circle—caused by windowing effects, pitch epochs, and noise (Yegnanarayana and Murthy, 1992; Murthy and Gadde, 2003)—make it problematic in speech applications. In answer to these issues Murthy and Gadde (2003) proposed a feature set, MODGDF, derived from a modified version of the group delay function. This modification suppresses zeros of the z-transform of the signal that are close to the unit circle and cause the group delay function to become undefined.

Computation of modified group delay features

The GDF is the negative derivative of the phase spectrum, $\theta(\omega)$, with respect to frequency, ω:

$$\tau(\omega) = -\frac{d(\theta(\omega))}{d\omega} \quad . \tag{5.9}$$

The GDF can be computed from the speech signal $\mathbf{x}$ as follows (Oppenheim and Schafer, 1975):

$$\tau(\omega) = \frac{X_R(\omega)Y_R(\omega) + Y_I(\omega)X_I(\omega)}{|X(\omega)|^2} \quad , \tag{5.10}$$

where $X(\omega)$ and $Y(\omega)$ denote the Fourier transforms of $x(n)$ and $nx(n)$, respectively. The real and imaginary parts of the Fourier transform are indicated by the subscripts R and I.

As mentioned previously, the GDF is undefined when the roots of the signal's z-transform are close to the unit circle. The modified group delay function (MGDF) (Murthy and Gadde, 2003) overcomes this problem by substituting $S(\omega)$, a cepstrally smoothed

version of the spectrum $|X(\omega)|$, in place of the same:

$$\tilde{\tau}(\omega) = \frac{X_R(\omega)Y_R(\omega) + Y_I(\omega)X_I(\omega)}{S(\omega)^2} \quad . \tag{5.11}$$

A further two parameters, α and γ, were introduced by Murthy and Gadde (2003) to reduce the spiky nature of the formant peaks, relative to the magnitude spectrum, giving the final MGDF definition:

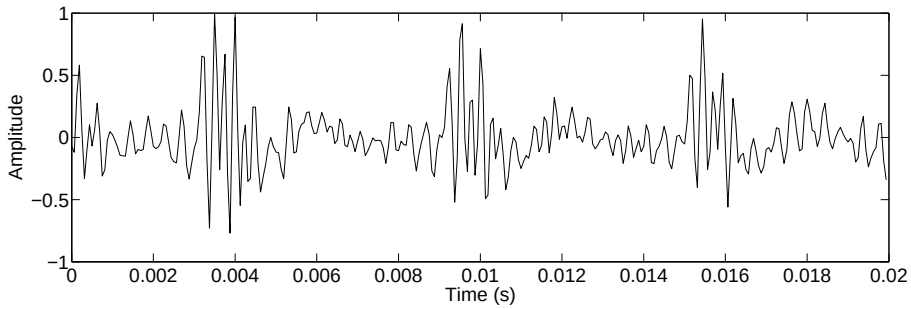
$$\tilde{\tau}_{\alpha,\gamma}(\omega) = \frac{\tilde{\tau}_\gamma(\omega)}{|\tilde{\tau}_\gamma(\omega)|} |\tilde{\tau}_\gamma(\omega)|^\alpha \quad , \tag{5.12}$$

where

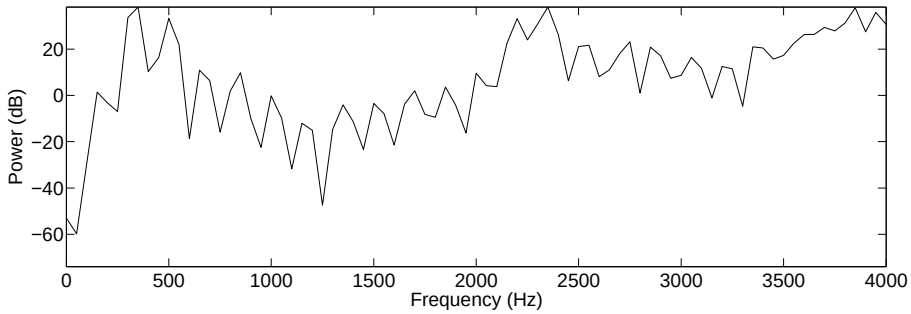
$$\tilde{\tau}_\gamma(\omega) = \frac{X_R(\omega)Y_R(\omega) + Y_I(\omega)X_I(\omega)}{S(\omega)^{2\gamma}} \quad . \tag{5.13}$$

In all the MGDF computations in this study the parameters are set as $\alpha = 0.4$ and $\gamma = 0.9$. A lifter window of length 8 is used for cepstral smoothing. Details of these values are discussed further by Hegde et al. (2007b).

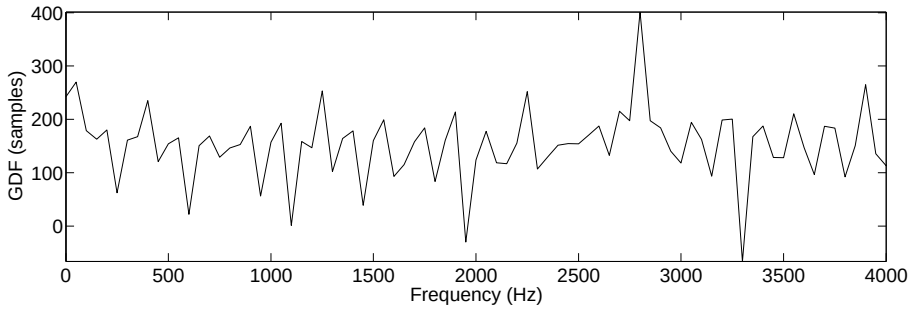
A comparison of the magnitude spectrum, GDF, and MGDF for a single frame of speech is shown in Figure 5.5. Figure 5.5(a) shows the time domain representation of the speech frame, extracted from an /i/ vowel. The magnitude spectrum is provided in Figure 5.5(b). Figures 5.5(c) and 5.5(d) show the GDF and MGDF representations, respectively, which show the time-domain delay, in samples, for each frequency component of the signal. As discussed above, zeros of the signals z-transform which are close to the unit circle manifest themselves as spikes in the signal's GDF. These zeros are principally caused by the source signal and not the vocal tract. The 'modifications' of the MGDF, discussed above, suppress these spikes allowing more accurate estimation of the vocal tract configuration. It can be seen that the magnitude spectrum and MGDF capture similar information with peaks in the spectral envelope evident at locations corresponding to the typical formant frequencies, whereas the GDF does not. This clearly shows the improvement offered by the MGDF over the GDF. The difference in the vertical scale of the GDF and MGDF representations is due in part to the MGDF's cepstral smoothing removing large spikes from the GDF and also due to the α parameter in the MGDF which is a compression factor. However, this difference is unimportant in this case as we are concerned with the features in the plots.



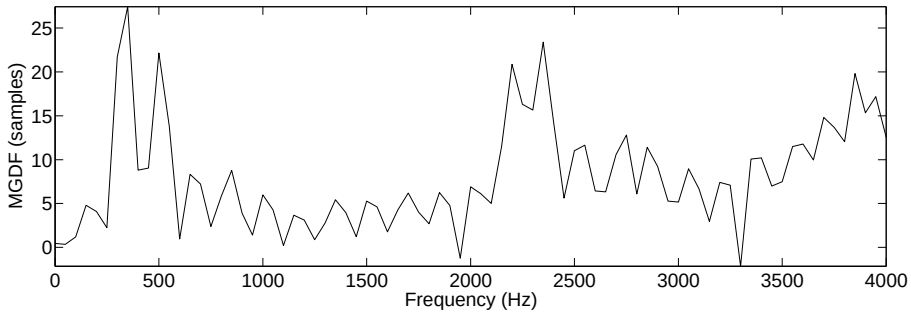
(a) Time domain signal



(b) Power spectrum



(c) Group delay function



(d) Modified group delay function

Figure 5.5: Comparison of magnitude and phase spectrum representations of a frame of speech taken from an /i/ vowel sound.

To produce a feature set more suitable for applications such as speech recognition cepstral coefficients can be computed from the MGDF using a DCT, in a similar manner to the conventional MFCC computation. These features, MODGDF, are computed as

$$\text{MODGDF}(\omega) = \sum_{n=0}^{N-1} \tilde{\tau}_{\alpha,\gamma}(n) \cos\left(\frac{\pi}{N}\left(n+\frac{1}{2}\right)\omega\right) \quad \omega = 0, 1, \ldots, N-1 \ . \tag{5.14}$$

5.3.3 Dynamic features

As previously described we, as is conventional, assume the speech signal is stationary over short-time periods and extract features from windows of length 20–40 ms. However, these *static* features do not encode the time-varying information contained within the speech signal. As a result, time derivatives, also known as *delta* features, are commonly appended to the static features in order to encode information about the spectral changes occurring between windows of speech. The delta of the nth feature coefficient at time t, $\Delta_{n,t}$, is computed as follows (Young et al., 2000)

$$\Delta_{n,t} = \frac{\sum_{\theta=1}^{\Theta}(\theta c_{n,t+\theta} - \theta c_{n,t-\theta})}{2\sum_{\theta=1}^{\Theta}\theta^2} \ , \quad n = 0, 1, \ldots, D \tag{5.15}$$

where $c_{n,t+\theta}$ and $c_{n,t-\theta}$ are the corresponding static coefficients. The parameter Θ controls number of windows, forward and backward in time, over which the deltas are computed. The value $\Theta = 2$ is used to compute deltas in the experiments presented in this dissertation.

5.4 Application of dimensionality reduction methods

At this point in the description of our framework we have a means of preprocessing selected speech signals and producing feature vectors representing the individual frames of the chosen speech signals. These feature vectors

$$\mathbf{c} = [c_0,\, c_1,\, c_2,\, \ldots,\, c_D]' \ , \tag{5.16}$$

are D-dimensional; that is, in the case of MFCC and MODGDF, the number of DCT coefficients retained is equal to D. For a particular set of speech signals the N associated

extracted feature vectors are concatenated to form an $N \times D$ matrix

$$\mathbf{X} = [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_N]' \quad , \tag{5.17}$$

where N is the number of speech frames, and hence also the number of feature vectors.

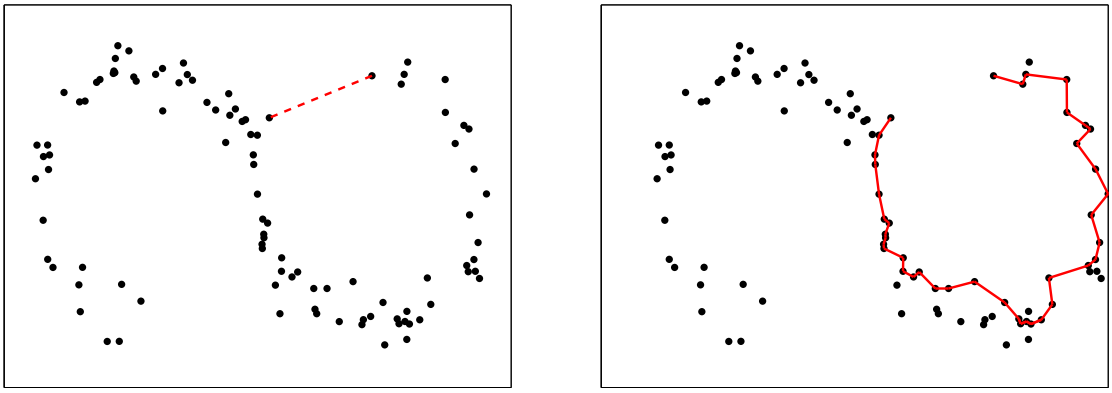
This matrix $\mathbf{X}$ may then be used as input to dimensionality reduction methods which output the $N \times d$ matrix $\mathbf{Y}$ consisting of the d-dimensional embeddings, where $d < D$, of the N original speech feature vectors. The parameters of the dimensionality reduction methods are discussed in the following sections.

5.4.1 Choice of d

The target dimensionality, d, is a parameter of each of the four dimensionality reduction methods discussed in Chapter 4. Traditionally the choice of d is influenced by two factors. Firstly, if one has some knowledge of the inherent dimensionality of the data set to be reduced, d can be set equal to that inherent dimensionality. The inherent dimensionality may have been estimated using some mathematical method, several of these methods are touched upon in Chapter 3. Alternatively, the inherent dimensionality of the data set may be determined based on knowledge of the data set. For example, if a data set were to contain measurements made of the movements of a mechanical arm one knew to have only two-degrees of freedom, a value of $d = 2$ may be appropriate.

Secondly, the choice of d may be influenced by the desired application. For example, if the dimensionality of a data set were being reduced to allow visualisation of the data then a value $0 < d < 4$ would be required. Likewise, if the data set were being reduced for the purpose of data compression, perhaps prior to transmission across a channel with limited bandwidth, the value of d would need to be set to ensure the required compression rate was achieved.

In our case we wish to make no assumption of the inherent dimensionality of speech and do not wish to rely on an inherent dimensionality estimation method. Thus, our choice of d is influenced solely by the intended applications, namely visualisation and pattern classification. For the visualisation experiments d is chosen as one, two, or three depending on the quality of the visualisation at each dimensionality. In the case of pattern classification, the experiments are designed to evaluate the amount of discriminative infor-



(a) Short-circuit (b) Geodesic

Figure 5.6: An example of short-circuiting. (a) If the number of nearest neighbours is too great, neighbours may be chosen which appear to be close, in terms of Euclidean distance, as shown by the dashed line; (b) but the 'neighbours' may be a large distance away geodesically, as shown by the solid line.

mation encoded in particular low-dimensional embeddings. The aim of these experiments is to compare the effectiveness of the different dimensionality reduction methods and also to contrast the amount of information encoded at varying dimensionalities. As a result, we require that embeddings be produced for a range of different dimensionalities from one up to the original feature vector dimensionality, $0 < d \leq D$. This allows the amount of discriminative information added as dimensionality increases to be analysed.

5.4.2 Choice of k

The second parameter is the number of nearest neighbours, k, to be used in the nonlinear dimensionality reduction algorithms: LLE, Laplacian eigenmaps, and Isomap. The sensitivity of these methods to the choice of k has been shown in a range of previous studies and thus it is important to choose an appropriate k value to ensure successful dimensionality reduction (Balasubramanian et al., 2002). The importance of choosing a suitable value for k is demonstrated in Figure 4.13, which compares the performance of each manifold learning method using various numbers of nearest neighbours. The number of nearest neighbours chosen must be large enough to detect the global structure of the data and avoid fragmenting the data into disjointed patches. However, if too many nearest neighbours are chosen the problem of short-circuiting may arise. Short-circuiting can cause the manifold learning algorithms to misinterpret the topological structure of the data and result in poor embeddings. An example of short-circuiting is shown in Figure 5.6.

However, determining a suitable value for k for a given data set and given method is currently an open problem. Several methods have been proposed that attempt to select the 'optimal' k value. These methods compute a measure of the quality of the low-dimensional embedding and select the k value that results in the best quality low-dimensional embedding, in relation to the measure. Kouropteva et al. (2002) propose one such method for choosing the number of nearest neighbours for use with the LLE algorithm. This method uses 'residual variance' (Roweis and Saul, 2000) as a measure of the quality of the embedding space. This approach has been extended by Samko et al. (2006) for selection of the 'optimal' parameter value for the Isomap algorithm. Also, Shao et al. (2007) and Shao (2008) describe methods to help choose a suitable neighbourhood size for Isomap based on the presence of short-circuits, which indicate a poor quality embedding. However, these approaches are relatively new and have not been applied, or proven to be useful, in a wide range of applications; aside from the limited testing performed in the original studies. Also, we are unaware of any such approaches for optimally determining the k value to use with the Laplacian eigenmaps technique. As a result, we have chosen to select the value of k empirically to ensure the manner in which the number of nearest neighbours is chosen is consistent for all of the nonlinear dimensionality reduction methods used. This empirical selection approach has been used in a number of other studies (van der Maaten et al., 2009).

To select the number of nearest neighbours, k, parameter for a particular nonlinear dimensionality reduction algorithm to be applied to a specific data set, the following procedure was used. The algorithm was run on the data set repeatedly, with a different value of k used on each run. The k values were constrained to the range $2 \leq k \leq 30$; this range was selected following initial exploratory experiments and is consistent with previous studies (van der Maaten et al., 2009). A different low-dimensional embedding was produced for each run. Each of these embeddings was then evaluated and the k value resulting in the best performing embedding was selected as optimal. The evaluation procedure used to determine the 'best performing' embedding varied depending on the desired application, i.e. visualisation or classification. These evaluation procedures are described in the following section.

5.5 Evaluation

There are two main aims of the evaluation component of this framework. Firstly, we wish to measure and compare the performance of the four dimensionality reduction algorithms. Secondly, as we are interested in investigating the underlying structure of speech data, we wish to examine the information retained in feature spaces of various dimensionalities in an effort to determine if speech data has a particular inherent dimensionality or low-dimensional structure.

To achieve both of these aims we require a means of assessing the quality of the low-dimensional embeddings resulting from the application of dimensionality reduction algorithms to high-dimensional features extracted from speech data. The quality of the embedding is characterised by the amount of 'meaningful information' retained after dimensionality reduction. The means by which the amount of 'meaningful information' present in the low-dimensional embeddings is assessed are detailed in the following sections.

5.5.1 Visualisation

One approach used within this framework to assess the quality of the low-dimensional embeddings is data visualisation. This involves producing one-, two-, or three-dimensional embeddings, as appropriate, and examining them to determine, subjectively, if any meaningful structure is present. As the data concerned is speech, several key sources of information within the speech signal are investigated, including: speaker characteristics, prosodic information, and phone specific information such as formant values.

Visualisations resulting from dimensionality reduction of speech data are discussed further in relation to the experiments described in Chapters 6 and 7.

5.5.2 Classification

While conducting a visual evaluation of the data may allow one to subjectively judge the quality of a low-dimensional embedding and reveal interesting patterns it is desirable to have a means of objectively measuring the quality of low-dimensional embeddings. In order to accomplish this we propose performing a number of pattern classification experiments. Assuming that the original feature vectors have been sampled from a discrete

number of classes—for example, phones—some pattern classifier can be trained on a subset of the, labelled, feature vectors. The remaining feature vectors can thus be used as unseen, unlabelled test samples that are assigned to a particular class by the trained classifier. The percentage of feature vectors assigned to the correct class, which we will refer to as the 'classification rate', provides a measure of the amount of discriminative information encoded by the feature vectors. If the feature vectors in question are the output of a dimensionality reduction method then this classification rate is a measure of the discriminative information retained by the dimensionality reduction method.

We propose to use the classification accuracy in order to compare the performance of the dimensionality reduction methods. If, for a particular classification task, the two-dimensional features produced by one dimensionality reduction algorithm yield a higher classification accuracy than features of equal dimensionality produced by a second algorithm, the former can be said to outperform the latter, producing a higher quality embedding for that particular classification task. The classification tasks performed, as detailed in Chapters 6 and 7, are designed to examine each algorithm's ability to retain several different, important sources of information within speech signals. Two types of classification task are used in this work: phone classification and speaker identification.

In all classification experiments we have performed nonlinear dimensionality reduction on a data set containing both the training and testing data. This removes the need for out-of-sample extensions to the nonlinear dimensionality reduction algorithms, as discussed in Section 4.3.4.

Phone classification

The objective of the phone classification experiments discussed in this dissertation is to evaluate how well different phones, and different classes of phones, are separated in the low-dimensional feature spaces output by the various dimensionality reduction methods. It should be noted that these speech classification, or recognition, experiments are not intended to rival current sophisticated state-of-the-art ASR approaches. Such approaches typically utilise sources of information outside that encoded within the speech feature vector; for example, language models that incorporate information about the probability of particular sequences of word, phone, or sub-phone units (Young, 1996; Young et al.,

2000). In our approach we are concerned only with the information encoded within the features resulting from dimensionality reduction and thus do not use any additional, e.g. contextual, information. This also influences the choice of classifier as we desire an algorithm which simply makes the best classification decision possible for each feature vector based on the information within the feature space.

In order to select an appropriate classifier a number of common classification algorithms were initially tested in baseline phone classification experiments. These classifiers included:

- Euclidean distance based linear discriminant function (LDF) where each class is modelled by a multivariate normal density and the covariance is estimated across all classes (Duda et al., 2000),
- K-nearest neighbour (K-NN) estimation (Duda et al., 2000),
- Gaussian mixture models[6] (GMM) (Duda et al., 2000),
- Support vector machines[6] (Vapnik, 1995; Schölkopf and Smola, 2002) with:
  - Linear kernel,
  - Polynomial kernel,
  - Radial basis function kernel.

Each classifier was tested in three different classification tasks. These tasks involved assigning 13-dimensional MFCC feature vectors, extracted from particular phones from the TIMIT speech corpus, to their correct class. The tasks involved the following classes:

Task 1 Five vowels: /ɑ/, /i/, /u/, /ɛ/, and /æ/.

Task 2 Ten vowels: /ɑ/, /i/, /u/, /ɛ/, /æ/, /ʌ/, /ə/, /ɔ/, /ɪ/, and /o/.

Task 3 Five phone classes: vowels (listed above), fricatives (/s/, /ʃ/), stops (/p/, /t/, and /k/), nasals (/m/, /n/) and, semivowels and glides (/l/, /y/).

IPA phone symbols are used above, for more details see Table 2.1. A more detailed description of the data and methodology is provided in Section 7.4. These three classification tasks were chosen to provide a thorough test of both vowel and non-vowel sounds.

[6] Appendix A presents further discussion of the GMM and SVM classifiers.

Task	Number of Centres						
	1	2	4	8	16	32	64
Five Vowel	67.200	**68.246**	67.908	67.969	65.938	65.653	65.510
Ten Vowel	48.692	**49.877**	48.400	46.338	44.295	42.811	42.716
Phone Class	42.231	**42.505**	42.231	40.659	40.703	39.978	40.076

Table 5.1: Mean classification rates (%) achieved using GMM classifiers with various numbers of centres. Mean classification rates for each of the three classification tasks are shown. Bold values indicate the maximum mean classification rate achieved in each task.

A number of the classifiers listed above have parameters which must be assigned appropriate values. In the case of the GMM-based classifier the number, Q, of mixture centres (normal distributions) to be used in the mixture must be chosen appropriately. We tested GMM classifiers with $Q = 1, 2, 4, 8, 16, 32, 64$ on each classification task. The mean classification rates for all dimensionalities $d = 1, \ldots, D$ achieved for each Q value are shown in Table 5.1. Examining Table 5.1, one can observe that the best mean classification rate is achieved for $Q = 2$ in all three classification tasks. For GMM classifiers where $Q > 2$ mean classification rate can be seen to decrease as Q increases. This is somewhat unexpected as increasing the number of mixture centres typically results in more accurate modelling of the probability densities, however this may be due to the GMMs overfitting the data as Q increases. As a result, a GMM classifer with $Q = 2$ mixture centres was used in the classifier comparison experiment detailed below.

The K-NN classification algorithm also has one parameter, namely the number of nearest neighbours, K, to use. We tested nearest neighbour classifiers with $K = 1, \ldots, 50$. The results are presented in Figure 5.7. It can be seen that the mean classification rate in each task generally increases with increasing K until the addition of further nearest neighbours no longer significantly improves performance. The K values resulting in the highest mean classification rate for each task were used in the classifier comparison experiment detailed below; i.e. for five vowel classification $K = 32$, for ten vowel classification $K = 47$, and for phone class classification $K = 26$.

It is also necessary to choose an appropriate kernel function to be used in the SVM classifier. In order to select an effective kernel, different SVM models using linear (5.18), polynomial (5.19), and radial basis function (RBF) (5.20) kernels were evaluated in a

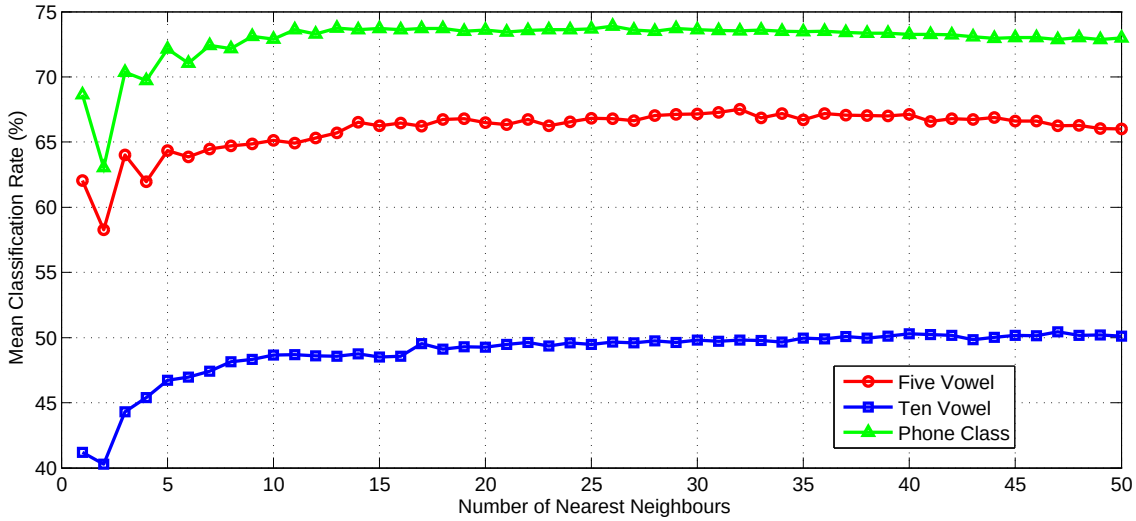


Figure 5.7: Mean classification rate vs. number of neighbours used in K-NN classification. Mean classification rates for each of the three classification tasks are shown.

number of phone classification tasks. The kernels used are given below (Hsu et al., 2009),

$$k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i' \mathbf{x}_j \quad , \tag{5.18}$$

$$k(\mathbf{x}_i, \mathbf{x}_j) = \left(\frac{1}{d} \mathbf{x}_i' \mathbf{x}_j \right)^3 \quad , \tag{5.19}$$

$$k(\mathbf{x}_i, \mathbf{x}_j) = exp\left(-\frac{1}{d} \|\mathbf{x}_i - \mathbf{x}_j\|^2 \right) \quad , \tag{5.20}$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ are feature vectors and d the feature vector dimensionality. SVM is a binary pattern classification algorithm. For our experiments it is necessary to construct a multiclass classifier. This was achieved using a one-against-one training scheme, training one classifier for every possible pair of classes. The final classification results were determined by majority voting (Schölkopf and Smola, 2002).

The results of the classifier comparison experiments are shown in Figure 5.8 and Table 5.2. The dimensionality of the feature vectors used in the experiment vary from 1 to 13—the original, full dimensionality. This results in 13 different classification rates for each classifier. Figure 5.8 shows the classification rate vs. feature dimension for each type of classifier in each classification task. In the classification tasks, the classification rate was found to increase with dimension, this is to be expected as adding more information should improve the classifiers ability to assign a feature vector to the correct class. However, it can be seen that the amount by which the classification rate improves, from one

Task	Classifier					
	LDF	SVM-Linear	SVM-Polynomial	SVM-RBF	K-NN	GMM
Five Vowel	67.169	67.754	67.631	**69.323**	67.508	68.246
Ten Vowel	48.738	49.092	50.046	**51.031**	50.446	49.877
Phone Class	64.396	69.560	73.176	**74.165**	73.890	42.505
Mean	60.101	62.136	63.618	**64.840**	63.948	53.543

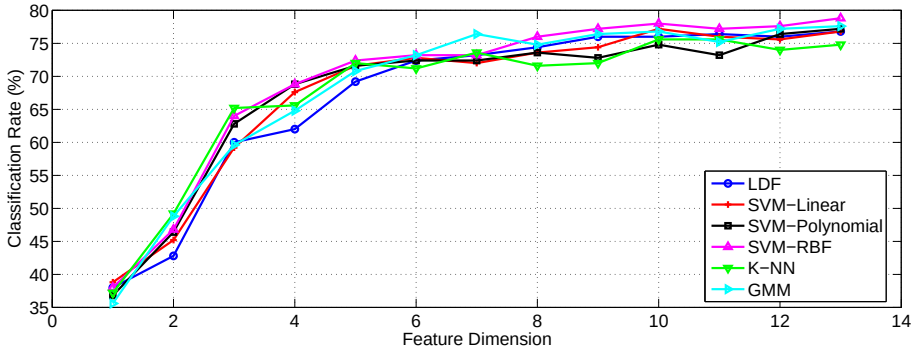
Table 5.2: Mean classification rate (%) for each type of classifier. Mean classification rates for each of the classification tasks and the mean over all three tasks are shown. Bold values indicate the maximum mean classification rate achieved in each task.

dimensionality to the next, decreases as the dimensionality is increased. This is due to the fact that the MFCC feature vectors are the result of a DCT which has the effect of ordering the coefficients in decreasing order of importance. Thus, the higher numbered MFCC coefficients contain less discriminatory information than the lower numbered coefficients. Examining the results of the vowel classification tasks, Figures 5.8(a)–(b), one can also observe that the overall performance of the six classifiers does not differ considerably. However, viewing the phone class classification results, Figure 5.8(c), one can see that the differences in performance between the different classifiers is more pronounced. In particular the GMM classifier performs poorly on the phone class classification task. This suggests that the GMM is unable to model the probability density of the phone class data, this may be due to high level of variation in this data set.

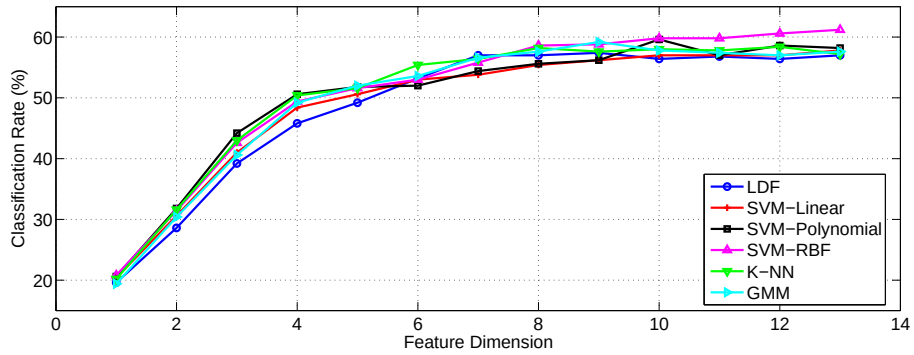
In order to assess the overall performance of the classifiers the mean classification rates, for all feature dimensionalities, were computed and are presented in Table 5.2. The SVM classifier with RBF kernel (5.20) was found to outperform all other classifiers in terms of mean classification rate in all three phone classification tasks. As a result, the SVM classifier with RBF kernel was used in all phone classification experiments conducted in this dissertation. However, one can observe that a range of different classifiers were found to produce similar classification performance and as such the choice of this particular classifier is relatively insignificant.

Speaker identification

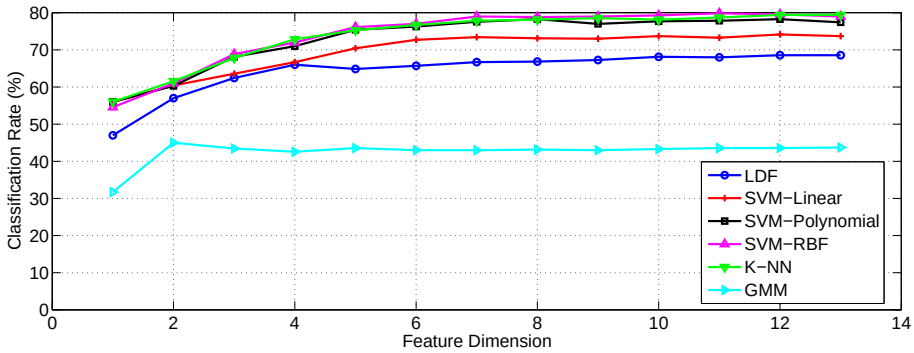
In addition to examining the ability of various dimensionality reduction methods to retain information useful for phone classification performance, we propose evaluating the ability of these methods to produce low-dimensional features that preserve information capable



(a) Five vowel classification task



(b) Ten vowel classification task



(c) Phone class classification task

Figure 5.8: Classification rate vs. feature dimensionality for each type of classifier.

of discriminating between different speakers. This can be accomplished by performing a number of speaker identification experiments.

Speaker identification is the task of automatically identifying which one of a set of known speakers produced a test speech utterance. A speaker identification system may be text-dependent (constrained phrase) or text-independent (any phrase). This may be viewed as a pattern classification problem and divided into two distinct steps: feature extraction and classification. To achieve a high degree of accuracy in this task it is important to choose a feature space in which individual speakers are well separated and easily distinguished from one another. A wide range of features have been used in speaker identification in the past, with MFCCs being the most popular and widely used feature. MFCCs are not specifically designed to parametrise speaker discriminative information, although they do implicitly contain information relating to a speakers physiology, including vocal tract and some glottal source information.

Following feature extraction it is desirable to transform the features into a relatively low-dimensional feature space while preserving information relevant to the speaker identification task. This allows simple measures of similarity to be applied to compare points within the transformed feature space. This dimensionality reduction also reduces the computational complexity and storage requirements involved in classifier training and testing. A number of linear dimensionality reduction methods have been successfully applied to speaker recognition in the past including LDA (Jin and Waibel, 2000), independent component analysis (Jang et al., 2001) and PCA (de Lima et al., 2002). These methods are limited to performing linear transformations of the data.

Due to unique physiologies and speaking styles different speakers produce sounds occupying distinct, possibly overlapping, submanifolds in acoustic space. In this dissertation we apply PCA and the manifold learning algorithm L-Isomap to speech data from a number of different speakers in an effort to produce low-dimensional features containing information relevant to speaker identity while discarding redundant information. The motivations for using L-Isomap as the sole nonlinear method are discussed in Section 7.6. The performance of the resulting features is compared with conventional MFCCs in a text-independent speaker identification task.

As we are concerned with examining the speaker discriminatory properties of a number

of different feature types we used a basic speaker identification system. The experimental setup we have used is based on that described by Reynolds (1995a). It lacks several features of a state-of-the-art speaker identification system, such as a universal background model (Reynolds, 2002), however it has been shown to be capable of accurate speaker identification. Further details of the experimental setup are described in Section 7.6.

5.6 Summary

The procedure described in the preceding sections provides a framework for applying dimensionality reduction methods to chosen sets of speech signals and evaluating the performance of each method. This framework allows the computation of two types of high-dimensional speech feature space, the magnitude spectrum derived MFCC space and the phase spectrum derived MODGDF space. Following the application of the dimensionality reduction methods to either of these speech feature spaces two types of evaluation may be performed: visualisation and classification. In the Chapters 6 and 7 this framework is applied to both synthetic and natural speech corpora in a number of different experiments.

Chapter 6

Experiments on Synthetic Speech Data

This chapter describes a number of experiments in which the framework described in Chapter 5 is applied to synthetic speech signals. Firstly, the motivations for using synthetic speech are put forward. Following this, the method used to generate synthetic speech is described. Visualisation and phone classification experiments and their results are then discussed. Finally, the conclusions drawn from these experiments are proffered.

6.1 Introduction

In Chapter 4 the dimensionality reduction methods PCA, Isomap, LLE, and Laplacian eigenmaps are applied to a number of non-speech data sets and their ability to produce meaningful low-dimensional representations is demonstrated. In order to evaluate the usefulness of these methods in speech processing applications we first created a number of synthetic speech signals to be analysed using our proposed framework. Using synthetic speech data facilitates the analysis of signals with known and controllable characteristics. Thus, after applying dimensionality reduction methods to these synthetic sounds we can examine the resulting lower-dimensional data set to determine the degree to which these characteristics have been retained.

LF Parameter	Male
T	10.0 ms
t_p	4.1 ms
t_e	5.5 ms
t_a	0.1 ms
t_c	5.8 ms
E_e	-1

Table 6.1: LF parameter values used for the glottal flow derivative waveform used as excitation for synthetic speech.

6.2 Synthetic speech generation

The speech synthesis technique used in this dissertation is similar to that described by McKenna (2004), who adapted a technique first proposed by d'Alessandro et al. (1998). Synthetic speech was generated by exciting an LP-modelled filter with an artificially generated excitation signal. The excitation signal was generated using an LF-modelled glottal pulse train (Fant et al., 1985). The LF model for the glottal flow derivative waveform and its parameters[1] are shown in Figure 6.1. The start of the open phase, and end of the previous closed phase, is t_0. The parameter t_p is the instant of maximum airflow. The start time of the return phase is indicated as t_e; the parameter E_e accounts for the amplitude of the main glottal excitation. The end time of the return phase, full closure of the glottis, is indicated as t_c. The parameter t_a measures the abruptness of glottal closure; it is the time from the starting point of the return phase, t_e, to the point where a tangent to the exponential at $t = t_e$ hits the zero axis. The length of a single pitch period is T, allowing the glottis to remain closed for a period after the glottal pulse. The LF parameter values used in the synthetic speech generated for this study correspond to those suggested for modal speech by Childers (1999). These parameter values are stated in Table 6.1.

This excitation signal was then applied to a set of 10 linear prediction coefficients representing five formants to produce a synthetic speech signal. The formant values used varied depending on the required experiment, as detailed in the following sections.

[1]The LF-model, as originally described by Fant et al. (1985), requires only four parameters to uniquely describe the shape of the glottal flow: t_p, t_e, t_a, and E_e. The additional parameters—T, t_c, and t_0—describe the timing of the glottal flow components, and are necessary for synthesis. The reader is referred to Fant et al. (1985) and Gobl (2003) for further details of the voice source and LF model.

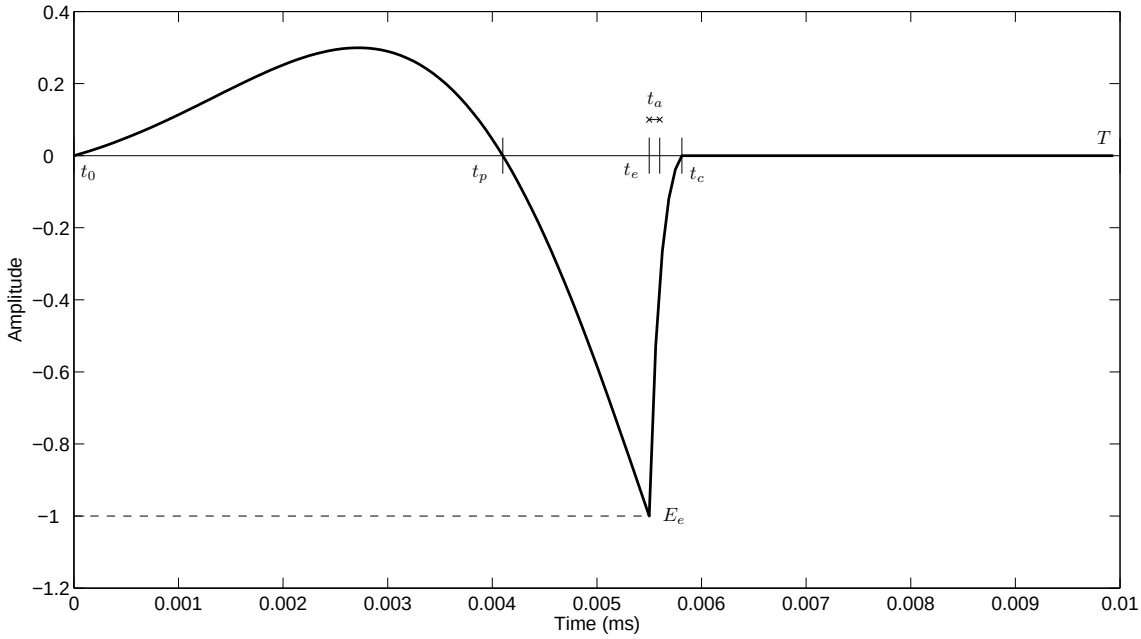


Figure 6.1: LF model for the glottal flow derivative waveform.

6.3 Visualisation

6.3.1 Introduction

The initial experiment conducted on synthetic speech aimed to visually examine the ability of each dimensionality reduction method to discover low-dimensional variation known to be present in a speech signal. A number of synthetic speech signals were generated in which important components of the speech signal were varied from the signal start to end. The purpose of introducing this variation is to determine if the dimensionality reduction methods can retain these important sources of variation in a lower-dimensional embedding of the signal while discarding less relevant information. The three components varied included: the first and second formants (F1 and F2) and fundamental frequency (f0). These are three of the primary sources of information within a speech signal.

6.3.2 Data

Four types of synthetic speech signals were generated, they are described as follows:

- Varying F1: Initial F1 frequency of 300 Hz, increasing in equal sized increments reaching 700 Hz at the signal end. The other four formants were kept constant from beginning to end. This resulted in a synthetic speech signal moving, approximately,

from an /u/ to an /ɑ/ sound in vowel space (see Figure 3.2). The fundamental frequency was set at 100 Hz ($T = 10.0\,\text{ms}$) for the duration, as in Table 6.1.

- Varying F2: Initial F2 frequency of 1000 Hz, increasing in equal sized increments reaching 2200 Hz at the signal end. The other four formants were kept constant from beginning to end. This resulted in a synthetic speech signal moving, approximately, from an /u/ to an /i/ in vowel space (see Figure 3.2). The fundamental frequency was set at 100 Hz ($T = 10.0\,\text{ms}$) for the duration, as in Table 6.1.

- Varying F1 and F2: Initial F1 frequency of 300 Hz, increasing in equal sized increments reaching 700 Hz at the signal end. Initial F2 frequency of 1000 Hz, increasing in equal sized increments reaching 2200 Hz at the signal end. The other three formants were kept constant from beginning to end. This resulted in a synthetic speech signal moving, approximately, from an /u/ to an /æ/ in vowel space (see Figure 3.2). As above, f0 was maintained at 100 Hz.

- Varying f0: The fundamental frequency was increased, in equal increments, from 80 Hz ($T = 12.5\,\text{ms}$) at the signal start to 250 Hz ($T = 4.0\,\text{ms}$) at the signal end, all other LF model parameters were as described in Table 6.1. Formant values were set equal to the start state of the above three signals.

Each synthetic speech signal was generated with a sampling frequency of 16 kHz and was 2 s in duration. The start and end values of the formant trajectories for the synthetic speech signals described above are summarised in Table 6.2. The equivalent formant bandwidths used for all synthesised speech are given in Table 6.3. Spectrograms of the three speech signals in which the formant values are varied are shown in Figures 6.2(a), 6.3(a), and 6.4(a); formant trajectories are overlaid on each. Also, displayed in Figure 6.5(a) is the time domain representation of the first and last 40 ms of the speech signal in which f0 is varied.

6.3.3 Experiments

Each of the four artificially generated speech signals described above were analysed using the framework detailed in Chapter 5. The speech signals were first preemphasised, with $p = 0.98$, and following this 13-dimensional MFCC feature vectors were computed from

Formant	F1 varied		F2 varied		F1 & F2 varied	
	Start	End	Start	End	Start	End
F1	**300 Hz**	**700 Hz**	300 Hz	300 Hz	**300 Hz**	**700 Hz**
F2	1000 Hz	1000 Hz	**1000 Hz**	**2200 Hz**	**1000 Hz**	**2200 Hz**
F3	2400 Hz	2400 Hz	2400 Hz	2400 Hz	2400 Hz	2400 Hz
F4	3500 Hz	3500 Hz	3500 Hz	3500 Hz	3500 Hz	3500 Hz
F5	5000 Hz	5000 Hz	5000 Hz	5000 Hz	5000 Hz	5000 Hz

Table 6.2: Start and end values of the formant frequency trajectories used to generate synthetic speech sounds. Bold values indicate the formants that vary.

Formant	Bandwidth
F1	60 Hz
F2	90 Hz
F3	150 Hz
F4	200 Hz
F5	300 Hz

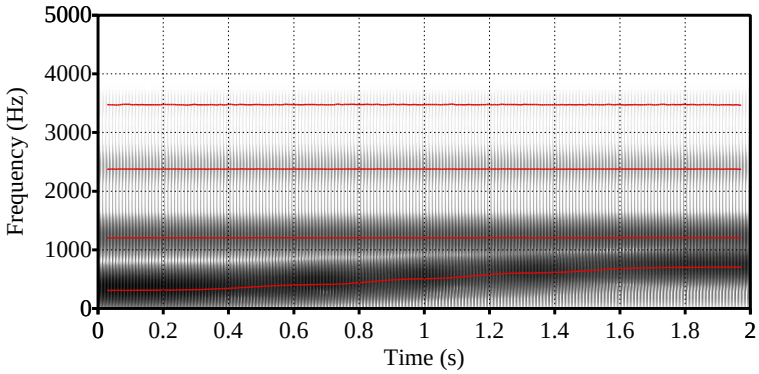
Table 6.3: Bandwidths for each formant used to generate synthetic speech sounds.

20 ms frames extracted with an overlap of 10 ms. This resulted in a data set of $N = 199$ MFCC feature vectors for each speech signal. Each of these data sets were then separately provided as input to the dimensionality reduction algorithms Isomap, LLE, Laplacian eigenmaps, and PCA. The two- and three-dimensional embeddings output by these methods were then visually inspected to determine if any low-dimensional structure had been retained.

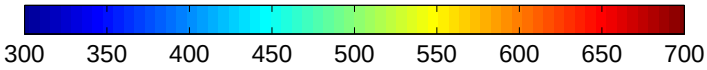
6.3.4 Results

The visualisation results corresponding to the four synthetic speech signals are shown in Figures 6.2–6.5. One can observe from Figures 6.2–6.4 that all four dimensionality reduction techniques successfully discovered the formant variation present in the original speech signals in the two-dimensional embedding spaces they produced. Notably there is no clear difference in the clarity or extent to which the different techniques preserved the formant variation in the low-dimensional space.

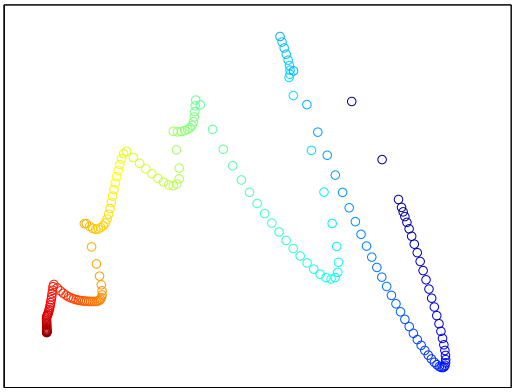
It is also apparent, given the low-dimensional embeddings of the speech signal with a varying f0 trajectory shown in Figure 6.5, that all four dimensionality reduction methods retained information relating to f0 variation in a low-dimensional space. However, to provide clear visualisations of the variation in f0, three-dimensional embeddings—rather than two-dimensional as in the case of formant variation—are required. It can also be seen that



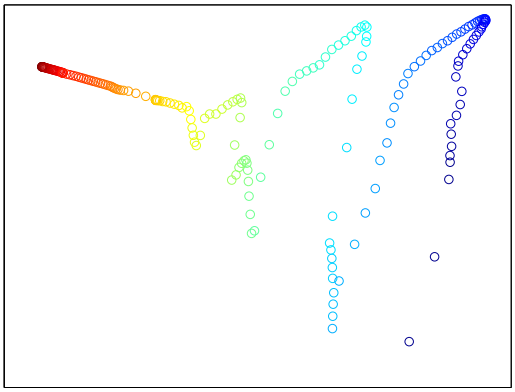
(a) Spectrogram of a synthetic vowel sound; F1 increases from 300–700 Hz. Formant tracks are shown as solid red lines.



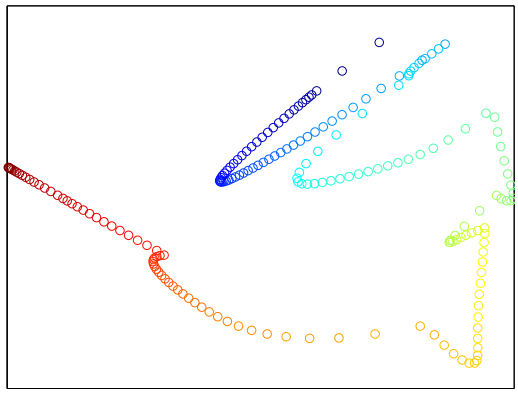
(b) Colour bar: F1 values (Hz) corresponding to the plots below.



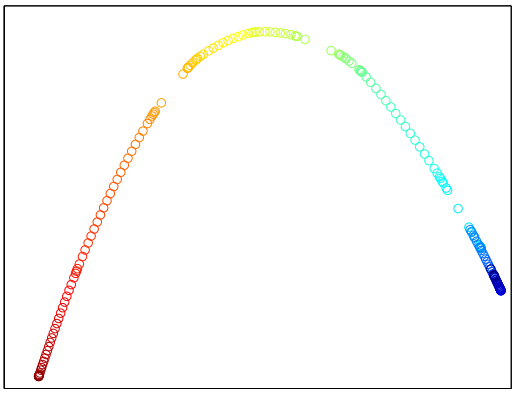
(c) PCA



(d) Isomap

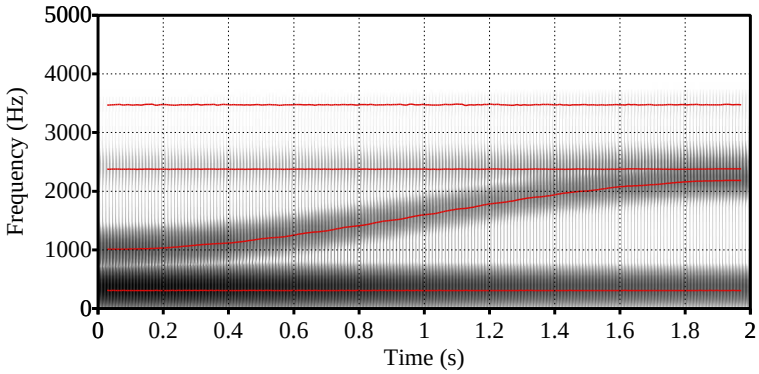


(e) LLE

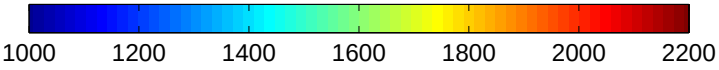


(f) Laplacian eigenmaps

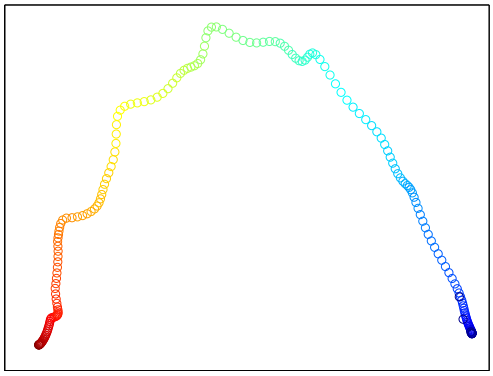
Figure 6.2: Two-dimensional embeddings produced by applying dimensionality reduction methods to synthetic speech with varying F1.



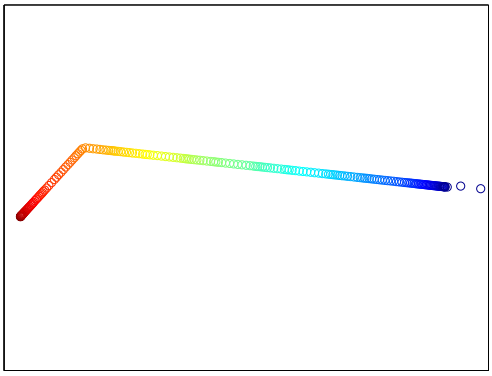
(a) Spectrogram of a synthetic vowel sound; F2 increases from 1000–2200 Hz. Formant tracks are shown as solid red lines.



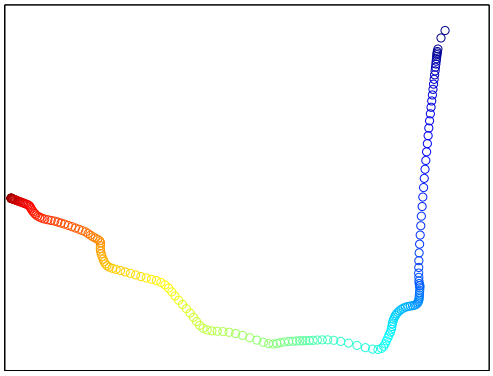
(b) Colour bar: F2 values (Hz) corresponding to the plots below.



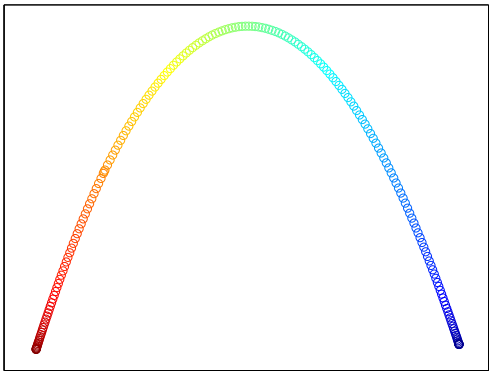
(c) PCA



(d) Isomap

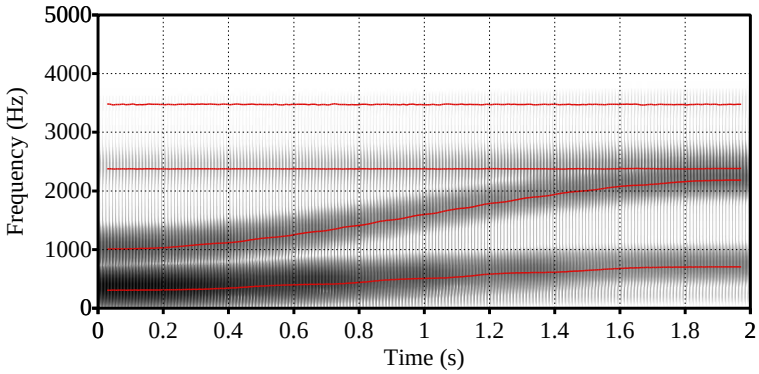


(e) LLE

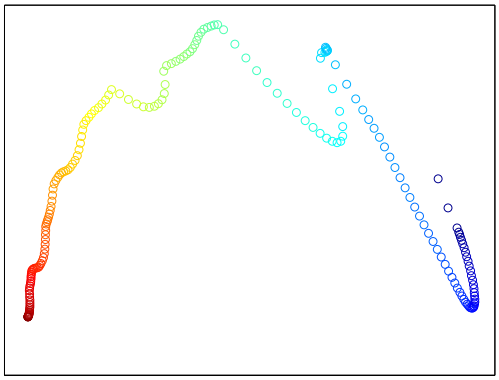


(f) Laplacian eigenmaps

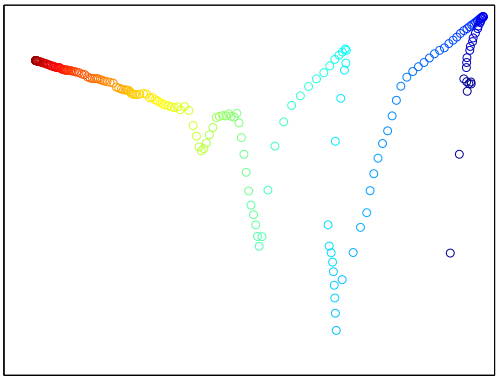
Figure 6.3: Two-dimensional embeddings produced by applying dimensionality reduction methods to synthetic speech with varying F2.



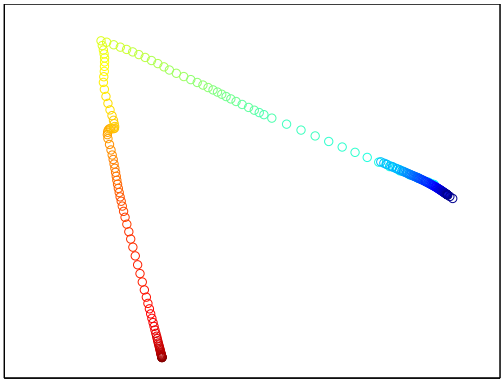
(a) Spectrogram of a synthetic vowel sound; F1 increases from 300–700 Hz, F2 increases from 1000–2200 Hz. Formant tracks are shown as solid red lines.



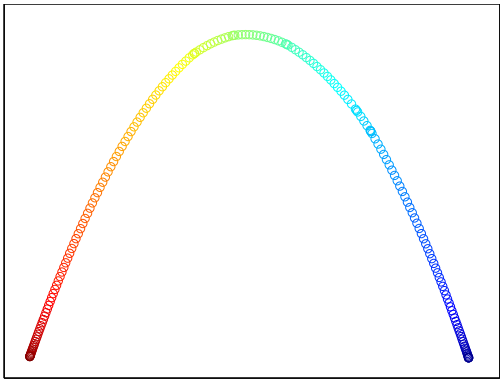
(b) PCA



(c) Isomap



(d) LLE



(e) Laplacian eigenmaps

Figure 6.4: Two dimensional embeddings produced by applying dimensionality reduction methods to synthetic speech with varying F1 and F2. For associated colour bars see Figures 6.2(b) and 6.3(b).

the variation in pitch is less well defined and separated than the formant variation shown in the previous figures. This indicates that more dimensions are required to accurately retain information relating to pitch variation than formant variation.

6.4 Classification of synthetic vowels

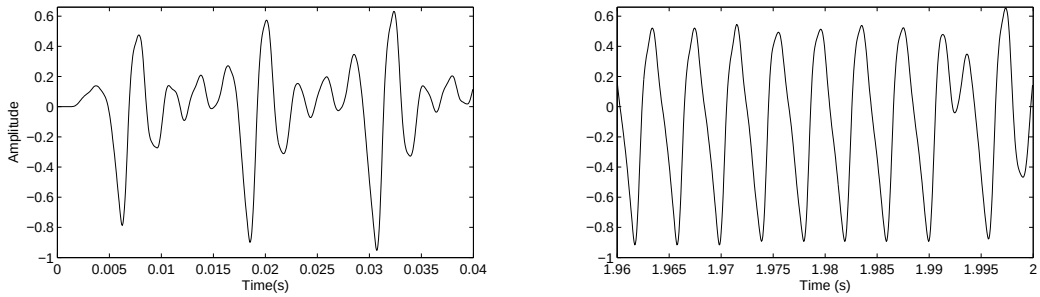
6.4.1 Introduction

The previous section provides a subjective, visualisation-based, assessment of the different dimensionality reduction methods' ability to produce meaningful low-dimensional representations of speech data. In this section we examine the ability of the dimensionality reduction methods to produce low-dimensional representations of speech data suitable for phone classification tasks. Again, the motivation to work with synthetic speech data is to facilitate control of the degrees of variation in the speech data.

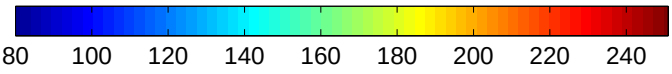
6.4.2 Data

The synthetic speech generation procedure described above in Section 6.2 allows the synthesis of arbitrary vowel sounds. In this, second, experiment we performed the phone classification tasks 1 and 2 described in our proposed framework, Section 5.5.2, using synthetic vowels. Thus, we required the ability to generate ten different vowel sounds. The vowels used, and frequencies of the first three formants for each vowel, based on those presented by Peterson and Barney (1952), are listed in Table 6.4. The fourth and fifth formants were kept fixed, as in Table 6.2, and the formant bandwidths used for all synthesised vowels are shown in Table 6.3.

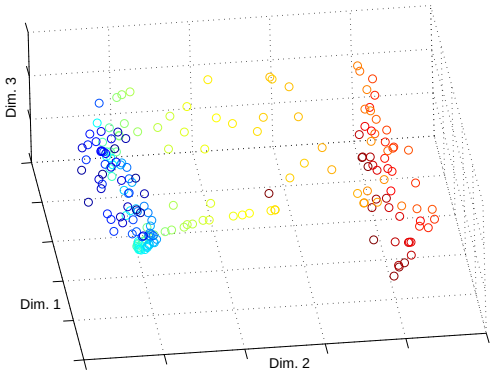
The classification of such well defined, noise free, spectrally consistent synthetic vowels would be a relatively simple task and would not reflect the difficulties associated with the classification of natural vowel sounds. In natural speech the formant values associated with vowel sounds, as listed in Table 6.4, are simply targets which the speaker attempts to reach but may in reality undershoot or overshoot due to factors such as coarticulation. As a result, we generated a set of synthetic vowel sounds in which the formant values where not fixed but varied slightly from one utterance to the next. This was accomplished by sampling the formant values from Gaussian distributions with means as indicated in



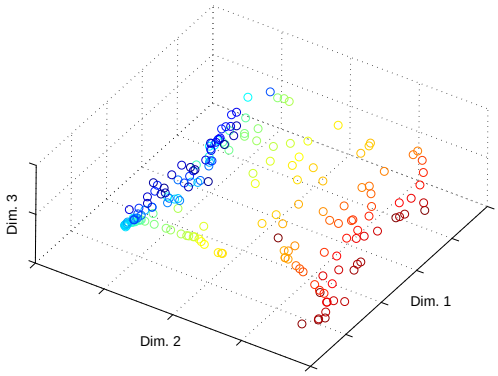
(a) The first 40 ms, above left, and last 40 ms, above right, from a synthetic /u/ vowel sound of total duration 2 s. The f0 value of the synthetic vowel increases from 80 Hz at the beginning of the vowel to 250 Hz at the end.



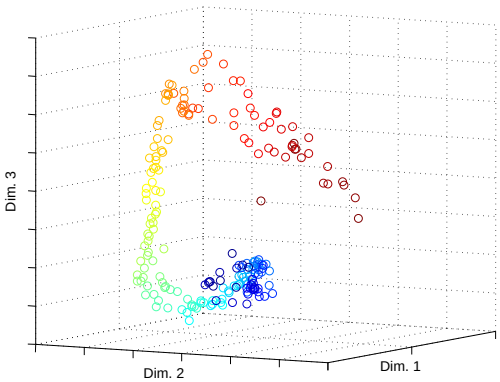
(b) Colour bar: f0 values (Hz) corresponding to the plots below.



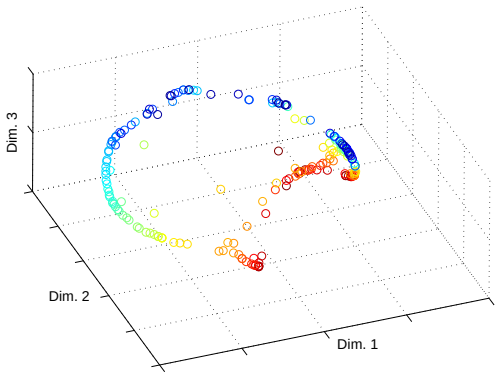
(c) PCA



(d) Isomap



(e) LLE



(f) Laplacian eigenmaps

Figure 6.5: Three-dimensional embeddings produced by applying dimensionality reduction methods to synthetic speech with varying f0.

Vowel	F1	F2	F3
/ɑ/	730 Hz	1090 Hz	2440 Hz
/i/	270 Hz	2290 Hz	3010 Hz
/u/	300 Hz	870 Hz	2240 Hz
/ɛ/	530 Hz	1840 Hz	2480 Hz
/ʌ/	640 Hz	1190 Hz	2390 Hz
/ɪ/	390 Hz	1990 Hz	2550 Hz
/ə/	500 Hz	1000 Hz	2100 Hz
/o/	450 Hz	1090 Hz	2300 Hz
/ɔ/	570 Hz	840 Hz	2410 Hz
/æ/	660 Hz	1720 Hz	2410 Hz

Table 6.4: Formant frequencies used for vowel synthesis.

Table 6.4 and with a standard deviation of 50 Hz for F1, 100 Hz for F2, and 250 Hz for F3. These deviations were chosen to add variation to each formant. The effect of this formant variation was evaluated in informal listening tests. All of the synthetic vowels were perceived as being very close to real speech. Also, in the vast majority of cases the synthetic vowel sounds were perceived as the intended vowel, despite the addition of formant variation. However, a small number of the synthetic utterances were perceived as falling somewhere between two vowel sounds, making them difficult to categorise as one particular phone. These sounds were those generated with formant values in the tails of the distributions discussed above, e.g. close to 50 Hz of variation in F1.

Furthermore, in addition to the incorporation of formant variation we introduced a degree of noise corruption to the synthetic vowels in order to better approximate natural speech conditions. This involved adding a Gaussian white noise component, centred on the instant of glottal closure of each pitch period, as performed by McKenna (2004) and d'Alessandro et al. (1998) . Three levels of noise component duration were used: 0% (noise free), 60%, and 100% of the pitch period. The intensity of the noise used was varied in four levels, with signal-to-noise ratios (SNR) of: ∞ dB (no added noise), 20 dB, 10 db and 5 dB. Given the various combinations of noise duration and intensity, two separate sets of noise combinations were used: a low noise set, whose constituent noise duration and intensity combinations are shown in Table 6.5; and a high noise set, detailed in Table 6.6. Separate experiments were performed on low and high noise data sets. For each synthetic vowel generated the type of noise corruption applied was selected at random from the possible combinations of either the low or high noise set.

Duration	SNR
0%	∞ dB
60%	20 dB
60%	10 dB
100%	20 dB
100%	10 dB

Table 6.5: Low noise set: possible combinations of duration and SNR of the noise components added to each pitch period of the synthetic vowel sounds.

Duration	SNR
60%	20 dB
60%	10 dB
60%	05 dB
100%	20 dB
100%	10 dB
100%	05 dB

Table 6.6: High noise set: possible combinations of duration and SNR of the noise components added to each pitch period of the synthetic vowel sounds.

6.4.3 Experiments

As mentioned above two phone classification tasks were performed on synthetic data:

Task 1 Five vowels: /ɑ/, /i/, /u/, /ɛ/, and /æ/.

Task 2 Ten vowels: /ɑ/, /i/, /u/, /ɛ/, /æ/, /ʌ/, /ə/, /ɔ/, /ɪ/, and /o/.

For each task 250 utterances of each vowel were synthesised, with formant and noise variation, as described above. Each synthesised utterance was 40 ms in length. Following the synthesis of all required vowel utterances, the feature extraction procedure described in Sections 5.2 and 5.3 was used to produce 13-dimensional MFCC feature vectors from the preemphasised, $p = 0.98$, and windowed 40 ms utterances.

The resulting feature vectors were concatenated forming an $N \times D$ matrix, where $D = 13$ and $N = 1250$ in the five vowel task and $N = 2500$ in the ten vowel task. PCA, Isomap, LLE, and Laplacian eigenmaps were individually applied to reduce the dimensionality of this MFCC data matrix. The target dimensionality of these algorithms was varied, producing transformed feature vectors of dimensionality $d = 1, \ldots, D$. As previously indicated in Section 5.4.2, the number of nearest neighbours, k, used in the manifold learning methods was chosen empirically. The value of k was varied in the range $k = 2, \ldots, 30$ and the value which produced the highest classification accuracy was chosen

for each of the manifold learning methods.

Phone classification experiments were performed using five different feature types: the original MFCC vectors and the features produced by applying PCA, Isomap, LLE, and Laplacian eigenmaps to the baseline MFCC vectors. For all feature types, a separate classification experiment, that is training and testing, was performed using feature vectors of dimensionality $d = 1, \ldots, D$. Thus, the ability of these feature transformation methods to produce useful low-dimensional features could be evaluated and changes in performance with varying dimension analysed. The original MFCC vectors served as a baseline, also varying in dimensionality as detailed above. As discussed in Section 5.5.2, SVM classifiers with RBF kernels were used in all classification experiments.

For all runs of the phone classification experiments, irrespective of feature type or dimensionality, the following steps were completed:

1. The feature vectors were partitioned into a training set, containing 200 labelled utterances per phone, and a test set, comprising the remaining 50 unlabelled utterances per phone.

2. An SVM classifier was trained on the labelled training feature vectors

3. The trained classifier was tested, using a one-against-one scheme followed by majority voting as discussed in Section 5.5.2, on the unlabelled test feature vectors.

4. The percentage of correctly classified feature vectors, the 'classification rate', was computed using the known vowel classes of the test feature vectors.

6.4.4 Results

The results of the low and high noise synthetic vowel classification experiments are exhibited in Figures 6.6 and 6.7, respectively. The percentage of phones correctly classified is given on the vertical axis. The horizontal axis represents feature vector dimensionality. Firstly, it should be noted that the comparative difficulty of each classification task, based on the maximum classification rate achieved by all feature types, was as expected. Higher classification rates were found in the five vowel classification task than in the ten vowel task, due to the increased possibility of phone confusion in the latter task. Also, classification rates in the high noise tasks were lower than in the low noise task. These results

Classification Task	Noise set	Max. Classification Rate
Five vowel	Low	94.4%
Ten vowel	Low	80.4%
Five vowel	High	70.4%
Ten vowel	High	40.4%

Table 6.7: Maximum classification rate achieved in each synthetic vowel classification task.

	/ɑ/	/i/	/u/	/ɛ/	/æ/
/ɑ/	49	0	0	0	1
/i/	0	50	0	0	0
/u/	0	0	50	0	0
/ɛ/	0	0	0	44	6
/æ/	2	0	0	6	42

Table 6.8: Confusion matrix: Five vowel classification using 13-dimensional MFCCs (low noise).

are summarised in Table 6.7.

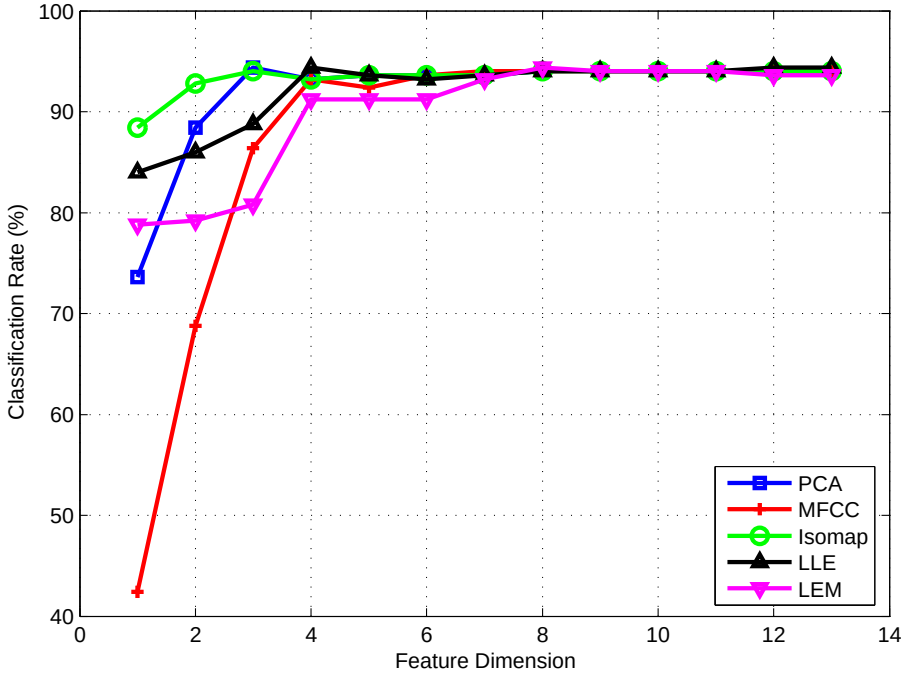
Confusion matrices for the low and high noise synthetic vowel classification experiments are also provided in Tables 6.8–6.11. These tables display confusion matrices based on classification using the original 13-dimensional MFCC feature vectors. It can be seen that the most frequent confusions occurred between phonetically similar vowels, for example /ɑ/ confused with /æ/, /æ/ with /ɛ/, /ɑ/ with /ʌ/, and /o/ with /u/.

Comparing the performance of the various feature types, the following points can be observed from the results of Figures 6.6 and 6.7:

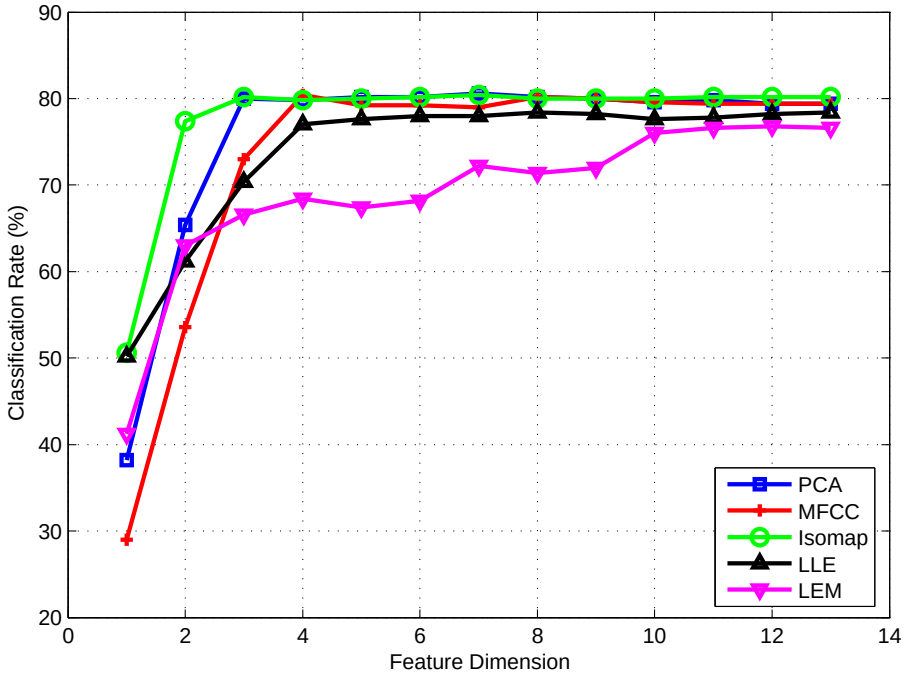
- In all four classification experiments, as the dimensionality of the feature vectors is increased from $d = 1, \ldots, 4$ the classification rate increases sharply. However, from $d = 4, \ldots, 13$ this increase in classification rate halts. This indicates that the fifth and higher dimensions do not add significant discriminatory information, suggesting that the inherent dimensionality, as discussed in Section 3.2.2, of the synthetic

	/ɑ/	/i/	/u/	/ɛ/	/æ/
/ɑ/	38	0	4	1	7
/i/	0	38	10	2	0
/u/	1	2	47	0	0
/ɛ/	4	9	3	20	14
/æ/	13	7	1	12	17

Table 6.9: Confusion matrix: Five vowel classification using 13-dimensional MFCCs (high noise).

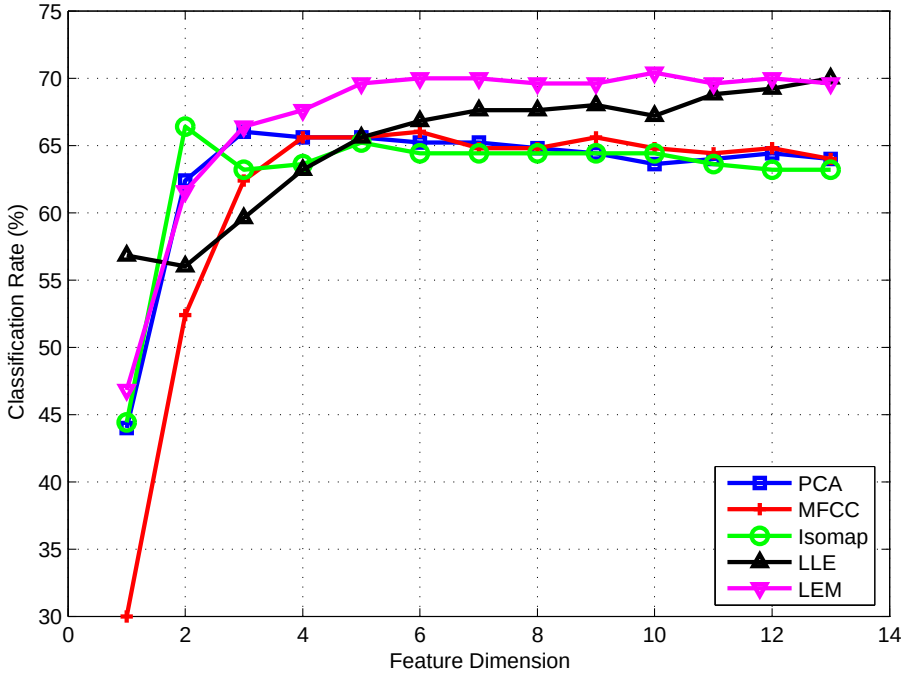


(a) Five vowel classification task

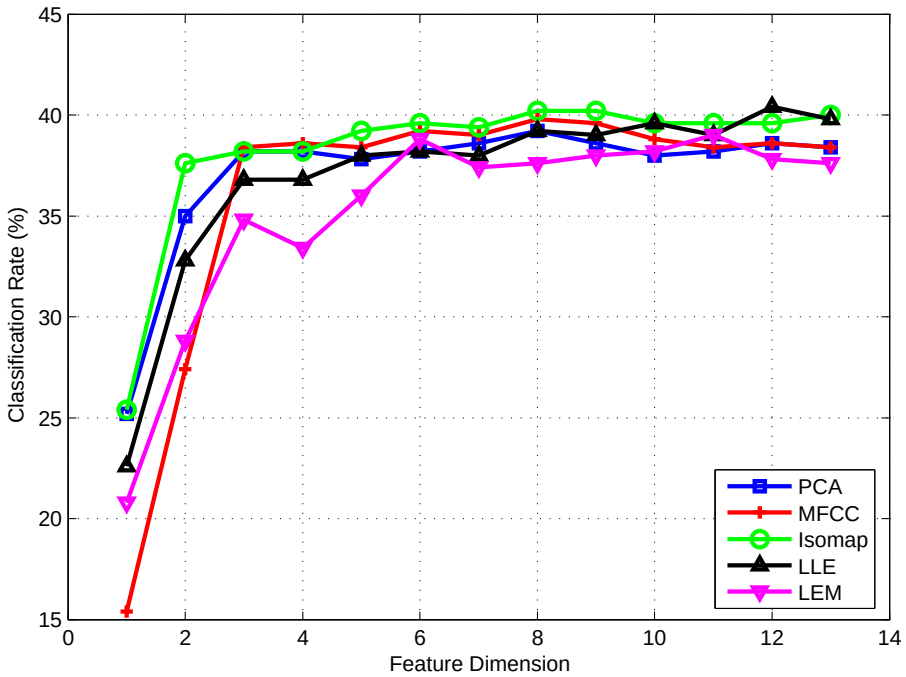


(b) Ten vowel classification task

Figure 6.6: Results of low noise synthetic vowel classification experiments.



(a) Five vowel classification task



(b) Ten vowel classification task

Figure 6.7: Results of high noise synthetic vowel classification experiments.

	/ɑ/	/i/	/u/	/ɛ/	/æ/	/ʌ/	/ə/	/ɔ/	/ɪ/	/o/
/ɑ/	34	0	0	0	0	14	1	1	0	0
/i/	0	47	0	0	0	0	0	0	3	0
/u/	0	0	48	0	0	0	0	0	0	2
/ɛ/	0	0	0	42	5	1	0	0	2	0
/æ/	1	0	0	6	41	2	0	0	0	0
/ʌ/	12	0	0	0	1	30	3	1	0	3
/ə/	0	0	0	0	0	4	27	3	0	16
/ɔ/	0	0	0	0	0	1	2	45	0	2
/ɪ/	0	1	0	3	0	0	0	0	46	0
/o/	0	0	0	0	0	1	9	3	0	37

Table 6.10: Confusion matrix: Ten vowel classification using 13-dimensional MFCCs (low noise).

	/ɑ/	/i/	/u/	/ɛ/	/æ/	/ʌ/	/ə/	/ɔ/	/ɪ/	/o/
/ɑ/	16	0	2	0	3	6	6	12	0	5
/i/	0	36	9	1	0	0	0	0	3	1
/u/	0	1	44	0	0	0	1	3	1	0
/ɛ/	1	6	1	15	11	4	2	1	5	4
/æ/	9	4	1	9	13	2	3	2	3	4
/ʌ/	11	0	4	5	1	9	8	8	2	2
/ə/	3	0	16	0	0	3	15	8	0	5
/ɔ/	6	0	7	0	0	2	8	25	0	2
/ɪ/	0	16	8	9	0	1	1	0	9	6
/o/	1	0	18	0	0	3	9	6	3	10

Table 6.11: Confusion matrix: Ten vowel classification using 13-dimensional MFCCs (high noise).

speech data analysed may be as low as four. As a comparison, the inherent dimensionality of a teapot rotating about a single axis is one, as described in Section 4.1. Thus, an estimate of four as the inherent dimensionality of the synthetic vowel data analysed indicates that the underlying system generating the synthetic speech could be described using just four parameters. This is consistent with previous studies, a review of which is provided in Chapter 3.

- Laplacian eigenmaps is something of an exception to the above statement. In Figure 6.6(b), the performance of Laplacian eigenmaps can be seen to plateau from dimension 4–6 and again from 7–9. This indicates that a significant amount of discriminatory information is contained in both dimensions 7 and 10. The same is true of dimensions 5 and 6 in Figure 6.7(b). This indicates that Laplacian eigenmaps has not successfully compacted the most discriminatory information into the lowest dimensions, unlike all of the other methods.

- In the majority of cases the dimensionality reduction methods outperformed the baseline MFCCs in low dimensions, $d \leq 3$. We propose that this is evidence of the dimensionality reduction methods' ability to reduce the dimensionality of speech data, discarding unimportant information and retaining meaningful information in low-dimensional feature spaces.

- Isomap embeddings offered the best classification rate in the majority, 51.92%, of tests. This figure takes into account all of the individual synthetic classification tests run. There were 52 ($13 \times 2 \times 2$) tests run; resulting from all 13 dimensionalities being tested in both five and ten vowel tests on both high and low noise data. Table 6.12 gives details of the percentage of tests in which each feature type yielded the greatest performance.

- The performance of the Laplacian eigenmaps algorithm was the most inconsistent of all the dimensionality reduction methods. The features output by this algorithm frequently performed worse than the other features, including the baseline MFCCs; however, Laplacian eigenmaps clearly yielded the best overall performance in the high noise five vowel classification task.

- Interestingly LLE also performed well in the high noise five vowel classification task

with an average classification rate second only to Laplacian eigenmaps. In fact, the two local methods, LLE and Laplacian eigenmaps, offered similar performance in all four tests.

- While the locality preserving manifold learning methods, Laplacian eigenmaps and LLE, did not consistently outperform PCA or MFCC, the globally motivated Isomap did. This suggests that global methods may be more capable of discovering meaningful, phone discriminatory information, than local manifold learning or linear dimensionality reduction methods. This may be because globally motivated methods attempt to preserve the global geometric structure of the underlying manifold, as discussed in Section 4.3.2. Preserving this structure ensures that points which are far apart on the manifold in high-dimensional space will also be far apart in low-dimensional space, and likewise for points which are close together. This is important for classification tasks as it helps to ensure that the distances between points in low-dimensional space are a true reflection of the distances between the points on the data manifold. This helps prevent points which are located a large distance apart on the manifold being located close together in low dimensional space and incorrectly classified as being close neighbours.

- Interestingly, in Figure 6.7(b) the two-dimensional embedding space produced by Isomap provides better classification accuracy than higher-dimensional embeddings produced by Isomap. This indicates that most of the discriminatory information is compacted into two dimensions and that further dimensions are effectively noise that reduces classification accuracy.

- The manifold learning methods performed particularly well in the high noise classification experiments, offering the best performance overall. This indicates that they are more capable of extracting discriminatory information than the other methods in the case of high levels of noise corruptions. This may be due to the manifold learning methods exploiting nonlinear structure in the data to more efficiently compress discriminatory information in the low-dimensional representations output by the methods.

Feature Type	Yielded Best Performance
Isomap	51.92%
Laplacian eigenmaps	21.15%
LLE	11.54%
PCA	7.69%
MFCC	7.69%

Table 6.12: Percentage of synthetic vowel classification tests in which each feature type yielded the maximum performance.

6.5 Conclusions

As demonstrated in the visualisation experiments, dimensionality reduction methods are clearly capable of discovering meaningful low-dimensional representations of synthetic speech data. Results of synthetic phone classification show that the dimensionality reduction methods offer improved performance, over baseline MFCCs, in very low dimensions ($d \leq 3$). For higher dimensions the dimensionality reduction methods were, in general, not found to offer a great improvement over the baseline MFCCs in the case of low noise; however, in the case of high noise manifold learning methods were found to yield higher classification rates than MFCCs and PCA-transformed features.

Chapter 7

Experiments on Natural Speech Data

The previous chapter describes the application of our proposed framework—including the PCA, Isomap, LLE, and Laplacian eigenmaps algorithms—to a range of synthetic speech signals. However, due to the small scale of the synthetic data sets used and idealised nature of the constituent synthetic speech signals it is difficult to conclusively evaluate the performance of the various dimensionality reduction algorithms based on these studies. Therefore, in this chapter we describe a number of studies applying our proposed framework, detailed in Chapter 5, to speech signals taken from a corpus of natural speech recordings.

7.1 The TIMIT speech corpus

The speech data used in the studies discussed in this chapter was taken from the TIMIT corpus (Garofalo et al., 1990). The speech in the corpus was recorded at Texas Instruments (TI), transcribed at the Massachusetts Institute of Technology (MIT), and produced and distributed by the National Institute of Standards and Technology. The TIMIT corpus has seen widespread use in numerous speech applications since its release.

TIMIT contains 6300 utterances, 10 spoken by each of 630 American English speakers. The speech recordings are provided at a sampling frequency of 16 kHz. Accompanying each utterance is a time-aligned phonetic transcription that labels the start and end time of every phone in the utterance.

7.2 Visualisation of f0 variation

7.2.1 Introduction

In Section 6.3, four dimensionality reduction methods were applied to synthetic speech to produce three-dimensional spaces in which f0 variation could be visualised. Given the success of these results we conducted additional experiments aimed at visualising f0 variation in natural speech recordings. The methodology used and results of these experiments are reported in the following sections.

7.2.2 Experiment

For these experiments all units of the following vowels were extracted from the TIMIT corpus: /ɑ/, /i/, /ɛ/, /æ/, and /o/. For each phone unit, frames of length 40 ms were extracted from the centre of each unit. Units of duration less than 100 ms were discarded. The raw speech frames were preemphasised, $p = 0.98$, and Hamming windowed. Following this preprocessing, 13-dimensional MFCC vectors were computed for each frame.

For each phone, all associated feature vectors were assembled into a high-dimensional data set. The data set from each phone was then separately provided as input to the dimensionality reduction algorithms Isomap, LLE, Laplacian eigenmaps, and PCA. A k value of 5 was used for all the manifold learning methods.

As each data set consisted of features extracted from a single phone uttered by many speakers, it was assumed that f0 was likely to vary considerably in each data set. In order to examine this, the f0 value of each of the original units was estimated. The f0 estimation procedure described by Sun (2002) was used to accomplish this. The two- and three-dimensional embeddings output by these methods were then visually inspected to determine if any low-dimensional structure, relating to f0, had been retained.

7.2.3 Results and discussion

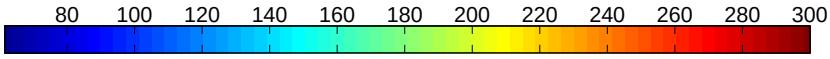
The three-dimensional embeddings output by the four dimensionality reduction methods are shown in Figure 7.1. The pitch of each unit is illustrated as a colour, as indicated in Figure 7.1(a). It should also be noted that the f0 values are estimates in this experiment and as a result may be subject to some error. This was not the case in the synthetic

speech experiments in which the f0 value was known and controlled.

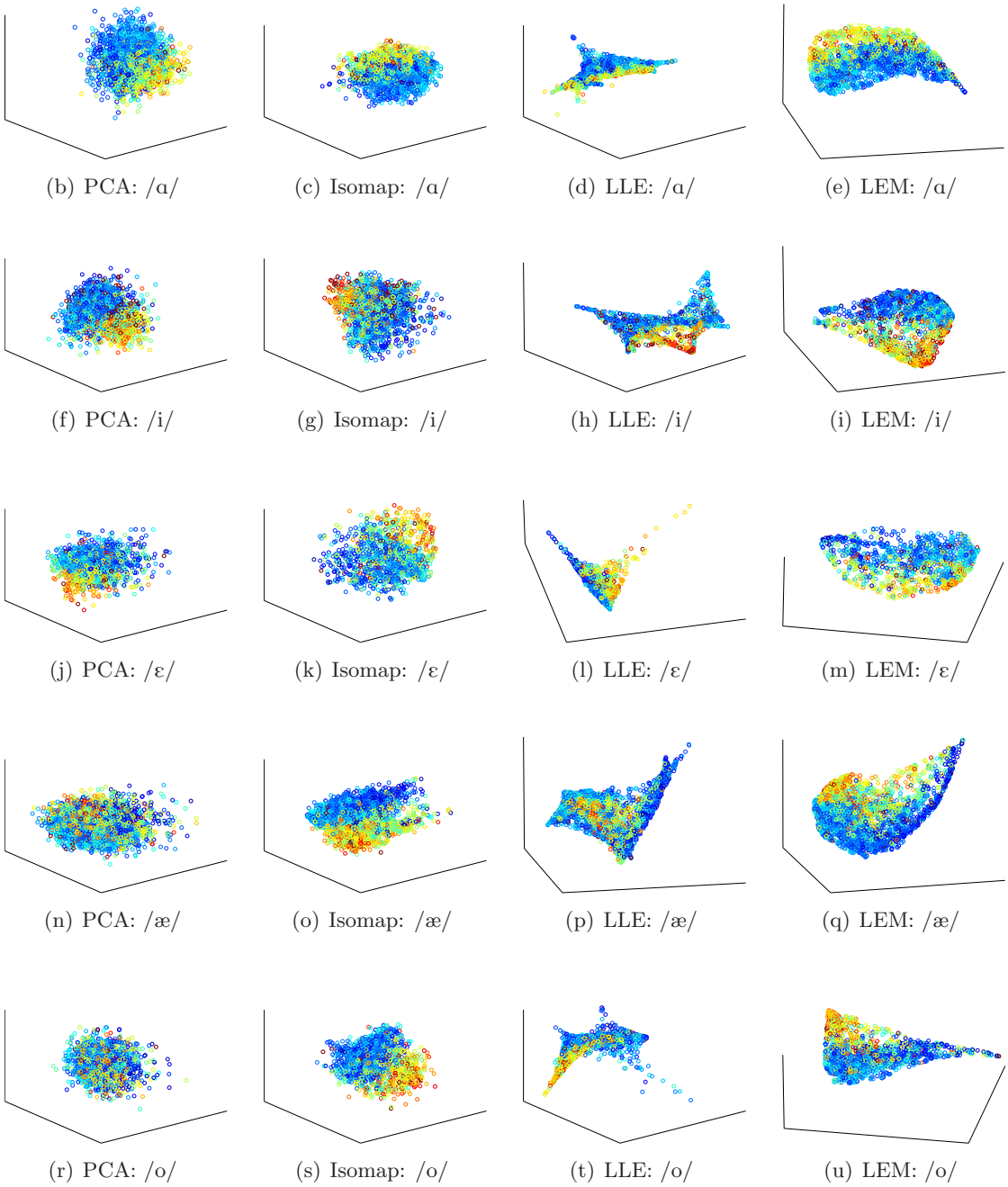
Each dimensionality reduction method produced a different embedding space for each of the five vowels. In most of the spaces low-dimensional f0 variation can be seen, with low pitch units clustered separately from higher pitch units. Visually comparing the embeddings produced by the various dimensionality reduction methods, Laplacian eigenmaps and Isomap consistently produced embeddings in which f0 information is accurately retained. However, while the LLE algorithm also facilitates visualisation of f0 variation, f0 is less well represented in LLE space. PCA, the linear method, produced three-dimensional representations similar to Isomap for the vowels: /ɑ/, /i/, and /ɛ/. However, PCA produced the worst visualisations of all four methods for the remaining two vowels, /æ/ and /o/, suggesting that the manifold learning methods may be more suitable as a tool to visualise f0 variation in speech data.

While information relating to pitch variation is visible in Figure 7.1, it is less well defined and consistent than was the case for the synthetic data, as shown in Figure 6.5. This is to be expected as the data used in the synthetic experiment was very consistent, with constant formant frequencies and no noise corruption; in contrast to the TIMIT data which is sampled from a large number of speakers resulting in larger prosodic and formant frequency variation. As a contrast we briefly examined the case of phones uttered by a single speaker. We computed time delay embeddings of single units of each of the five vowels—/ɑ/, /i/, /ɛ/, /æ/, and /o/—taken from a single male TIMIT speaker. These time delay embeddings were computed using Takens' (1981) method, as discussed in Section 3.3, and are shown in Figure 7.2.[1] The underlying geometric structure of the individual vowels is evident in Figure 7.2; however when many such vowels are analysed together the increased prosodic and formant frequency variation makes it more difficult to determine a consistent underlying geometric structure—hence the inconsistencies in the f0 visualisations of Figure 7.1, where a large number of vowels from a large number of speakers have been analysed.

[1]The time delay embeddings in Figure 7.2 are not as 'clean' as those presented in Figure 3.5 as the former were computed from vowels extracted from continuous speech while the latter were computed from sustained vowel sounds.



(a) Colour bar: f0 values (Hz) corresponding to the plots below.



(b) PCA: /ɑ/ (c) Isomap: /ɑ/ (d) LLE: /ɑ/ (e) LEM: /ɑ/

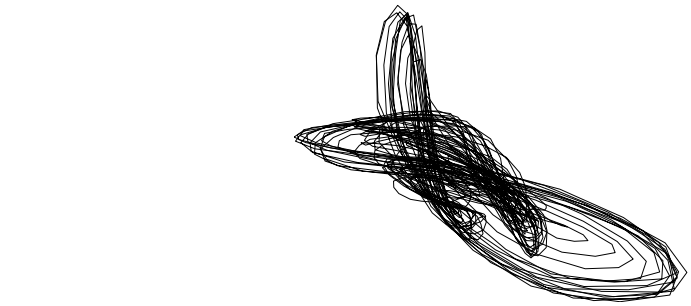
(f) PCA: /i/ (g) Isomap: /i/ (h) LLE: /i/ (i) LEM: /i/

(j) PCA: /ɛ/ (k) Isomap: /ɛ/ (l) LLE: /ɛ/ (m) LEM: /ɛ/

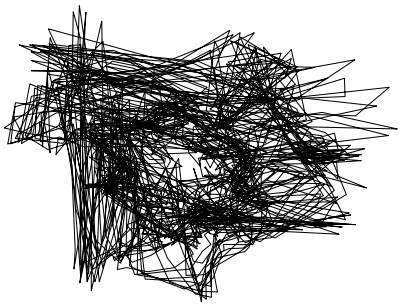
(n) PCA: /æ/ (o) Isomap: /æ/ (p) LLE: /æ/ (q) LEM: /æ/

(r) PCA: /o/ (s) Isomap: /o/ (t) LLE: /o/ (u) LEM: /o/

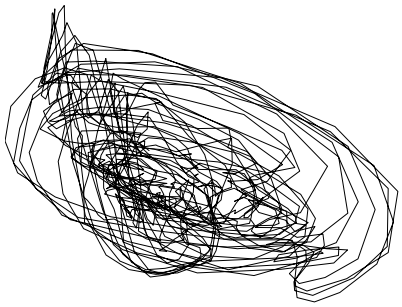
Figure 7.1: Visualisation of f0 variation in three-dimensional embedding spaces produced by PCA, Isomap, LLE, and Laplacian eigenmaps.



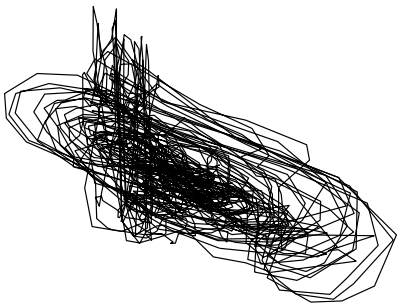
(a) /ɑ/



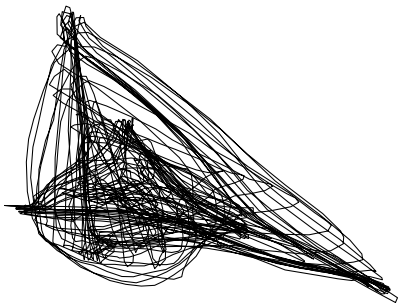
(b) /i/



(c) /ɛ/



(d) /æ/



(e) /o/

Figure 7.2: Two-dimensional time delay embeddings of the vowels /ɑ/, /i/, /ɛ/, /æ/, and /o/; $\tau = 1.25\,\text{ms}$.

7.3 Visualisation of vowel variation

7.3.1 Introduction

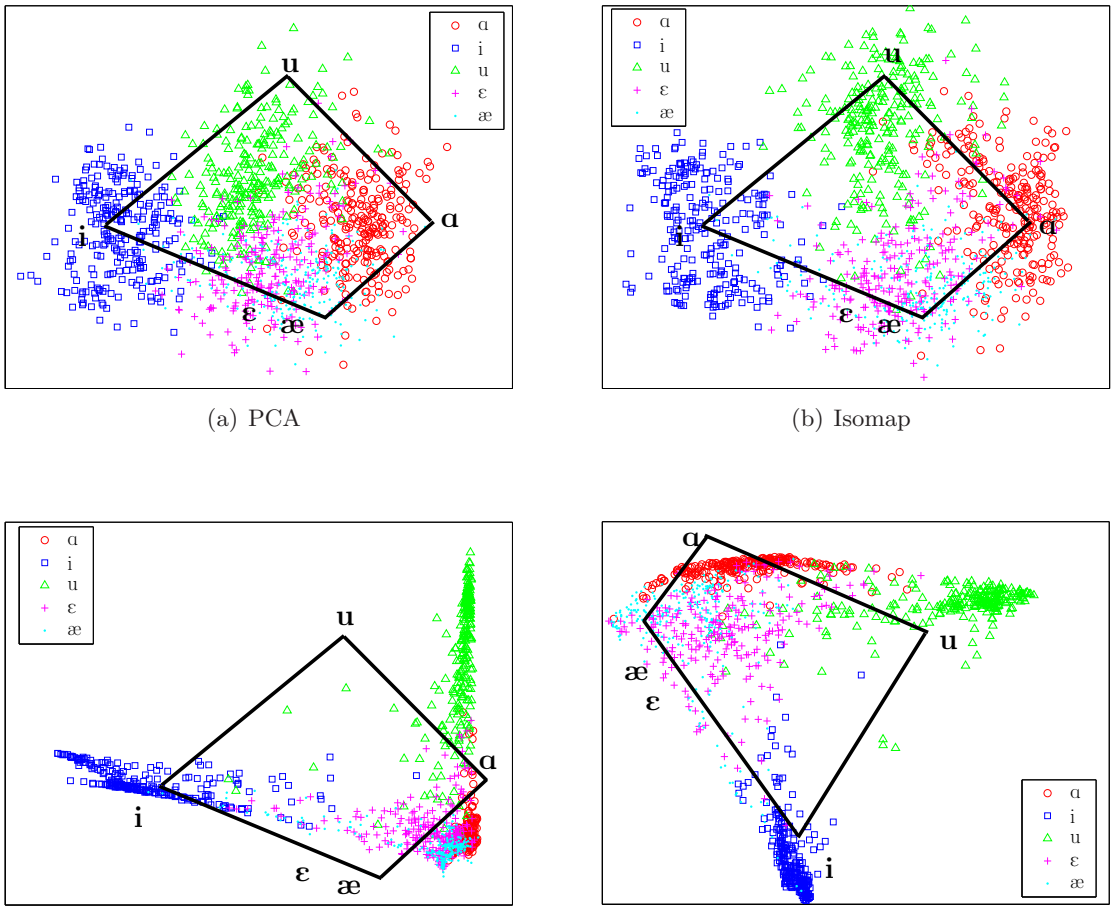
In addition to visually investigating the ability of the various dimensionality reduction methods to preserve f0 variation in two- and three-dimensional embeddings, experiments were also conducted to visually inspect if the methods were capable of retaining phone specific information. The methodology used is reported in the next section and a discussion of the results of these experiments then follows.

7.3.2 Experiment

The following five vowels were chosen for analysis: /ɑ/, /i/, /ɛ/, /æ/, and /u/. For each phone, 250 randomly selected units were extracted from the TIMIT corpus. For each phone unit, a frame of length 40 ms was extracted from the phone center. The raw speech frames were then preemphasised, $p = 0.98$, and Hamming windowed. Following this, 13-dimensional MFCC vectors, including the zeroth cepstral coefficients, were computed for each frame. These MFCC vectors were used as the high-dimensional speech representation in this experiment. The MFCC features from all five vowels were combined to form an input data set. This input data set was then provided to the dimensionality reduction algorithms—Isomap, LLE, Laplacian eigenmaps, and PCA—and four different two-dimensional embeddings of the original 13-dimensional input data set were output. As above, a k value of 5 was used for all the manifold learning methods. The output two-dimensional embeddings were then visually inspected to determine if any phone-related information had been retained.

7.3.3 Results and discussion

The two-dimensional embeddings output by each of the four dimensionality reduction methods are shown in Figure 7.3. Each point represents a single phone unit, with units from different phones depicted using different symbols, as indicated in the legends. A simplified representation of the IPA vowel chart, as discussed in Section 3.1, containing only the five relevant phones is overlaid on each two-dimensional space in order to aid visualisation of the structure of the vowel manifold discovered by the dimensionality reduction



(a) PCA

(b) Isomap

(c) LLE

(d) Laplacian eigenmaps

Figure 7.3: Two-dimensional embeddings produced by applying dimensionality reduction methods to 250 units of each of the five vowels: /ɑ/, /i/, /ɛ/, /æ/, and /u/. A simplified representation of the corresponding IPA vowel chart is overlaid on each embedding.

methods.

In each of the two-dimensional spaces a number of phone clusters are visible. This is perhaps most evident in the spaces output by PCA and Isomap, Figures 7.3(a) and 7.3(b). In these spaces the different phones are separated into individual clusters, with the exception of /ɛ/ and /æ/ which have a larger degree of overlap—a result of the similarity in articulation of these two phones. Similar phone clusters are also visible in the two-dimensional spaces output by LLE and Laplacian eigenmaps, Figures 7.3(c) and 7.3(d). However, the clusters are visually less well separated in the two-dimensional space produced by LLE.

Comparing these two-dimensional embeddings to the IPA vowel chart, which depicts the articulation of a number of different vowel sounds with respect to the two dimensions of tongue frontness and height, a correspondence was found. This correspondence is illus-

trated by the IPA vowel charts overlaid on the two-dimensional embeddings in Figure 7.3. In each case the IPA chart has been orientated to, approximately, align with the phone data in the two-dimensional space. The locations of the phone clusters in two-dimensional space can be seen to correspond with the position of the phones on the IPA chart. Thus, the information retained by the dimensionality reduction methods is closely related to the dimensions of the IPA chart—namely, tongue frontness and height or, similarly, F1 and F2 as discussed in Section 3.1. Visual inspection reveals that the spaces output by PCA and Isomap are quite similar and correspond closely to the IPA chart, whereas the spaces output by LLE and Laplacian eigenmaps are more 'deformed' and match the IPA chart less closely. This is likely due to the differing motivations of the methods, with PCA and Isomap—global methods—preserving the global geometry of the manifold, while LLE and Laplacian eigenmaps—local methods—attempt to maintain the local neighbourhood relationships. Thus, Figure 7.3 illustrates the importance of retaining global geometric structure.

This experiment provides some insight into the ability of the various methods to discover underlying manifold structure in speech data. More objective and quantifiable comparisons of the spaces output by the dimensionality reduction methods are detailed in the following sections.

7.4 Phone classification using magnitude spectra based features

7.4.1 Introduction

In Section 6.4, the ability of the four dimensionality reduction methods to produce meaningful low-dimensional representations of high-dimensional speech features was evaluated in a number of synthetic vowel classification experiments. This section presents the results of a number of similar experiments conducted to evaluate the performance of the dimensionality reduction methods in a number of natural speech classification experiments. These experiments also extended the feature set used; while the synthetic speech experiments used static features only, the studies in this section also investigate the incorporation of dynamic features, as are conventionally used in speech recognition tasks.

The principal purpose of the experiments described in this section is to evaluate the ability of the various methods to reduce the dimensionality of features conventionally used in speech recognition tasks. This evaluation provides an insight into the inherent dimensionality of the speech space represented by these high-dimensional features and also serves to display the potential value of the dimensionality reduction methods in speech recognition applications.

7.4.2 Experiments

The phone classification experiments were conducted in a manner similar to those described in Section 6.4; although using natural speech rather than synthetic speech. As before, SVM classifiers with RBF kernels were used in all classification experiments.

Phone classification experiments were performed using five different feature types: baseline MFCC vectors and features produced by applying PCA, Isomap, LLE, and Laplacian eigenmaps to the baseline MFCC vectors. Two types of baseline MFCC vectors were used: standard static MFCCs only and static MFCCs concatenated with dynamic information. This dynamic information took the form of delta coefficients, as previously discussed in Section 5.3.3. The experimental procedure detailed below was repeated separately for the baseline MFCCs both with and without deltas.

Each of the five feature types were evaluated in the three phone classification tasks detailed in Section 5.5.2: distinguishing between five vowels; ten vowels; and classifying 19 different phones into their associated phone classes, of which there are five.

Based on the phonetic transcriptions and associated phone boundaries provided in TIMIT all units of the phones required for the classification tasks were extracted from the corpus. For each phone unit, frames of length 40 ms were extracted with a frame shift of 20 ms. Units of duration less than 100 ms were discarded. The raw speech frames were preemphasised, $p = 0.98$, and Hamming windowed. Following this preprocessing 13-dimensional MFCC vectors, including the zeroth cepstral coefficients, were computed for each frame. Standard delta coefficients, Δ, were also computed. These MFCC vectors and those concatenated with their deltas, MFCC+Δ, serve as both baseline features and high-dimensional inputs for PCA, Isomap, LLE, and Laplacian eigenmaps.

For each of the three phone classification experiments, 250 units representing each of

the required phones were chosen at random from those extracted above to make up the data set. PCA, Isomap, LLE, and Laplacian eigenmaps were individually applied to the equivalent sets of MFCC and MFCC+Δ vectors. The number of nearest neighbours, k, used in Isomap, LLE, and Laplacian eigenmaps was set empirically.

In order to examine the ability of the feature transformation methods to compute concise representations of the input vectors retaining discriminating information, the dimensionality of the resulting feature vectors was varied from 1 to D; where $D = 13$ for static MFCC features and $D = 26$ for MFCC+Δ features. A separate classifier was subsequently trained and tested using feature vectors with each of the different dimensionalities. Thus, the ability of these feature transformation methods to produce useful low-dimensional features could be evaluated and changes in performance with varying dimension analysed. As a baseline the original MFCC and MFCC+Δ vectors were used, also varying in dimensionality as detailed above.

In all classification experiments 80% of the data was assigned as training data with the remaining 20% withheld and used as testing data. The data was partitioned such that the training and test sets had no speakers in common, thus ensuring speaker independence.

7.4.3 Results

Static Features

Firstly, the results of experiments conducted using 13-dimensional MFCCs as baseline feature vectors and inputs to the dimensionality reduction methods are presented. In each experiment the classifier was evaluated on each of the five feature types: baseline MFCC vectors and PCA, Isomap, LLE, and Laplacian eigenmaps embeddings of these baseline vectors. The dimensionality of the feature vectors used in the experiment vary from 1 to 13—the original, full dimensionality.

Figure 7.4 shows the results of the five vowel classification task using the baseline MFCC, PCA, Isomap, LLE, and Laplacian eigenmaps features. Results are presented for evaluation on both the training data and testing data. The percentage of phones correctly classified is given on the vertical axis. The horizontal axis represents the dimensionality of the feature vector. The results in Figure 7.4 can be summarised as follows:

- The performance of the baseline MFCC vectors improved with increasing dimen-

sionality.

- PCA features offered improvements over baseline MFCCs in most, 76.92%, of the dimensions tested. This improvement is largest in the lower dimensions.

- For the training data, maximum classification accuracy in dimensions 2–13 was demonstrated with Isomap features, outperforming all other features including the original 13-dimensional MFCC vectors.

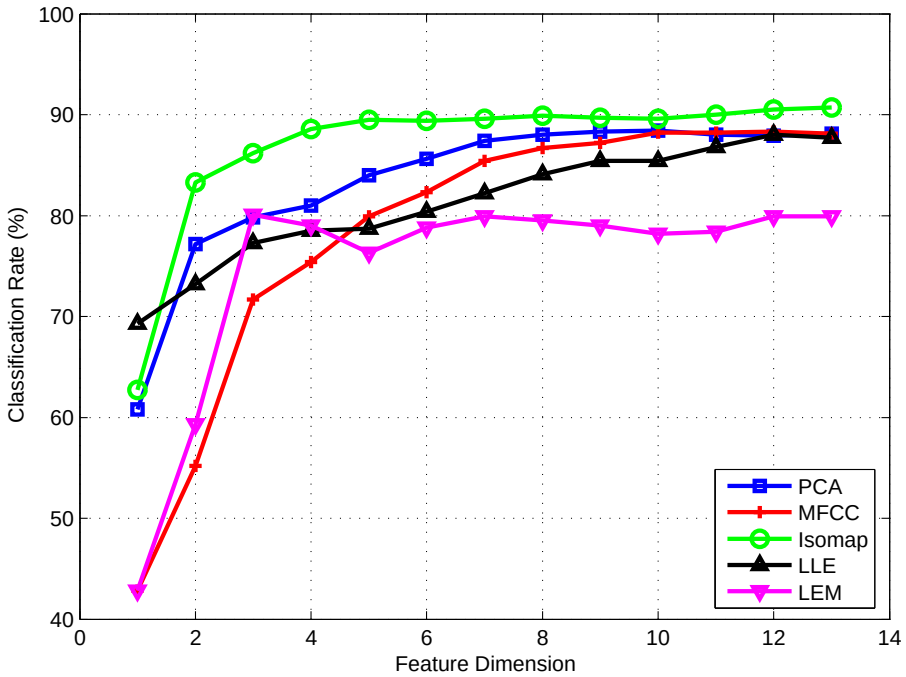
- Isomap features also offered performance comparable to, and in some dimensions better than, other features on the testing data. In fact, Isomap yielded 78% accuracy with only two dimensions—as shown in Figure 7.4(b)—the other feature types required a much greater number of dimensions, $d > 10$, to reach this level of classification accuracy. Interestingly, the classification rates achieved using the subsequent, higher-dimensional Isomap features, $2 < d < 11$, are lower than that of the two-dimensional features. This indicates that a substantial amount of discriminatory information is compacted into the first two dimensions and that the subsequent dimensions, $2 < d < 11$, add no useful information; rather they degrade the classifier's performance.

- LLE features yielded improved performance over other features in low dimensions, $d < 3$. However, in higher dimensions LLE features did not consistently offer a performance increase over other methods.

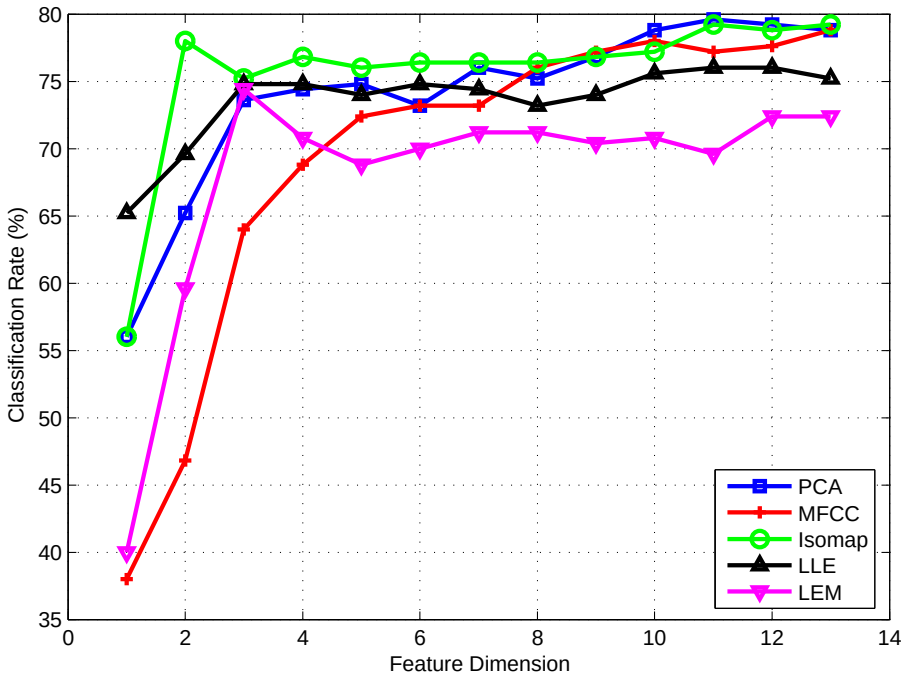
- Laplacian eigenmaps features outperformed the baseline MFCCs when $d \leq 4$ but yielded worse performance than all other features for all higher dimensionalities.

- Comparing the results of evaluations on the training data with those run on the the testing data it can be observed that the mean classification accuracy for training data classification was higher than that of the testing data classification. However, the difference in performance on training data and unseen test data was relatively small, indicating no significant classifier overfitting has occurred.

The mean classification accuracy results for each feature type in the five vowel classification task are summarised in Table 7.1. The mean accuracy scores were computed for the testing data evaluation. Averages are computed for three dimensionality ranges. It can be seen



(a) Training Data



(b) Testing Data

Figure 7.4: Five vowel classification results for baseline MFCC, PCA, Isomap, LLE and Laplacian eigenmaps features on data from the TIMIT speech corpus.

Dimensions	MFCC	PCA	Isomap	LLE	LEM
1–4	54.400	67.300	71.500	71.100	61.200
5–13	75.956	76.933	77.378	74.800	70.756
1–13	69.323	73.969	75.569	73.662	67.815

Table 7.1: Mean classification accuracy, computed for the testing data evaluation, in the five vowel classification task for MFCC, PCA, Isomap, LLE, and Laplacian eigenmaps features.

that Isomap resulted in the highest average accuracy overall, followed by PCA, LLE, MFCC, and finally Laplacian eigenmaps. LLE and Isomap both performed better than PCA and MFCC in low dimensions.

Results for ten vowel classification are shown in Figure 7.5. Results are presented for evaluation on the testing data only; results of evaluations conducted on the training data are provided in Appendix B, Figure B.1. The results are similar to those of the task above, with reduced classification accuracy due to increased complexity of the classification task and possibility of phone confusion. The important findings may be summarised as follows:

- Isomap yielded better performance than MFCCs and PCA-transformed MFCCs for the testing data in low dimensions ($d < 8$).
- A classification accuracy of 57.8% was achieved on the testing data with Isomap features of only $d = 3$. This classification accuracy was only reached by higher-dimensional, $d \geq 8$, MFCC and PCA features. For purposes of comparison, it should be noted that as this classification task involves five classes the expected classification rate by naïve random guessing would be 20%.
- Laplacian eigenmap's performance was comparatively worse than in the previous task, yielding the worst performance in almost all dimensionalities; excluding $d = 1$ for the training data and $d = 1, 2$ for the test data where Laplacian eigenmaps outperformed the baseline MFCCS.

The mean classification accuracy results for each feature type in the ten vowel classification task are presented in Table 7.2. Again, Isomap resulted in the highest average accuracy overall. Also, LLE and Isomap both performed better than PCA and MFCC in low dimensions.

Phone class classification results are presented in Figure 7.6. Results are presented for

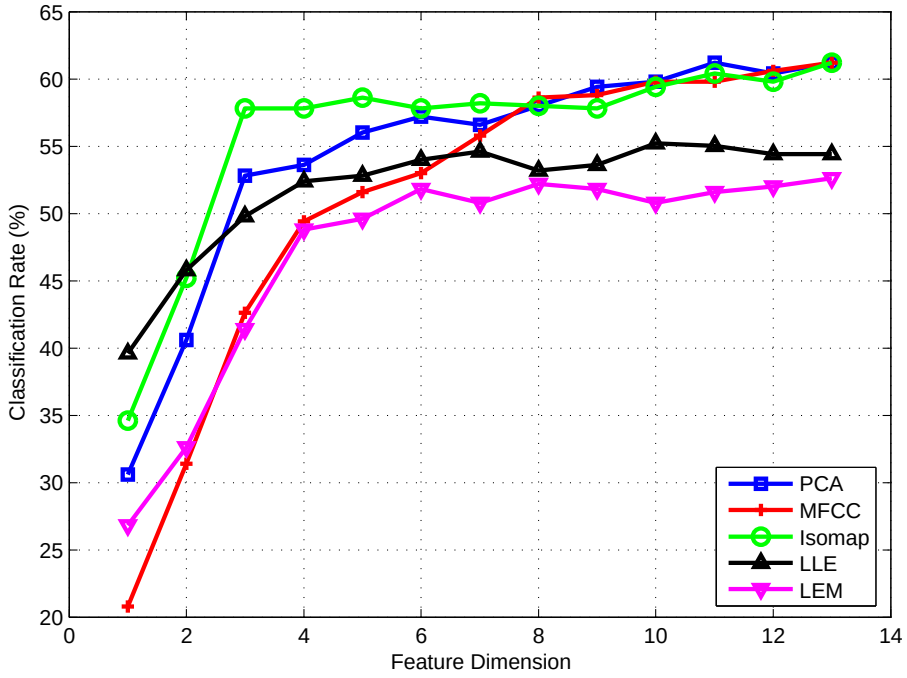


Figure 7.5: Ten vowel classification results for baseline MFCC, PCA, Isomap, LLE and Laplacian eigenmaps features on data from the TIMIT speech corpus. Evaluation performed on testing data.

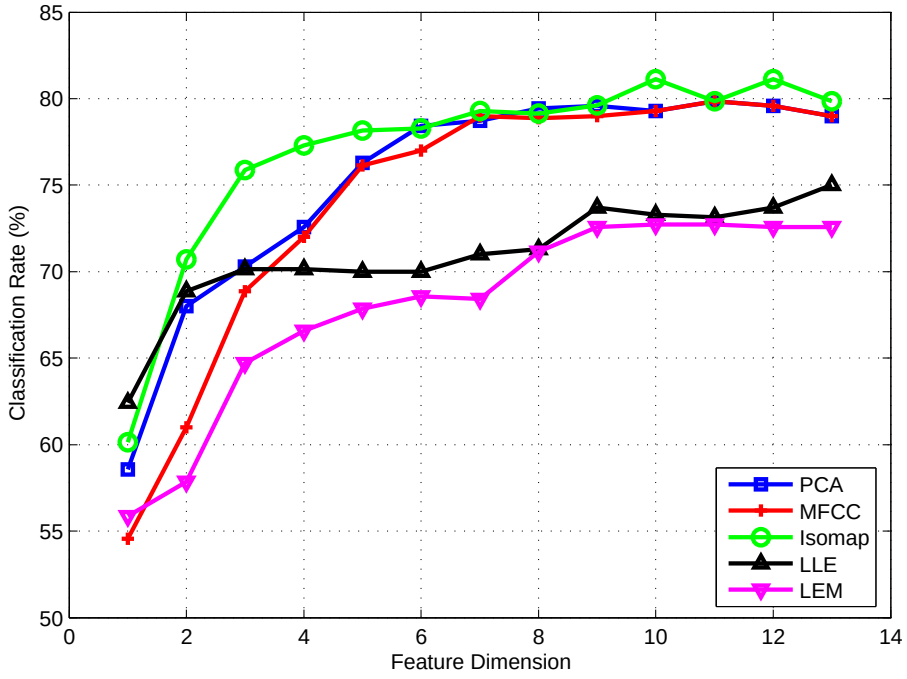


Figure 7.6: Phone class classification results for baseline MFCC, PCA, Isomap, LLE and Laplacian eigenmaps features on data from the TIMIT speech corpus. Evaluation performed on testing data.

Dimensions	MFCC	PCA	Isomap	LLE	LEM
1–4	36.050	44.400	48.850	46.900	37.400
5–13	57.689	58.867	59.022	54.133	51.467
1–13	51.031	54.415	55.892	51.908	47.138

Table 7.2: Mean classification accuracy, computed for the testing data evaluation, in the ten vowel classification task for MFCC, PCA, Isomap, LLE, and Laplacian eigenmaps features.

Dimensions	MFCC	PCA	Isomap	LLE	LEM
1–4	64.107	67.357	71.000	67.893	61.250
5–13	78.635	78.905	79.607	72.349	71.016
1–13	74.165	75.352	76.959	70.978	68.011

Table 7.3: Mean classification accuracy, computed for the testing data evaluation, in the phone class classification task for MFCC, PCA, Isomap, LLE, and Laplacian eigenmaps features.

evaluation on the testing data only; results of evaluations conducted on the training data are provided in Appendix B, Figure B.2. The following is evident:

- Again, LLE features performed well in very low dimensions, $d < 3$, but yielded low classification rates in higher dimensions; relative to MFCC, PCA, and Isomap.
- Isomap features yielded the best accuracy in the majority of dimensions tested.
- PCA and MFCC features yielded similar performance, with PCA features offering improved accuracy for low-dimensional features.
- Again, the embeddings produced by Laplacian eigenmaps resulted in comparatively poor classification accuracy.

The mean classification accuracy results for each feature type in the phone class classification task are summarised in Table 7.3.

Dynamic Features

As detailed in the previous section, experiments were also performed using 26-dimensional MFCC+Δ vectors as high-dimensional inputs to the four dimensionality reduction methods. The results of performing phone classification using the features output by each of these methods and the original MFCC+Δ vectors are shown in Figures 7.7, 7.8, and 7.9. Ten vowel and phone class classification results are presented for evaluation on the

testing data only; results of evaluations conducted on the training data are provided in Appendix B, Figures B.3 and B.4.

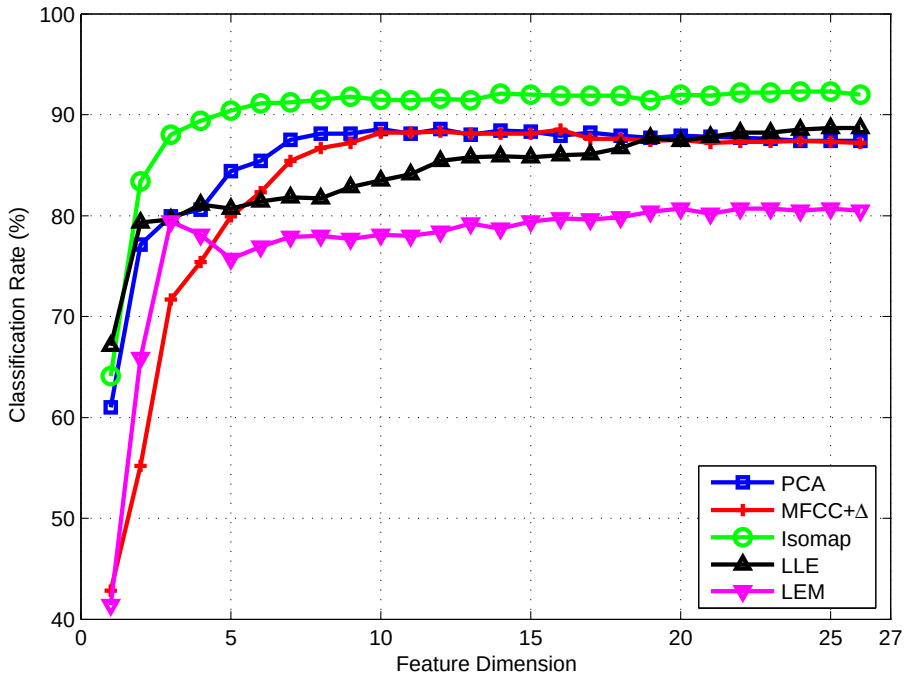
It can be seen that these results are similar to those using the static features. Again, PCA-transformed features yielded similar performance to the baseline features, with PCA-transformed features offering improved accuracy for low-dimensional features. The reason PCA-transformed features yielded higher classification rates than the original baseline features in low dimensions is because the linear transformation used in PCA, described in Section 4.3.1, effectively compresses the principal sources of variation, the discriminatory information, into the lowest dimensions. Thus the PCA-transformed features have more discriminatory information in the lower dimensions than the baseline MFCC features.

The manifold learning methods offered improved performance over both MFCC+Δ and PCA-transformed features in low dimensions; with the exception of Laplacian eigenmaps, the performance of which was again inconsistent and frequently lower than all other features. In general, features output by Isomap offered the best performance, outperforming all other feature types in 85.89% of the classification tests performed on dynamic features.

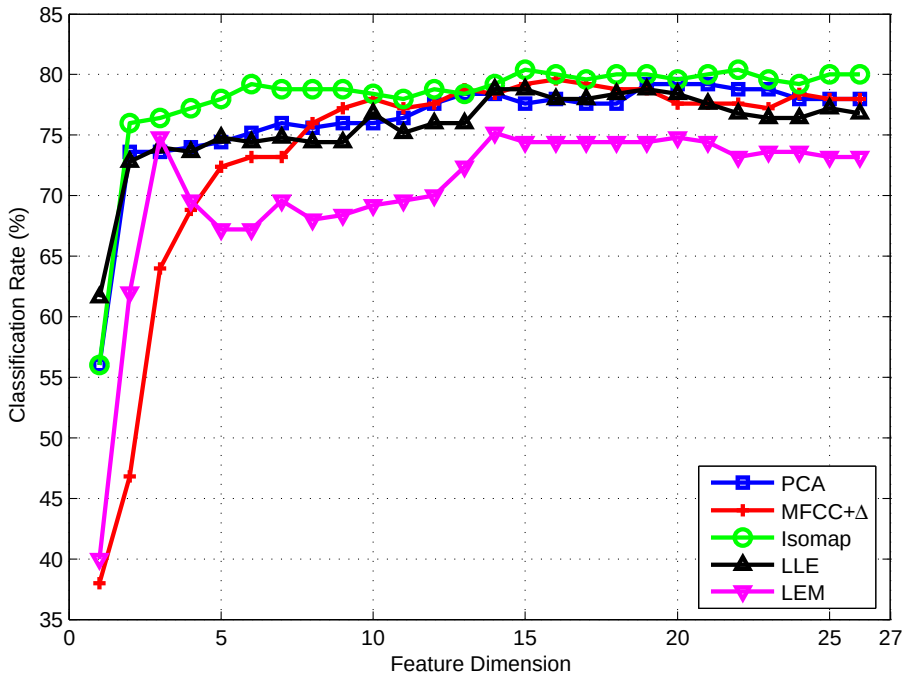
7.4.4 Conclusions

In this study a phone classification approach using nonlinear manifold learning based feature transformation was proposed and evaluated against a baseline linear dimensionality reduction method, PCA, and conventional MFCC features. All of the dimensionality reduction methods presented, with the exception of Laplacian eigenmaps, consistently outperformed the baseline MFCC features for low dimensions. This illustrates the capability of these methods to extract discriminating information from the original MFCC features.

Examining the general trends of the various feature types in Figures 7.4–7.9 as the dimensionality of the vectors used for classification increases, an 'elbow' is visible between $d = 2, \ldots, 4$. At this point the classification accuracy begins to plateau and the addition of further dimensions does not cause a significant increase in accuracy. This indicates that the speech data has an inherent low-dimensional structure. Further, this 'elbow' is most prominent in the classification rates achieved using Isomap-transformed features. This suggests that the manifold learning method is the most capable of discovering this



(a) Training Data



(b) Testing Data

Figure 7.7: Five vowel classification results for baseline MFCC+Δ, PCA, Isomap, LLE and Laplacian eigenmaps features on data from the TIMIT speech corpus.

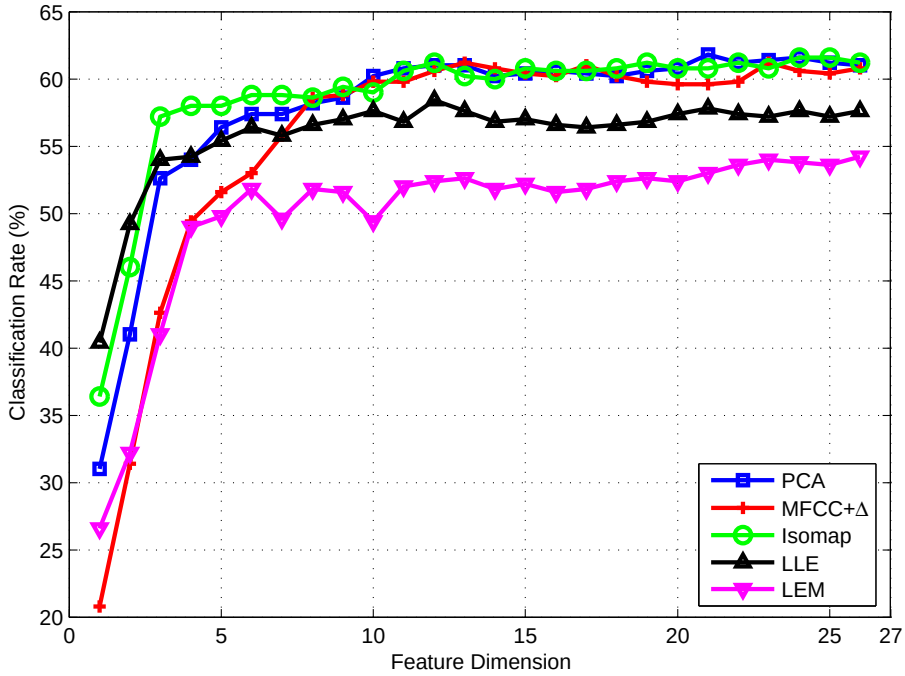


Figure 7.8: Ten vowel classification results for baseline MFCC+Δ, PCA, Isomap, LLE and Laplacian eigenmaps features on data from the TIMIT speech corpus. Evaluation performed on testing data.

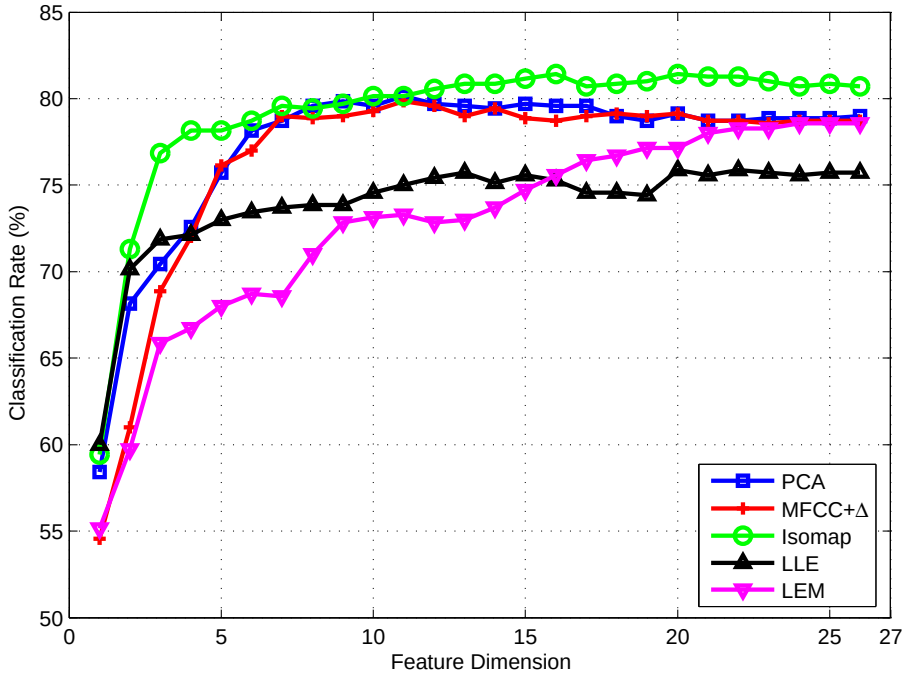


Figure 7.9: Phone class classification results for baseline MFCC+Δ, PCA, Isomap, LLE and Laplacian eigenmaps features on data from the TIMIT speech corpus. Evaluation performed on testing data.

underlying low-dimensional structure.

Higher classification accuracy is shown for the Isomap- and LLE-derived features compared to baseline MFCC and PCA-transformed features for low dimensions. This indicates that these manifold learning algorithms are more capable of retaining information required to discriminate between phones, especially in low-dimensional space, when compared to the linear method. This may be due to the ability of these methods to exploit nonlinear structure in the speech space.

In general, Isomap was found to yield superior performance to both MFCC and PCA features. Comparing the manifold learning methods, Isomap generally demonstrated better classification accuracy than LLE and Laplacian eigenmaps. The relative success of the Isomap algorithm indicates that preserving global structure rather than local relationships may be more important for speech feature transformation.

7.5 Phone classification: Comparison and combination of features derived from the magnitude and phase spectrum

7.5.1 Introduction

The experiments described in this section are similar to those described in Section 7.4. Phone classification tasks were performed using features of varying dimensionalities produced by applying a number of dimensionality reduction methods to high-dimensional speech features. However, rather than simply using conventional features derived from the magnitude spectrum, MFCCs, we also investigated features derived from the phase spectrum, MODGDFs. The primary aim of these classification experiments was to compare how much meaningful discriminatory information is contained in MFCC and MODGDF representations and investigate the low-dimensional embeddings produced by applying the dimensionality reduction methods to these representations. The motivation behind such experiments is to compare the low-dimensional structure of MFCCs and MODGDFs.

In addition to using each feature set alone, MFCC and MODGDF features have previously been concatenated and the resulting joint feature vectors shown to improve speech recognition performance, indicating that they may contain complementary infor-

mation (Alsteris and Paliwal, 2005; Hegde et al., 2007a). Building on such findings, we examine the performance of these joint features in phone classification tasks and propose a method to reduce the dimensionality of these joint features in an attempt to improve classification accuracy without increasing the computational cost associated with processing the higher-dimensional features.

7.5.2 Experiments

The objective of these experiments was to perform phone classification using MFCCs, MODGDFs, and the low-dimensional feature representations resulting from the application of PCA, Isomap, LLE, and Laplacian eigenmaps to these high-dimensional features. As a result, an experimental setup similar to that described in Section 7.4 above was used.

However, as these experiments used MODGDFs, in addition to MFCCs as used previously, the frame size and overlap used were altered based on the recommendations of previous studies of MODGDF extraction. For each of the phones required for the three phone classification tasks, as detailed in Section 5.5.2, frames of duration 20 ms were extracted with a frame shift of 10 ms. This frame rate was chosen based on parameters used in previous speech recognition studies using MODGDFs (Hegde et al., 2007b). This frame rate was also used for extraction of the MFCC features to ensure consistency between the two feature sets.

7.5.3 Results

Comparison of baseline MFCC and MODGDF features

Results of each classification experiment using full dimensional MFCC and MODGDF feature sets are shown in Table 7.4. MFCCs were found to outperform MODGDFs in each test, both with and without the inclusion of delta, Δ, coefficients. Previously published results comparing speech recognition performance using MFCC and MODGDF features are inconsistent, with some studies showing better accuracy with MFCCs (Alsteris and Paliwal, 2005) while other studies indicate the opposite (Murthy and Gadde, 2003). This may be due to various inconsistencies in the corpora, feature extraction procedures, and classification algorithms used.

When MODGDF and MFCC feature vectors were concatenated and used as features

Feature Set	Dim.	Classification Task		
		Phone Class	Ten Vowel	Five Vowel
MFCC	13	79.714	59	76
MODGDF	13	75.143	54	67.6
MFCC+Δ	26	80.286	61.6	76
MODGDF+Δ	26	75.571	57.4	69.2
MODGDF+MFCC	26	82	61.8	76.8
[MODGDF+Δ] + [MFCC+Δ]	52	82	62	76.8

Table 7.4: Vowel and phone class classification accuracy (%) using baseline MFCC and MODGDF features. Feature dimensionality (Dim.) is also shown. The symbol + indicates feature concatenation.

in the various classification tasks the resulting classification accuracies were found to be higher than when using the baseline MFCC features. An increase of 0.8% was found in the five vowel classification task, and larger increases of 2.286% and 2.8% observed in the phone class and ten vowel classification tasks, respectively. This indicates that the magnitude and phase spectrum contain complementary information; these findings are consistent with previously published results (Alsteris and Paliwal, 2005; Hegde et al., 2007b). Further improvements were observed in the ten vowel task when delta coefficients were also included.

Reduced dimensionality: Static features

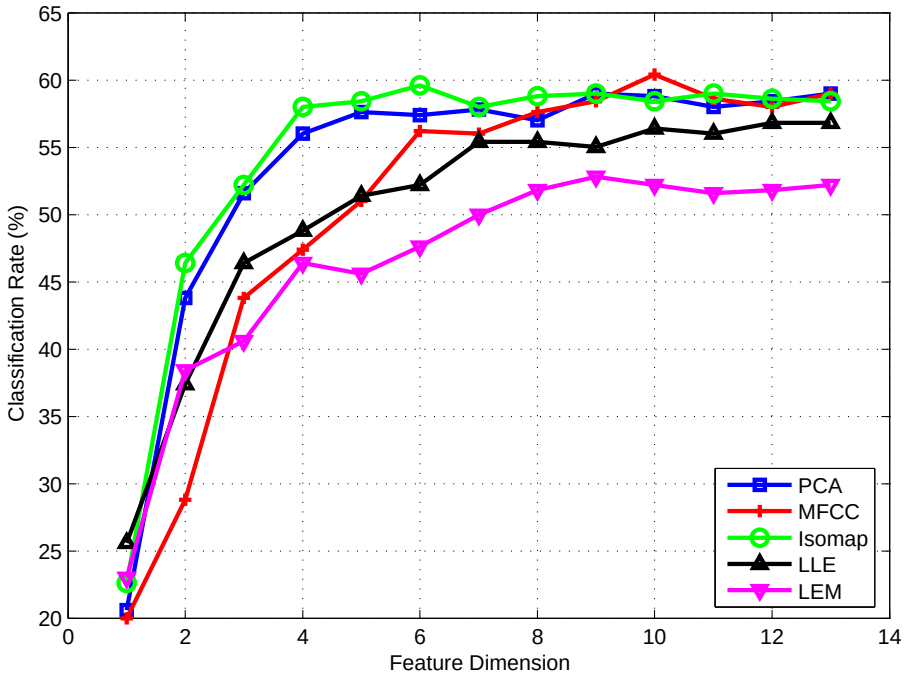
In this section we discuss the results of the application of dimensionality reduction methods to static MFCC and MODGDF features in an attempt to determine if these representations have underlying low-dimensional structure. PCA, Isomap, LLE, and Laplacian eigenmaps were each applied to both MFCCs and MODGDFs and the resulting features evaluated in the three classification experiments detailed in Section 5.5.2. In each experiment SVM classifiers were trained and tested on MFCCs, MODGDFs, and features resulting from dimensionality reduction of these baseline features using PCA, Isomap, LLE, and Laplacian eigenmaps. The dimensionality of the feature vectors used in the experiments was varied from 1 to the original dimensionality; this was 13 for static features.

Results of the ten vowel classification experiments are shown in Figure 7.10; results of the five vowel and phone class classification experiments were consistent with these results and are included in Appendix B. Figure 7.10(a) illustrates results using MFCCs and Figure 7.10(b) shows results for MODGDFs. The percentage of phones correctly classified

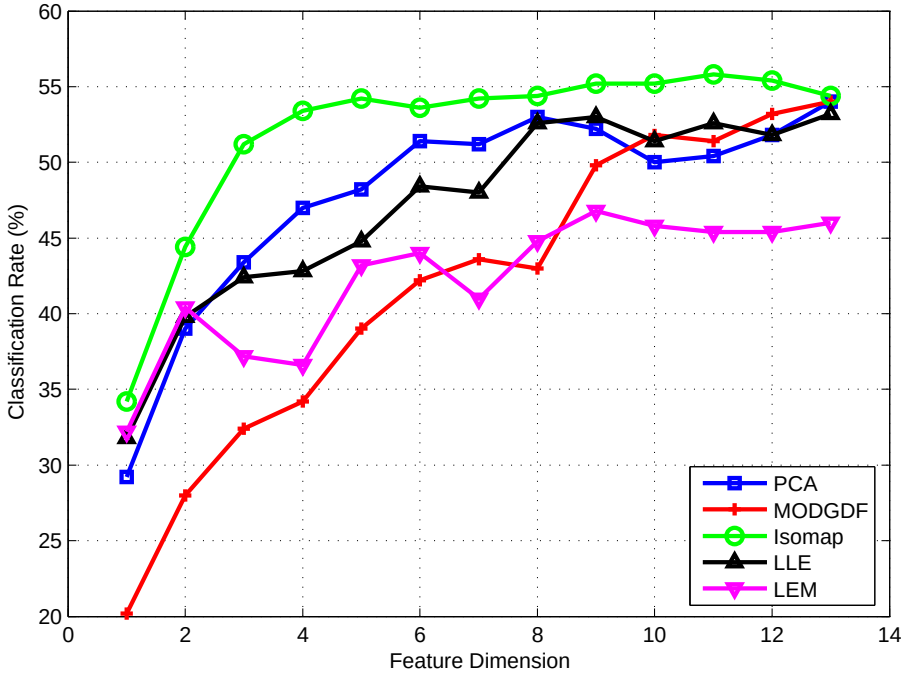
is given on the vertical axis. The horizontal axis represents feature vector dimensionality.

The results may be summarised as follows:

- MFCCs were found to outperform MODGDFs for all feature dimensionalities.
- MFCCs also reached a performance plateau more rapidly than MODGDFs indicating that discriminatory information is more compactly represented in MFCCs. This is supported by the observation that the dimensionality reduction methods offered greater relative improvement in accuracy for MODGDFs compared to MFCCs.
- The dimensionality reduction methods were found to offer improved classification accuracy over the baseline features, with at least one dimensionality reduction method outperforming the baseline features in all but one test. This improvement was largest for low-dimensional feature vectors. The improved performance in low dimensions as a result of dimensionality reduction suggests that both magnitude and phase information has an intrinsic low-dimensional structure and may benefit from dimensionality reduction prior to use.
- Isomap yielded the highest classification accuracy in 76.9% of the tests performed. This finding is consistent with experiments performed in Section 7.4. This suggests the presence of nonlinear structure in the data which the linear PCA algorithm is incapable of finding but the manifold learning algorithm is able to exploit.
- LLE was found to outperform both MFCC and MODGDF baseline features in low dimensions; $d <= 5$ and $d <= 9$ respectively. However, in higher dimensions the baseline features yielded higher classification accuracy.
- Laplacian eigenmaps performance was poor, as in Section 7.4.3, not once yielding the highest classification accuracy. However, in the vowel classification tasks Laplacian eigenmaps-transformed MODGDFs outperformed the original MODGDF features for $d < 7$, supporting the above claim dimensionality reduction may be beneficial when using MODGDF features.



(a) MFCC



(b) MODGDF

Figure 7.10: Ten vowel classification accuracy using (a) MFCC and (b) MODGDF features. The performance of each feature after dimensionality reduction by PCA, Isomap, LLE, and LEM is also shown.

Reduced dimensionality: Dynamic features

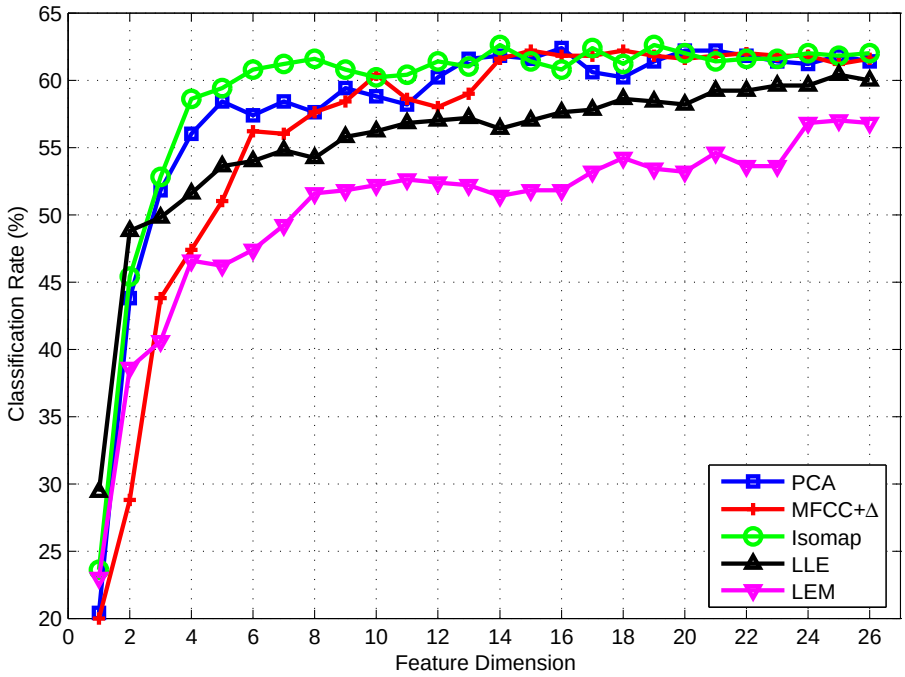
Results of the ten vowel classification experiments using MFCC+Δ and MODGDF+Δ feature vectors are shown in Figure 7.11; results of the five vowel and phone class classification experiments are available in Appendix B.

From these results it is evident that:

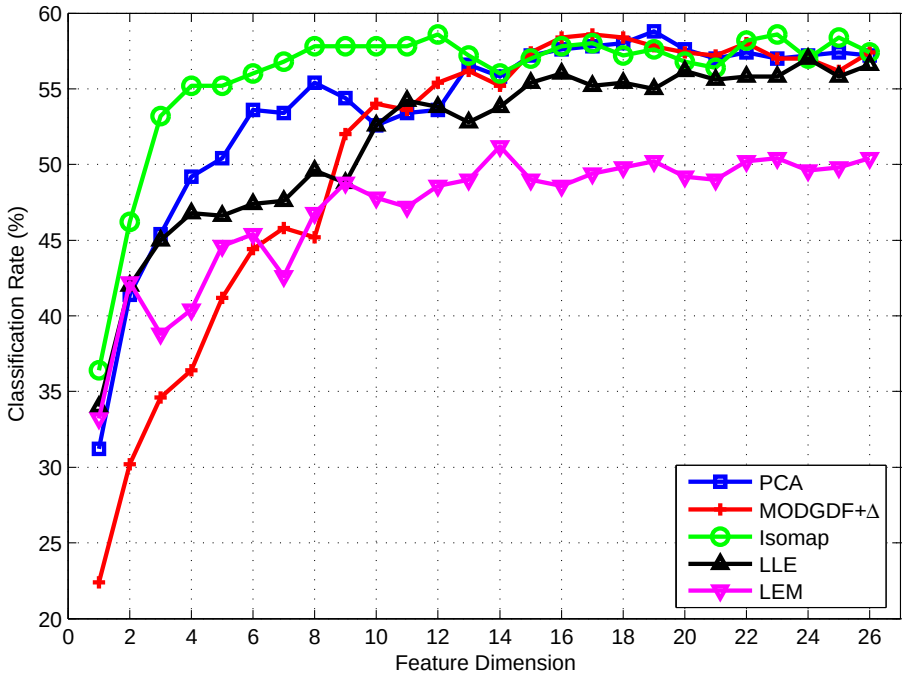
- The MFCC+Δ results are consistent with the the results of the previously conducted MFCC+Δ experiments described in Section 7.4.
- In the vowel classification experiments all of the dimensionality reduction methods yielded higher classification accuracy than the baseline MFCC+Δ and MODGDF+Δ features in low-dimensional spaces.
- Isomap yielded the best performance in the majority of tests. This is consistent with our previous findings. Also, the relative improvement in classification accuracy using Isomap-transformed features compared to the baseline feature types was greatest in low dimensions. This demonstrates Isomap's ability to produce a low-dimensional embedding that retains meaningful information and accurately discovers the underlying structure of the data set. While MFCC+Δ and MODGDF+Δ features increase classification accuracy with growing numbers of dimensions, Isomap, with lower dimensionality, is able to draw the best from both static and dynamic features.
- The largest improvement over the baseline feature vectors was yielded by Isomap in the phone classification task, again displaying the benefits of performing classification in an embedding space produced by the globally motivated manifold learning method.

Reduced dimensionality: MFCC and MODGDF feature combination

As discussed above, with respect to the baseline MFCC and MODGDF results, classification performance can be improved by including information from both the magnitude and phase spectra. This is typically achieved by simply concatenating the MFCC and MODGDF feature vectors (Alsteris and Paliwal, 2005; Hegde et al., 2007a). The improvement gained is demonstrated by the results of the ten vowel classification experiment using joint



(a) MFCC+Δ



(b) MODGDF+Δ

Figure 7.11: Ten vowel classification accuracy using (a) MFCC+Δ and (b) MODGDF+Δ features. The performance of each feature after dimensionality reduction by PCA, Isomap, LLE, and LEM is also shown.

MFCC and MODGDF feature vectors, shown in Figure 7.12(a). A noticeable improvement in the classification rate achieved using the baseline MFCC+MODGDF features can be seen when the dimensionality exceeds $d = 13$; that is, the point at which the MFCC feature vectors are joined to the MODGDF feature vectors. This is also the case when dynamic information is included, as shown in in Figure 7.12(b); however in this case the increase in classification accuracy occurs at $d = 26$.

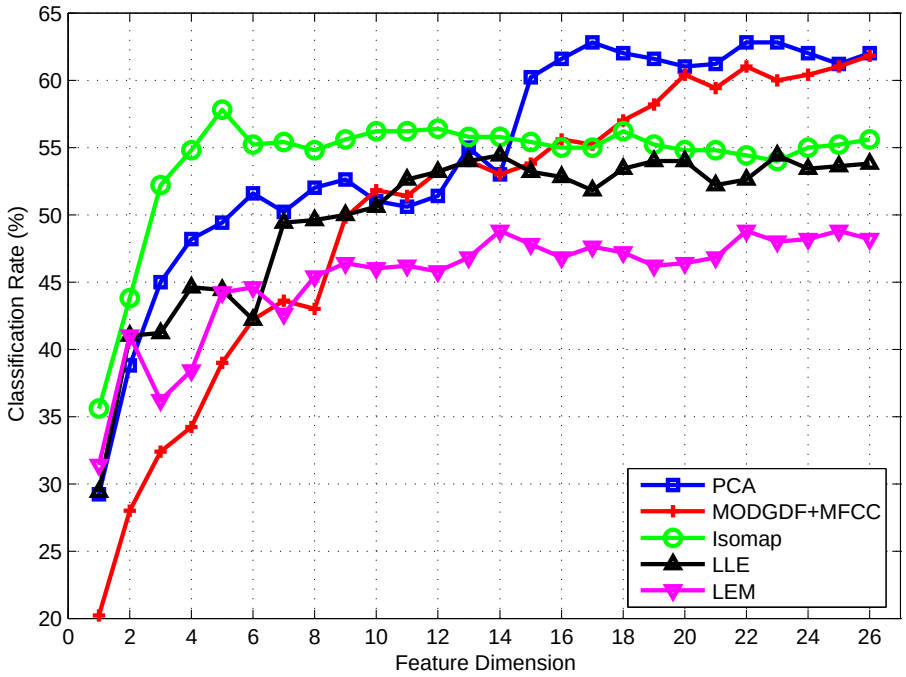
However, this simple feature vector concatenation increases—generally doubles—the dimensionality of the feature vector and hence the computational cost of any subsequent processing of these joint feature vectors. In order to reduce this dimensionality PCA, Isomap, LLE, and Laplacian eigenmaps were each separately applied to the joint feature vectors. Results for the ten vowel classification experiments using joint feature vectors with and without deltas are shown in Figure 7.12; results of the five vowel and phone class classification experiments, available in Appendix B, are consistent with the ten vowel results. A discussion of the results follows:

- The features output from the manifold learning algorithms outperformed the baseline joint features in low dimensions but did not offer performance comparable to the full dimensional joint features.

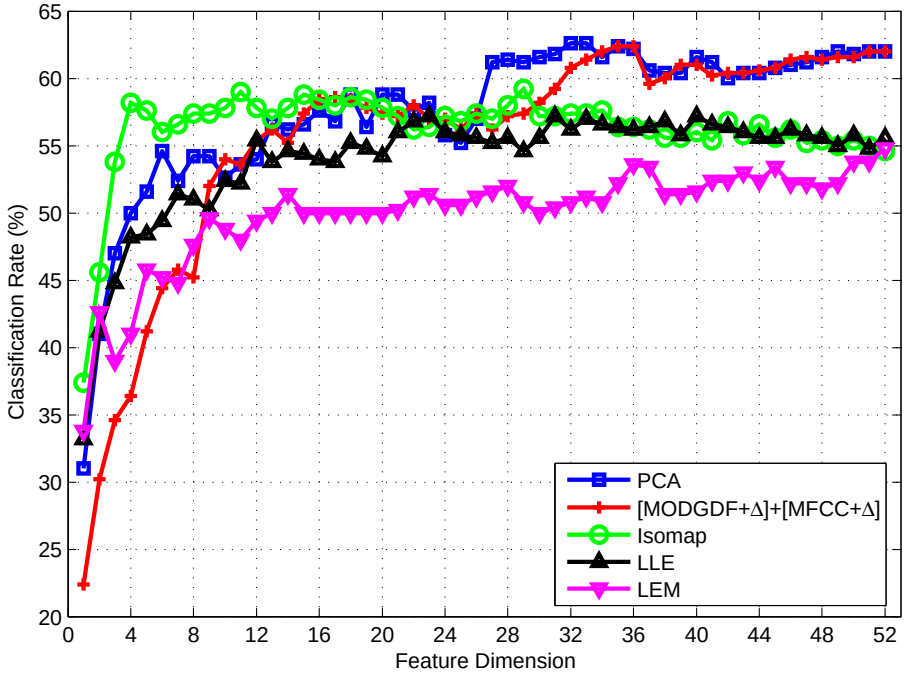
- Of the three manifold learning methods Isomap yielded the best mean classification rate over the three tasks.

- Comparing the classification rates achieved using Isomap and PCA reduced features, PCA outperforms Isomap in 56.41% of the dimensionalities tested. However, examing the first $\frac{D}{2}$ dimensionalities, i.e. $d = 1, \ldots, 13$ for MFCC+MODGDF and $d = 1, \ldots, 26$ for [MODGDF+Δ] + [MFCC+Δ], Isomap yields the best performance in 80.34% of the tests run. Once again, this demonstrates the ability of the Isomap algorithm to retain significant information in low dimensions.

- It is interesting to note that the classification performance of the manifold learning methods did not increase significantly above a feature dimensionality of, approximately, five. However, for the baseline features and PCA-transformed features a marked increase in classification accuracy was achieved above $d = \frac{D}{2}$, that is, the point at which the MODGDF and MFCC feature vectors were concatenated. The



(a) [MODGDF+MFCC]



(b) [MODGDF+Δ]+[MFCC+Δ]

Figure 7.12: Ten vowel classification accuracy using (a) [MODGDF+MFCC] and (b) [MODGDF+Δ]+[MFCC+Δ] features. The performance of each feature after dimensionality reduction by PCA, Isomap, LLE, and LEM is also shown.

Feature set	Dim.	Acc. (Inc.)
PCA(MODGDF+MFCC)	17	62.8 (1)
PCA([MODGDF+Δ] + [MFCC+Δ])	32	62.6 (0.6)

Table 7.5: Ten vowel classification accuracy (%) using joint MFCC and MODGDF features. Feature dimension (Dim.), classification accuracy (Acc.), and increase in accuracy over the full dimensional baseline feature (Inc.) are shown.

Feature set	Dim.	Acc. (Inc.)
PCA(MODGDF+MFCC)	22	76.8 (0)
PCA([MODGDF+Δ] + [MFCC+Δ])	40	78 (1.2)

Table 7.6: Five vowel classification accuracy (%) using joint MFCC and MODGDF features. Feature dimension (Dim.), classification accuracy (Acc.), and increase in accuracy over the full dimensional baseline feature (Inc.) are shown.

inability of the manifold learning algorithms to fully exploit the complementary information in these two feature types may be due to the fact that combining these two incongruous data sets breaks the assumption that there is underlying manifold structure to be discovered. In contrast, the PCA method, which simply preserves variation and makes no assumption of underlying manifold structure, is capable of retaining the complementary phone discriminating information present in the two feature types.

- PCA was found to offer performance increases compared to the original joint features using significantly lower-dimensional features. The improvements resulting from PCA in all three classification tasks are detailed in Tables 7.5–7.7. This demonstrates that PCA can be used as a means to combine the complementary information of MFCC and MODGDF features, without large increases in feature dimensionality.

7.5.4 Conclusions

Both magnitude- and phase-based features were evaluated in phone classification experiments, with the results showing that MFCCs provided better performance than MOD-

Feature set	Dim.	Acc. (Inc.)
PCA(MODGDF+MFCC)	20	82 (0)
PCA([MODGDF+Δ] + [MFCC+Δ])	35	82.14 (.14)

Table 7.7: Phone class classification accuracy (%) using joint MFCC and MODGDF features. Feature dimension (Dim.), classification accuracy (Acc.), and increase in accuracy over the full dimensional baseline feature (Inc.) are shown.

GDFs. The ability of dimensionality reduction methods to exploit low-dimensional structure in both these feature types is also demonstrated in this section. When applied to the MFCC and MODGDF features, Isomap was consistently found to yield a low-dimensional embedding that offered the best phone classification performance in low-dimensional spaces. Thus, suggesting that this low-dimensional structure may be nonlinearly embedded in higher-dimensional space.

Regarding the intrinsic dimensionality of the speech signals examined, an 'elbow' in the classification rate line charts is visible between $d = 2, \ldots, 6$, above this point the addition of further dimensions does not yield a significant increase in classification rate. This indicates that the speech data has an inherent low-dimensional structure, corroborating the conclusions drawn in Section 7.4.4.

Joining MFCC and MODGDF features using simple concatenation was found to increase classification performance, indicating the two feature types contain complementary information. Applying PCA to these concatenated MFCC and MODGDF feature vectors was shown to increase performance without requiring a large increase in dimensionality and the associated increased computational cost. Manifold learning methods were not found capable of achieving this. However, Isomap was found to maintain more discriminatory information in low-dimensional spaces than PCA, when applied to the concatenated features.

7.6 Speaker identification

7.6.1 Introduction

As discussed in Section 5.5.2, in addition to using phone classification as a means of objectively evaluating the low-dimensional representations output by the various dimensionality reduction methods, we also investigate the ability of linear and nonlinear dimensionality reduction to yield low-dimensional features capable of discriminating between speakers. The following subsections detail the speaker identification system used and the resulting performance based on features produced by both linear and nonlinear dimensionality reduction methods.

7.6.2 Experiment setup

The experimental setup we used is based on that described by Reynolds (1995a), who describes a GMM-based speaker identification system and shows this system to be capable of accurate identification of speakers from the TIMIT corpus. This system lacks some of the more recent speaker identification technologies, such as a universal background model.[2] Such a speaker identification system is adequate for this study as we are concerned with examining the ability of linear and nonlinear dimensionality reduction methods to retain speaker discriminatory features, and not concerned with evaluating the performance of a state-of-the-art speaker identification system.

Feature extraction

Twenty speakers, ten male and ten female, were selected from the TIMIT database for use in this study.[3] All ten utterances recorded by each speaker were used.

Each speech signal was first preemphasised, $p = 0.95$, and following this Hamming windowed frames of length 20 ms were extracted with an overlap of 10 ms. Non-speech frames were removed using an energy-based speech activity detector (Reynolds, 1995b). As performed by Reynolds (1995a), twenty MFCCs were computed for each speech frame; the zeroth cepstral coefficient was discarded and the remaining nineteen coefficients retained.

Feature transformation

The 19-dimensional MFCC vectors extracted for each of the chosen speakers, detailed above, made up the high-dimensional input feature matrix. Both PCA and L-Isomap were applied to this matrix and low-dimensional features ranging in dimension from 1–19 produced. The number of nearest neighbours, k, used in L-Isomap was empirically chosen as 16. A total of 300 feature vectors, sampled randomly from the data set, were chosen as landmark points for L-Isomap.

[2] For a review of current state-of-the-art approaches to speaker recognition refer to Campbell et al. (2009).

[3] The speakers chosen were: FAEM0, FAJW0, FALK0, FALR0, FAPB0, FBAS0, FBCG1, FBCH0, FBJL0, FBLV0, MABC0, MADC0, MADD0, MAEB0, MAEO0, MAFM0, MAJP0, MAKB0, MAKR0, and MAPV0.

Choice of L-Isomap

The L-Isomap algorithm was the only manifold learning algorithm used; LLE and Laplacian eigenmaps were not applied in the speaker identification experiments. This was due to the size of the data set. The speaker identification experiments required the embedding of a large number of feature vectors, $N = 61924$. However, the computational demands of the manifold learning methods, discussed in Section 4.3.4, make it impractical to apply them to large data sets. As a result, we employed the L-Isomap algorithm which, as discussed in Section 4.3.2, adapts the standard Isomap algorithm to work on a small subset of so called *landmark* points, thus overcoming the issues raised by the large size of the data set.

It should also be noted that previous experiments, detailed in Sections 7.4 and 7.5, conducted on both synthetic and natural speech show Isomap to produce the most meaningful low-dimensional representations of high-dimensional speech data in a range of tasks—outperforming both LLE and Laplacian eigenmaps in the majority of tests performed. This is further motivation for the use of L-Isomap.

Gaussian mixture model classification

Many forms of pattern classifier have been used for speaker identification in the past. These include dynamic time warping, artificial neural networks, vector quantization, support vector machines and GMMs. The GMM-based approach (Reynolds, 1995a, 2002) is currently the most widely used text-independent speaker identification classifier. Further discussion of GMM-based classification is provided in Appendix A.

A single state GMM consisting of eight mixtures, as used by (Reynolds, 1995a), was trained using the expectation maximisation (EM) algorithm (Dempster et al., 1977) for each of the twenty chosen speakers. In order to classify a test utterance, the likelihoods of the feature vectors extracted from the unknown speaker's utterance are evaluated for each speaker GMM. The unknown speaker is identified as the speaker whose GMM yields the maximum accumulated likelihood.

The GMM speaker identification system was trained and tested using each of the three feature types: baseline MFCC and both PCA- and L-Isomap-transformed MFCCs. The dimensionality of the feature vectors was varied from 1 to 19—the original, full dimen-

sionality.

The system was evaluated with two different amounts of training data: four and eight utterances. The data was divided into four sets each containing the required number of training utterances, chosen randomly, with the remaining utterances used as test data. The GMM speaker identification system was trained and tested on each of these four sets and the total accuracy calculated across all tests.

7.6.3 Results

Speaker identification

The performance of each of the three feature types in the twenty class speaker identification task is shown in Figure 7.13. Eight utterances were provided as training. The percentage of test utterances identified as spoken by the correct speaker is indicated on the vertical axis. The horizontal axis represents the dimensionality of the feature vector. The performance of the baseline 19-dimensional MFCC features is 99.38% which is consistent with previously published results (Reynolds, 1995a).

The results in Figure 7.13 can be summarised as follows:

- L-Isomap yielded the best performance in very low dimensions ($d < 4$) but performance in higher dimensions was inconsistent and generally lower than the other two feature types.
- PCA features offered the highest mean speaker identification accuracy.
- PCA resulted in a maximum accuracy of 100% with as few as seven dimensions.
- The performance of each feature type generally increased proportional to dimensionality, this is to be expected as the higher-dimensional features contain additional information on which to base a classification decision.

The mean accuracies over three dimensionality ranges are presented in Table 7.8. It can be seen that L-Isomap yielded the highest average accuracy in the low-dimensional range but averaged over all dimensions offered the worst speaker identification performance. On average, PCA is shown to have performed better than MFCC and L-Isomap.

Results of the four training utterance experiments are shown in Figure 7.14 and summarised in Table 7.9. The accuracy is shown to decrease relative to the eight utterance

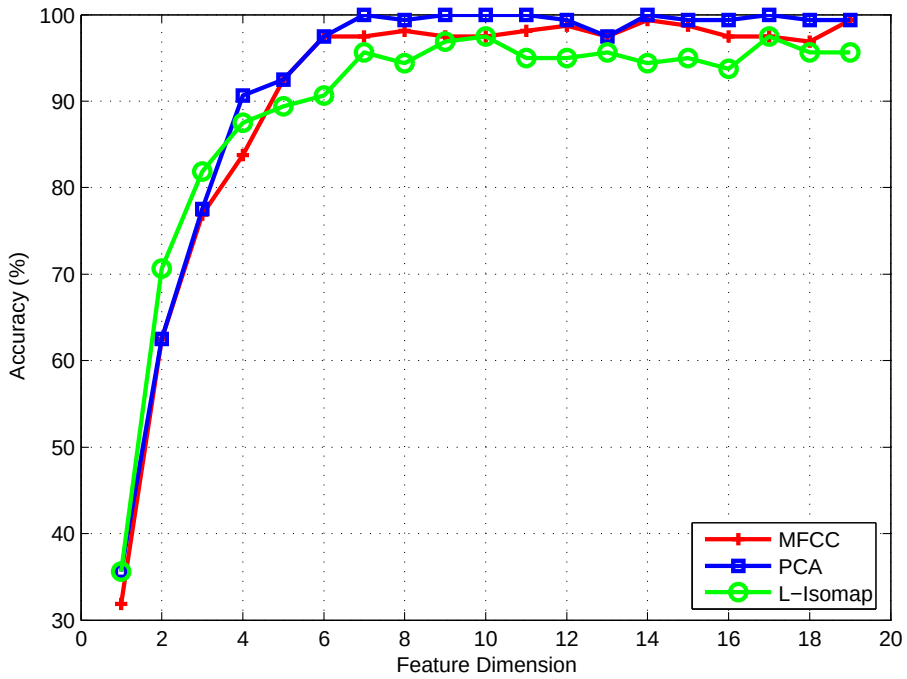


Figure 7.13: Speaker identification accuracy (%) for baseline MFCCs and both PCA- and L-Isomap-transformed features. Eight utterances provided as training.

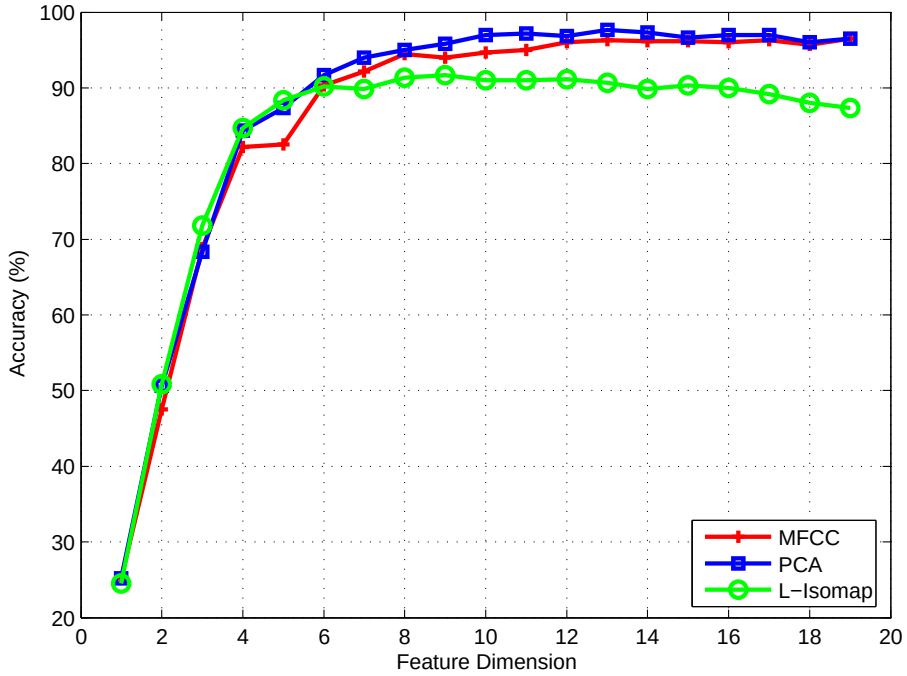


Figure 7.14: Speaker identification accuracy (%) for baseline MFCCs and both PCA- and L-Isomap-transformed features. Four utterances provided as training.

Dimensions	MFCC	PCA	L-Isomap
1–3	57.083	58.542	62.708
4–19	96.758	98.398	94.336
1–19	90.493	92.105	89.342

Table 7.8: Mean speaker identification accuracy (%) for each feature type over three dimensionality ranges. Eight training utterances provided.

Dimensions	MFCC	PCA	L-Isomap
1–3	46.597	48.820	49.306
4–19	93.125	94.831	90.026
1–19	85.779	87.566	83.596

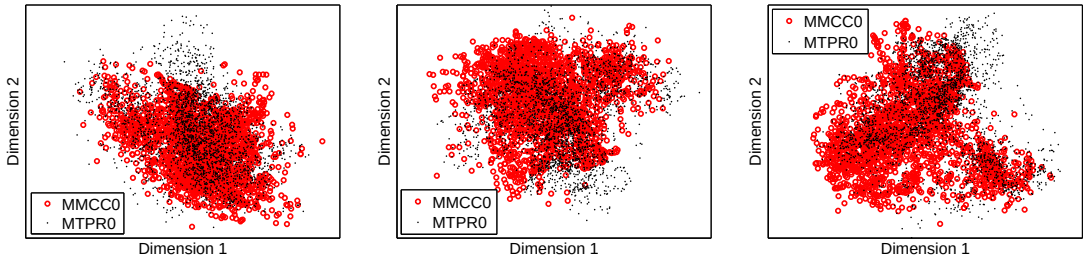
Table 7.9: Mean speaker identification accuracy (%) for each feature type over three dimensionality ranges. Four training utterances provided.

experiment. This is due to the reduced training data. The trends evident in the eight training utterance case, Table 7.8, are supported by results for the system trained on four utterances. However, in this case the L-Isomap algorithm produced only marginal improvements in lower dimensions and identification accuracy can be seen to decrease above 9-dimensional L-Isomap features. Again, this is likely due to the small amount of training data provided.

Visualisation

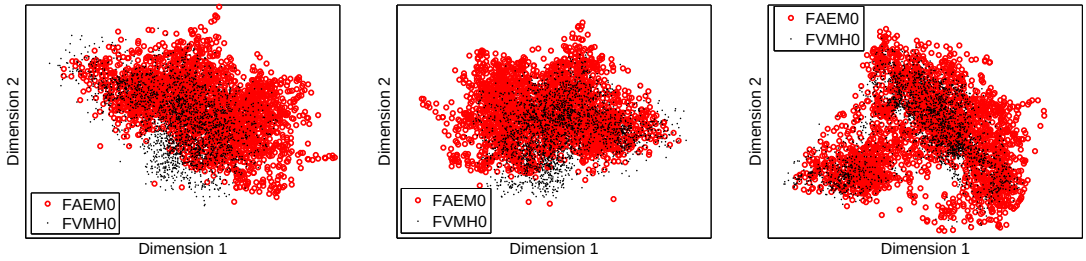
Two-dimensional visualisations of the different low-dimensional feature spaces produced in the classification experiment above are shown in Figures 7.15–7.17. For clarity, data from only two speakers is shown in each figure. Figure 7.15 shows data from two male speakers, Figure 7.16 presents data from two female speakers, and Figure 7.17 illustrates data points from both a male and a female speaker. The particular speakers used in these figures were chosen to be representative of the data set.

Based on visual inspection of this data, it can be seen that there is no clear difference between the feature spaces with regards to separating speakers of the same gender. However, as shown in Figure 7.17, the male and female speakers are visually more clustered and separable in the two-dimensional space produced by L-Isomap than the equivalent PCA and MFCC feature spaces. Thus, these results support the speaker identification results showing that L-Isomap is capable of outperforming PCA and MFCC features in low dimensions.



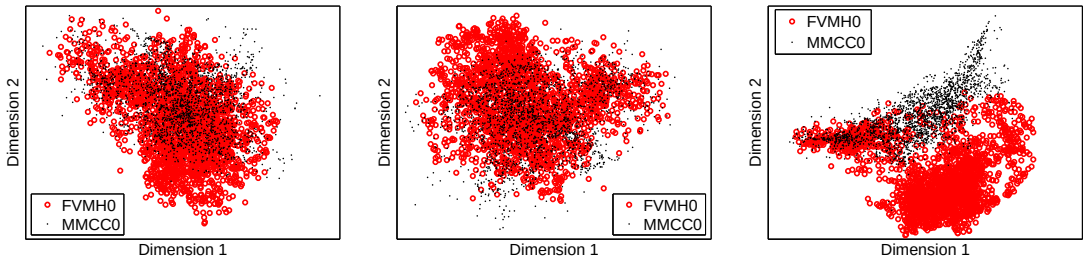
(a) MFCC (b) PCA (c) L-Isomap

Figure 7.15: Two-dimensional representations of all speech frames extracted from two male speakers, MMCC0 and MTPR0, in the TIMIT corpus.



(a) MFCC (b) PCA (c) L-Isomap

Figure 7.16: Two-dimensional representations of all speech frames extracted from two female speakers, FAEM0 and FVMH0, in the TIMIT corpus.



(a) MFCC (b) PCA (c) L-Isomap

Figure 7.17: Two-dimensional representations of all speech frames extracted from both a male and female speaker, MMCC0 and FVMH0, in the TIMIT corpus.

7.6.4 Conclusions

This study applied both PCA and L-Isomap to conventional MFCCs and evaluated the resulting features in GMM-based speaker identification experiments. L-Isomap was found to offer the best speaker identification accuracy for low-dimensional features which indicates that L-Isomap may be useful for two- and three-dimensional visualisation of speaker data. L-Isomap's ability to achieve better speaker separation than PCA in these low dimensions may be due to L-Isomap exploiting nonlinear relationships which PCA is incapable of finding. This reinforces the proposal that speech data lies on a low-dimensional manifold nonlinearly embedded in acoustic space. It also shows that as few as three features can provide a significant amount of information for speaker identification.

However, results indicate that for higher dimensions L-Isomap-transformed features are not as useful as conventional MFCCs or PCA-transformed features. Thus, linear dimensionality reduction is found to be more useful than manifold learning for the speaker identification system described in this study. This is in contrast to the previous phone classification experiments which found Isomap capable of outperforming linear methods. One possible cause of the poor performance of the Isomap features, relative to the MFCC and PCA-transformed features, may be the increased phonetic variability in the data set used in this study.

PCA-transformed features offered the highest speaker identification accuracy, greater than that of baseline MFCCs, using as few as seven dimensions. PCA retained information relevant to speaker identification while reducing redundant information. Removal of this redundant information aids GMM classification and also reduces the computational demands on the speaker identification system.

Chapter 8

Conclusions and Future Work

The aim of the work presented in this dissertation is to evaluate the performance of a number of linear and nonlinear dimensionality reduction methods when applied to speech data and to examine the possibility that speech has an inherent nonlinear low-dimensional manifold structure. To accomplish this, three manifold learning methods and one classic linear dimensionality reduction method have been applied in a number of speech processing experiments involving both synthetic and natural speech recordings. In this chapter the work presented in this dissertation is summarised, the main conclusions of this work are presented, and suggestions for future work are proposed.

8.1 Summary

In Chapter 3 a large number of previous studies of the underlying dimensionality of speech were reviewed. These studies differ in motivation and the techniques used, however when surveyed as a body of work these studies present compelling evidence that speech data is inherently low-dimensional. A number of these studies go further, proposing that this low-dimensional structure is *nonlinearly* embedded in high-dimensional space[1].

The aim of this dissertation is to investigate this proposal by applying a number of recently proposed manifold learning algorithms to speech data in order to determine if these algorithms can find inherent low-dimensional structure, and to evaluate the possible benefits of using these low-dimensional features in speech processing tasks. Chapter 4

[1] For an example of such an embedding refer to Figure 1.1 which depicts a two-dimensional structure nonlinearly embedded in three-dimensional space.

describes the manifold learning algorithms—Isomap, LLE, and Laplacian eigenmaps—and the classic, linear, PCA algorithm which is used for the purposes of comparison. The abilities of these algorithms are then demonstrated on a number of non-speech data sets. Previous exploratory applications of manifold learning methods to speech are also described. This dissertation aims to further this previous work by thoroughly evaluating the algorithms in a variety of previously untested speech processing applications.

This evaluation was achieved using the framework proposed in Chapter 5 which outlines a methodology for applying dimensionality reduction methods to chosen sets of speech signals and evaluating the performance of each method. Two contrasting types of feature were used in this framework: MFCCs, derived from the magnitude spectrum; and MODGDFs, computed from the phase spectrum. Also, two different types of evaluation procedure were used: visualisation and classification. The visualisation procedure is relatively straightforward, involving a visual inspection of the two- and three-dimensional spaces output by the dimensionality reduction methods. Evaluation based on classification offers a more objective means of measuring the performance of the various methods. Procedures for both phone classification and speaker identification are described in Section 5.5.2. A number of different classifiers were tested to determine an appropriate classifier to use for phone classification. The SVM classifier with RBF kernel (5.20) was found to outperform all other classifiers in terms of mean classification rate in phone classification tasks. As a result, the SVM classifier with RBF kernel was used in all phone classification experiments conducted in this dissertation. A traditional GMM-based classification system was used for the speaker identification evaluations.

In Chapter 6 the manifold learning and linear dimensionality reduction methods were applied to synthetic speech. The motivation for using synthetic speech data was the ability to know and control its characteristics. The synthetic speech signals were generated using an LP-modelled filter excited with an LF-modelled glottal pulse train. MFCC feature vectors were computed for all the synthetic signals using the framework described in Chapter 5. All of the dimensionality reduction methods were shown to be capable of facilitating the visualisation of formant and f0 variation. To provide clear visualisations of f0 variation, three-dimensional embeddings—rather than two-dimensional as was sufficient to retain information relating to formant variation—were required. The variation in f0

was also found to be less well defined and separated in the low-dimensional visualisation spaces than the formant variation. This demonstrates that more dimensions are required to accurately retain information relating to pitch variation than formant variation.

In addition to illustrating the ability of the dimensionality reduction methods to find low-dimensional structure in synthetic speech, the methods were also tested in vowel classification tasks. The purpose of these tasks was twofold: first, to evaluate the amount of discriminatory information that was retained by each of the dimensionality reduction methods; second, to examine the amount of meaningful information retained in speech feature representations of varying dimensionality. The results of these vowel classification tasks showed the ability of the dimensionality reduction methods to retain information characterising individual vowels in low-dimensional space. Also, the first four feature dimensions were found to contain almost all the discriminatory information, with dimensions five and above proving insignificant with respect to classification accuracy. This shows that the synthetic vowel sounds have an inherent dimensionality of four, supporting the hypothesis that speech sounds have underlying low-dimensional structure. This finding is in agreement with previously published results, as reviewed in Chapter 3. A comparison of the performance of the dimensionality reduction methods found that the Isomap algorithm performed best in 51.92% of the classification tests run. The performance of the other manifold learning methods, LLE and Laplacian eigenmaps, was inconsistent. However, the manifold learning methods did offer improved classification performance over MFCCs and PCA-transformed features in experiments on highly noise-corrupted synthetic speech.

Motivated by the results of the synthetic speech experiments, Chapter 7 presents a wider range of studies performed on natural speech from the TIMIT corpus. First, the synthetic f0 visualisation experiments of Chapter 6 were extended. High-dimensional data sets consisting of MFCC feature vectors extracted from five vowels—/ɑ/, /i/, /ɛ/, /æ/, and /o/—were reduced to three dimensions using each of the dimensionality reduction methods. Upon inspection, f0 variation was clearly visible in the embedding spaces. However, the f0 structure was less well represented than was the case for the synthetic speech, this was likely due to the increased formant and speaker variation in the natural speech corpus. Notably, the linear method, PCA, produced the worst visualisations of all four methods for two of the vowels, /æ/ and /o/, suggesting that the manifold learning methods

are more suitable as a tool to visualise f0 variation in speech data.

A further visualisation experiment was then performed in which MFCC features extracted from units of five different vowels were reduced to just two dimensions. Individual vowel clusters were visible in the resulting two-dimensional spaces and a close correspondence to traditional formant space and articulatory space was found. Visual inspection revealed that the spaces output by PCA and Isomap corresponded more closely to articulatory space than those output by LLE and Laplacian eigenmaps. This is likely due to the differing motivations of the methods, with PCA and Isomap preserving the global geometry of the manifold, while LLE and Laplacian eigenmaps preserve local geometries on the manifold. This illustrates the importance of retaining global geometric structure.

A range of phone classification experiments were then performed on MFCC feature vectors both with and without dynamic information. All of the dimensionality reduction methods, with the exception of Laplacian eigenmaps, consistently produced low-dimensional features that outperformed the equivalent MFCC features. Once again, illustrating that these methods retain information capable of discriminating between phones. Isomap- and LLE-derived features achieved higher classification accuracy than the baseline MFCC and PCA-transformed features in low dimensions, indicating that these manifold learning algorithms are more capable of retaining information required to discriminate between phones, especially in low-dimensional space, when compared to the linear method. This demonstrates the ability of these methods to exploit nonlinear structure in the speech space. In general, Isomap was found to yield superior performance to the other dimensionality reduction methods. This indicates that preserving global structure rather than local relationships is more important for speech feature transformation.

Next, in Section 7.5, these phone classification experiments were repeated using features derived from the magnitude spectrum, MFCCs, and features derived from the phase spectrum, MODGDFs. Results showed that MFCCs provided better performance than MODGDFs. The dimensionality reduction methods were shown to exploit low-dimensional structure in both these feature types. When applied to MFCC and MODGDF features, Isomap consistently yielded a low-dimensional embedding that offered the best phone classification performance in low-dimensional spaces. Joining MFCC and MODGDF features was found to increase classification performance, indicating the two feature types con-

tain complementary information. Applying PCA to these joint MFCC and MODGDF feature vectors was shown to increase performance without requiring a large increase in dimensionality and the associated increased computational cost. Manifold learning methods were not found capable of achieving this. However, Isomap was found to maintain more discriminatory information in low-dimensional spaces than PCA, when applied to the concatenated features.

Results from both phone classification studies yielded information relating to the intrinsic low-dimensional structure of speech. 'Elbows' in the classification rate vs. dimension plots indicate that dimensions above $d = 2, \ldots, 6$ contain little discriminatory information. In the majority of cases the 'elbow' was observed at or below $d = 4$. Also, the fact that Isomap-transformed features performed well and the 'elbow' was most prominent in the classification rates achieved using these features suggests that this low-dimensional structure is nonlinear.

The final study conducted on natural speech is reported in Section 7.6. This study applied both PCA and L-Isomap as feature transformation front-ends in a speaker identification system. L-Isomap was found to offer the best speaker identification accuracy for low-dimensional features. This shows that L-Isomap is useful for two- and three-dimensional visualisation of speaker data. Results also shows that as few as three features can provide a significant amount of information for speaker identification. These two findings reinforce the proposal that speech data lies on a low-dimensional manifold nonlinearly embedded in acoustic space. However, results indicated that higher-dimensional L-Isomap-transformed features are not as useful as conventional MFCCs or PCA-transformed features. This is in contrast to our previous experiments that showed Isomap capable of outperforming linear methods. One possible cause of the poor performance of the Isomap features, relative to the MFCC and PCA-transformed features, may be the increased phonetic variability in the data set used in this study.

8.2 Overall conclusions

The manifold learning methods were shown to be capable of producing meaningful low-dimensional representations of speech data suggesting speech has low-dimensional manifold structure. In general, these methods were found to outperform PCA in low dimensions,

indicating that speech may lie on a manifold nonlinearly embedded in high-dimensional space. Examining the various low-dimensional embeddings produced, the speech analysed was found to have an inherent dimensionality varying between two and six. The geodesically-motivated Isomap algorithm was found to consistently outperform the locally-motivated LLE and Laplacian eigenmaps algorithms, suggesting that methods that aim to preserve global structure—rather than local structure—are most appropriate for speech tasks.

Phone classification experiments showed that Isomap can offer improvements over standard features and PCA-transformed features. Investigation of features derived from the magnitude spectrum and phase spectrum found both to have similar low-dimensional structure and confirm that the phase spectrum contains useful information for phone discrimination. Results indicated that combining magnitude and phase spectrum information yields improvements in phone classification tasks. A method applying PCA to combine features derived from the magnitude spectrum with features derived from the phase spectrum for increased phone classification accuracy, without large increases in feature dimensionality, was also described.

In addition to phone classification experiments, PCA and L-Isomap were evaluated as feature transformation front-ends for speaker identification. L-Isomap was found to offer the best performance in low dimensions but results showed that higher-dimensional L-Isomap-transformed features did not perform as well as conventional MFCCs or PCA-transformed features.

8.3 Future work

Much work remains to be done in studying how the manifold learning approaches might improve existing speech processing applications or inspire new approaches to existing problems. The contributions made by this dissertation are intended to motivate future research in this area by demonstrating the underlying low-dimensional structure of speech and showing that manifold learning methods, particularly Isomap, can yield improvements in several speech problems. However, a number of practical difficulties exist in applying these methods in a wide range of speech processing algorithms. Hence these difficulties warrant further investigation.

One such difficulty is the question of how to choose the number of nearest neighbours k to be used in the manifold learning algorithms. The sensitivity of these methods to k has been shown in previous studies (Balasubramanian et al., 2002) and is discussed in Section 5.4.2. In this work an empirical approach was used in which a range of k values were exhaustively tested to select a suitable value. Such an approach is appropriate for an investigation of manifold learning algorithm performance, as conducted in this dissertation, however it would cause difficulties in many practical applications where it is not possible to perform this exhaustive testing; for example, due to time constraints. A number of approaches to select the optimal k value have been proposed (Kouropteva et al., 2002; Samko et al., 2006; Shao et al., 2007; Shao, 2008). It would be interesting to investigate these approaches with respect to applying manifold learning methods to speech.

In the classification experiments performed in this dissertation both the training and testing data were combined into a single data set, and the dimensionality of this data set was reduced using each of the dimensionality reduction methods. This approach was required as the manifold learning algorithms used lack a means to map new high-dimensional data to a previously created low-dimensional embedding space. Naturally, this approach is not practical in real ASR applications where the training data and testing data are encountered separately. Hence, another issue worthy of investigation is the use of out-of-sample extensions (Bengio et al., 2004; Law and Jain, 2006) to the manifold learning algorithms in speech applications. These extensions enable the projection of new high-dimensional data points into an existing low-dimensional space. Such investigations could facilitate the use of manifold learning algorithms as feature transformation front-ends in large-scale speech processing applications such as ASR and speaker recognition. Given the findings of this work it would be particularly interesting to apply the Isomap algorithm in this way.

The investigations described in this work focus on two high-dimensional feature representations: MFCCs and MODGDFs. The implications of choosing one high-dimensional representation over another are worthy of further study. For example, how does the low-dimensional structure of various features—for example: Fourier spectrum, LPC coefficients, line spectral frequencies, etc.—differ?

Finally, one interesting possibility worthy of further research is the prospect of synthesising speech from the data points on the low-dimensional speech manifold. This is

motivated by the view, as supported by the findings of this work, that speech intrinsically lies on a low-dimensional manifold. If one mapped data from this low-dimensional space into the original high-dimensional feature space, one could then produce a synthetic speech signal from the high-dimensional feature vector—assuming an appropriate feature representation, e.g. LPC, is used. For example, consider the three-dimensional 'pitch-space' shown in Figure 6.5. If one chose an arbitrary point in this three-dimensional space, mapped it into high-dimensional feature space, and then synthesised the equivalent speech signal it would theoretically have an f0 value determined by its position in the low-dimensional 'pitch space'. There are many possible applications of such a synthesis system. However, this requires a means of mapping from embedding space to the original high-dimensional feature space. Manifold learning methods generally do not facilitate such a mapping. Possible approaches to overcome this problem have been described in previous studies of multi-pose face synthesis (Wang et al., 2003; Zhang et al., 2004) which may also be applicable to speech.

Appendix A

Classifiers

This appendix provides an overview of the two types of classifiers principally used in this dissertation: Support vector machines and Gaussian mixture models.

A.1 SVM

Support vector machines are a set of recently developed methods that can be used for both classification and regression. This section presents a brief introduction to the basic theory behind SVM classification to provide the reader with some insight into the classification algorithm used for the majority of the classification experiments presented in this work. A more complete description of SVMs is provided by Vapnik (1995), and Schölkopf and Smola (2002).

Consider the two-class classification problem, with a set of training vectors and corresponding class labels,

$$\mathcal{X} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}, \quad \mathbf{x} \in \mathbb{R}^{\mathbb{D}}, \ y \in \{-1, 1\} \ , \tag{A.1}$$

where y_i indicates the class of the vector $\mathbf{x}_i$. The goal in classification is to separate the two classes. SVMs aim to construct a hyperplane that optimally separates the classes. Any hyperplane can be stated as,

$$(\mathbf{w} \cdot \mathbf{x}) + b = 0, \quad \mathbf{w} \in \mathbb{R}^D, b \in \mathbb{R} \ , \tag{A.2}$$

where $\mathbf{w}$ is a normal vector, orthogonal to the hyperplane, and b is the offset of the hyperplane from the origin. Figure A.1 depicts a hyperplane (solid line) separating two classes of data. Ideally, we would choose values of $\mathbf{w}$ and b such that the margin between the separating hyperplane and each class of data is as large as possible. This is referred to as a maximum margin hyperplane. The hyperplane shown in Figure A.1 is maximum margin, and the corresponding margins (dashed lines) are also shown. These 'margin' hyperplanes are given by,

$$(\mathbf{w} \cdot \mathbf{x}) + b = 1 \ , \tag{A.3}$$

and

$$(\mathbf{w} \cdot \mathbf{x}) + b = -1 \ . \tag{A.4}$$

The distance between these two hyperplanes, the margin, is $\frac{2}{||\mathbf{w}||}$. Thus, to choose a maximum margin $||\mathbf{w}||$ must be minimised. Further, a set of constraints are added to prevent data points appearing within the margin,

$$(\mathbf{w} \cdot \mathbf{x}) + b \geq +1 \quad \text{for} \ \ y_i = +1 \ , \tag{A.5}$$

and

$$(\mathbf{w} \cdot \mathbf{x}) + b \leq -1 \quad \text{for} \ \ y_i = -1 \ . \tag{A.6}$$

This can be written as a single constraint,

$$y_i(\mathbf{w} \cdot \mathbf{x} + b) - 1 \geq 0 \quad \forall \ \ i \ . \tag{A.7}$$

Thus, the problem of choosing $\mathbf{w}$ and b to maximise the margin can be written as a convex optimization problem: minimise $||\mathbf{w}||$ subject to the constraint of Equation (A.7). This problem can be solved using standard quadratic programming methods. The full details of this solution are beyond the scope of this appendix. However, it is worth noting that the solution of this problem has an expansion,

$$\mathbf{w} = \sum_i v_i \mathbf{x}_i \ , \tag{A.8}$$

in terms of a limited number of training data points, $\mathbf{x}_i$, that lie on the margin, as shown

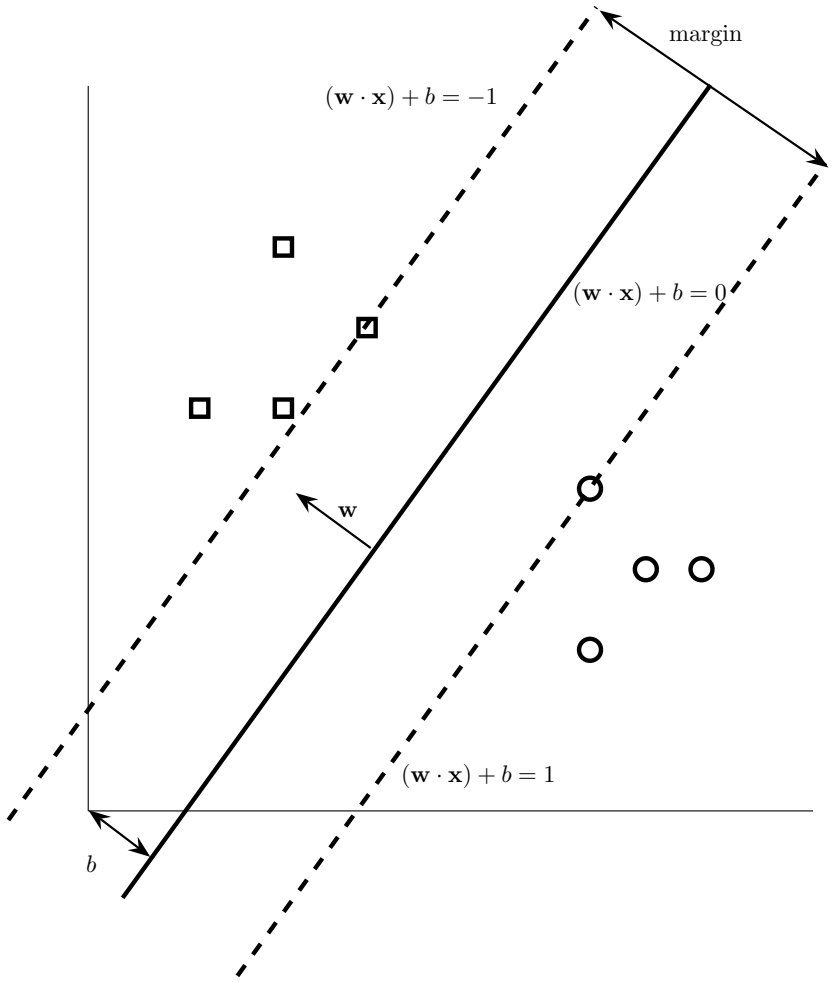
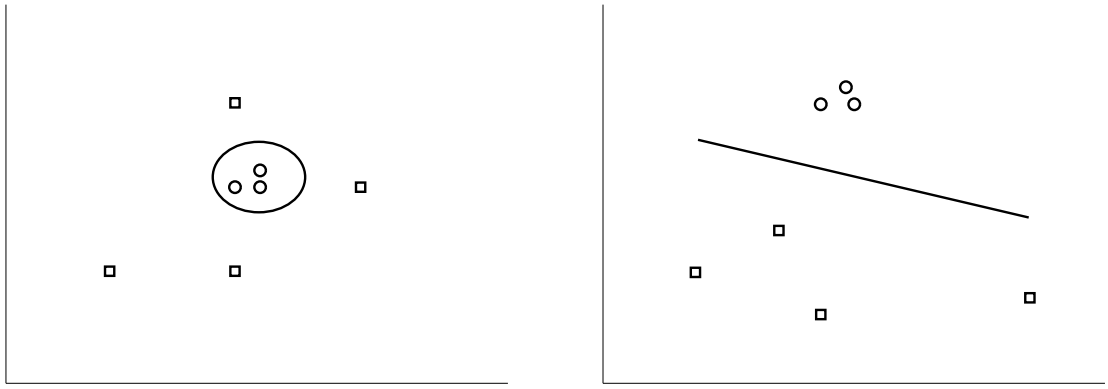


Figure A.1: SVMs learn a maximum margin hyperplane that best separates two-classes. Circles have a label $y_i = +1$ while squares have a label $y_i = -1$. Data points on the margin (dashed lines) are support vectors.

in Figure A.1. These training data points are the 'support vectors' that give SVMs their name. The support vectors contain all the information necessary to perform classification.

The maximum margin hyperplane discussed above is a linear classifier. This algorithm has been extended to facilitate nonlinear classification. The basic idea is that classification may be easier in some higher-dimensional feature space. Thus it is desirable to construct the maximum margin hyperplane in this high-dimensional space. However, computation of the inner products in the above algorithm may be computationally expensive in a high-dimensional space. This problem can be avoided by using the so-called kernel trick (Aizerman et al., 1964). The kernel trick maps the input data points into a higher-dimensional space in which a linear classifier can be used. This is equivalent to performing



(a) Input space (b) Feature space

Figure A.2: SVMs nonlinearly map the training data into a higher-dimensional feature space in which a maximum margin hyperplane can be constructed.

nonlinear classification in the input space, as illustrated in Figure A.2. This kernel trick is accomplished by replacing the dot products in the linear maximum margin hyperplane algorithm above with a kernel function. Many kernel functions have been proposed and used successful with SVMs. The kernel functions used in this work are listed in Section 5.5.2.

A.2 GMM

In Gaussian mixture model (Reynolds, 1995b) classification the observed variables are modelled using a combination of Gaussian probability density functions (pdf), known as a Gaussian mixture model. Each class of data to be classified is represented by a different, multidimensional, Gaussian mixture model. For example, in speaker recognition, the observed variables are some parametrised version of a speakers speech—for example, MFCC vectors—and each speaker is modelled by a different Gaussian mixture model.

A Gaussian mixture model is formed by a weighted linear combination of Gaussian pdfs. The Gaussian pdf of a D-dimensional feature vector $\mathbf{x}$ for the mth mixture, also called a state, is given by

$$b_i(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{D}{2}} |\mathbf{C}_i|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}-\mathbf{u}_i)'\mathbf{C}_i^{-1}(\mathbf{x}-\mathbf{u}_i)} \quad . \tag{A.9}$$

The probability of an observed variable belonging to a particular Gaussian mixture model may then be defined as

$$p(\mathbf{x}|\lambda) = \sum_{i=1}^{M} p_i b_i(\mathbf{x}) \quad , \tag{A.10}$$

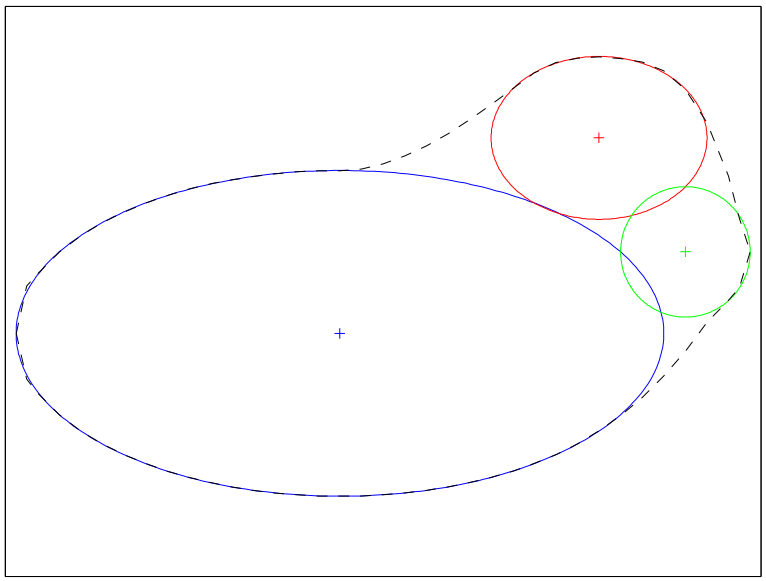


Figure A.3: Gaussian mixture model, dashed line, shown as a combination of Gaussian pdfs.

where p_i are the mixture weights, $b_i(\mathbf{x})$ are the component Gaussian pdfs of which there are M in total. To normalise $p(\mathbf{x})$ the weights must sum to 1

$$\sum_{i=1}^{M} p_i = 1 \ , \tag{A.11}$$

this assures the Gaussian mixture density integrates to 1 and is thus a true pdf. The parameter vector λ represents the GMM mean, covariance, and weight parameters, i.e.

$$\lambda = \{p_i, \mathbf{u}_i, \mathbf{C}_i\} \ . \tag{A.12}$$

The GMM can form an approximation of any pdf, given an appropriate number of mixture components. An example of a GMM is shown in Figure A.3.

The task of training a GMM classifier amounts to estimating the GMM model parameters λ. This is achieved by maximising $p(\mathbf{X}|\lambda)$ with respect to λ, where $\mathbf{X}$ is a matrix containing all observations from a particular class of data—for example, speech feature vectors from a particular speaker. This maximisation is performed using the EM algorithm (Dempster et al., 1977).

The trained GMM classifier can then be used to assign new observed variables to a particular class by computing the probability of each GMM given an observed variable

and selecting the GMM with the highest probability. This is a form of maximum a posteriori (MAP) classification. Using Bayes theorem, The probability of a GMM given an observation is given as

$$P(\lambda_j|\mathbf{x}_i) = \frac{p(\mathbf{x}_i|\lambda_j)P(\lambda_j)}{P(\mathbf{x}_i} \quad . \tag{A.13}$$

The denominator, $p(\mathbf{x}_i)$, may be ignored as it is a constant. Also, the a priori probability of an observation belonging to a particular class $p(\lambda_j)$ is assumed to be the same for each class so this can be ignored. Thus, the classification problem reduces to finding the GMM, λ, that maximises $p(\mathbf{x}_i|\lambda_j)$. This is achieved by computing the probability of the observed variable given each of the trained GMMs, as given in Equation (A.10), and selecting the GMM that yields the highest probability.

If there is more than one observed variable—for example, a set of speech feature vectors—this computation must be performed for all observed variables. This is typically achieved by simply assuming all observations are independent:

$$p(\{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{N-1}\}|\lambda_j) = \prod_{n}^{N-1} p(\mathbf{x}_n|\lambda_j) \quad . \tag{A.14}$$

Appendix B

Further Classification Results

This appendix presents further classification results of the experiments described in Chapter 7.

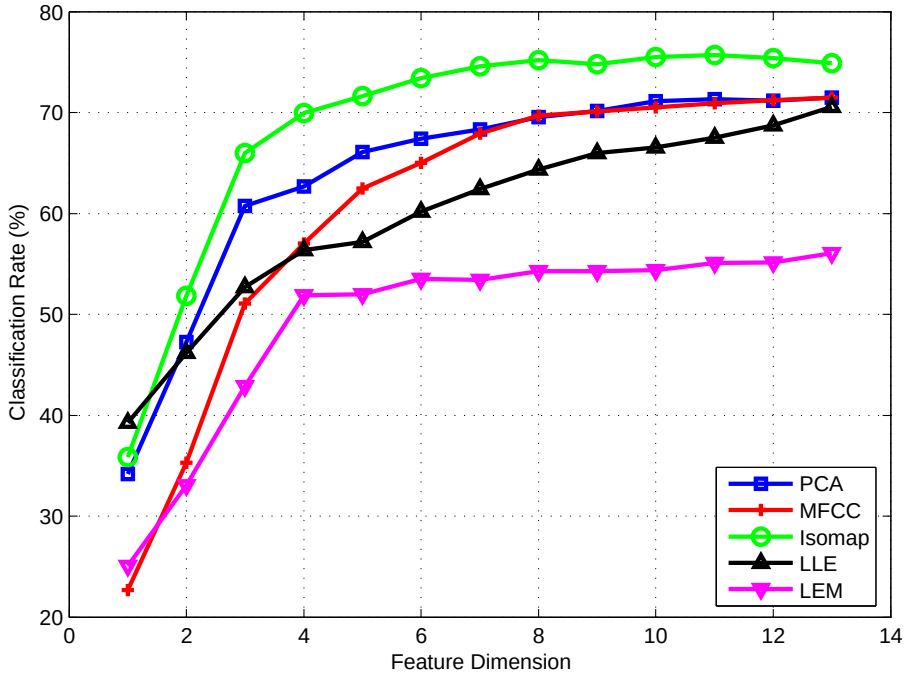


Figure B.1: Ten vowel classification results for baseline MFCC, PCA, Isomap, LLE and Laplacian eigenmaps features on data from the TIMIT speech corpus. Evaluation performed on training data.

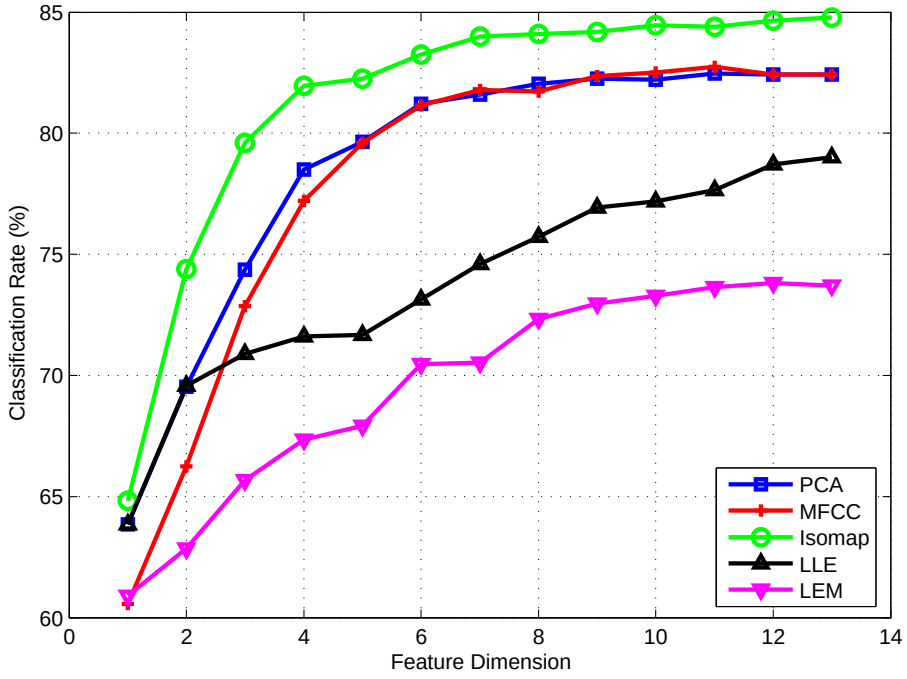


Figure B.2: Phone class classification results for baseline MFCC, PCA, Isomap, LLE and Laplacian eigenmaps features on data from the TIMIT speech corpus. Evaluation performed on training data.

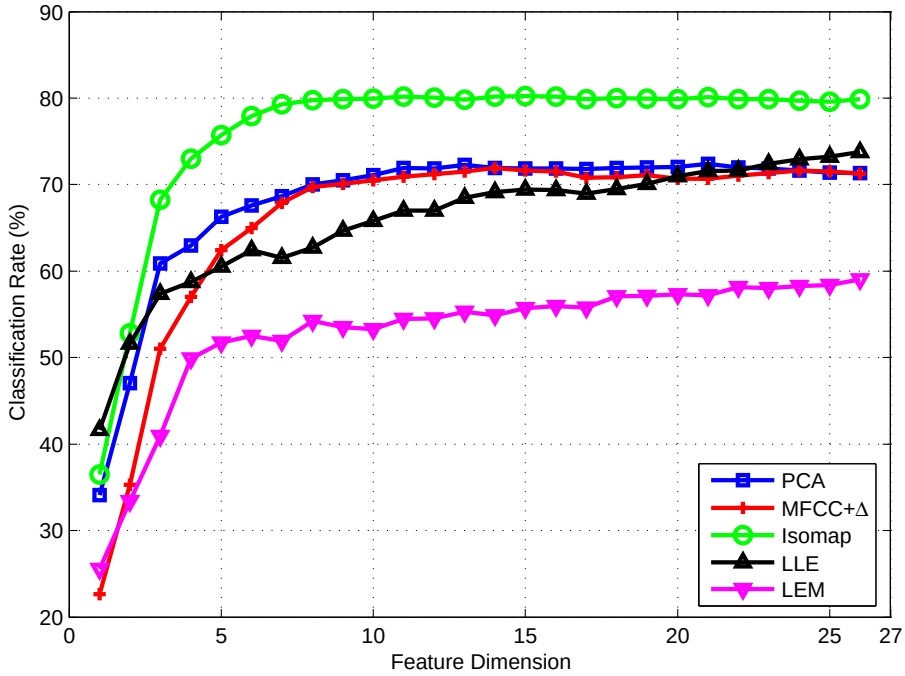


Figure B.3: Ten vowel classification results for baseline MFCC+Δ, PCA, Isomap, LLE and Laplacian eigenmaps features on data from the TIMIT speech corpus. Evaluation performed on training data.

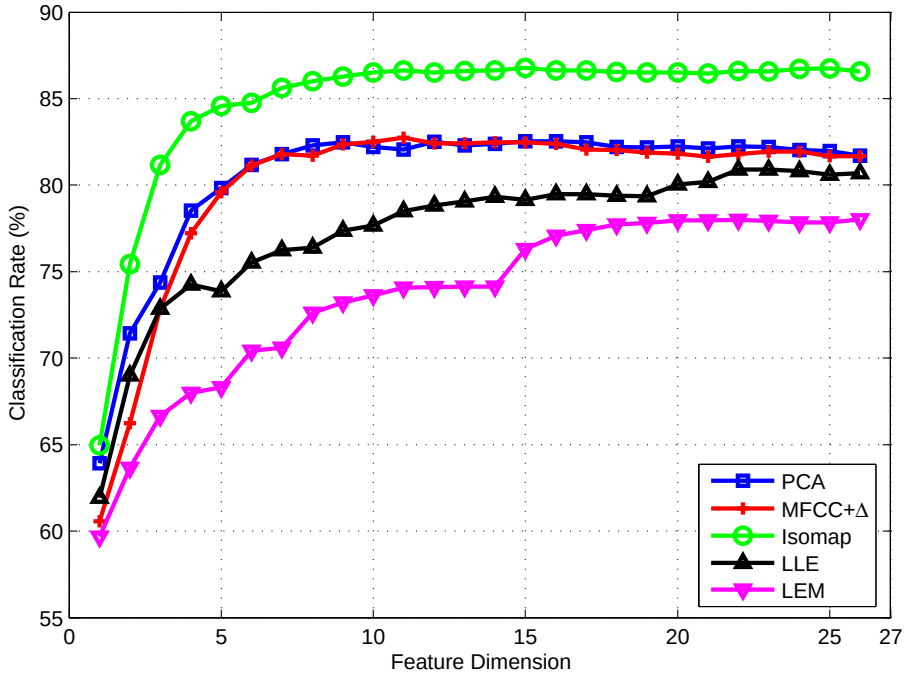
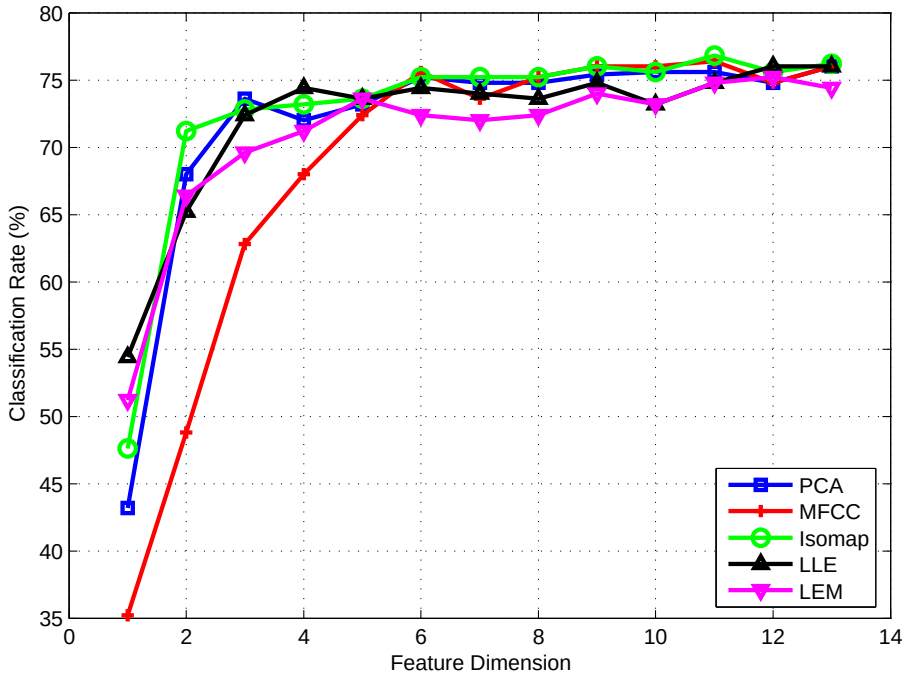
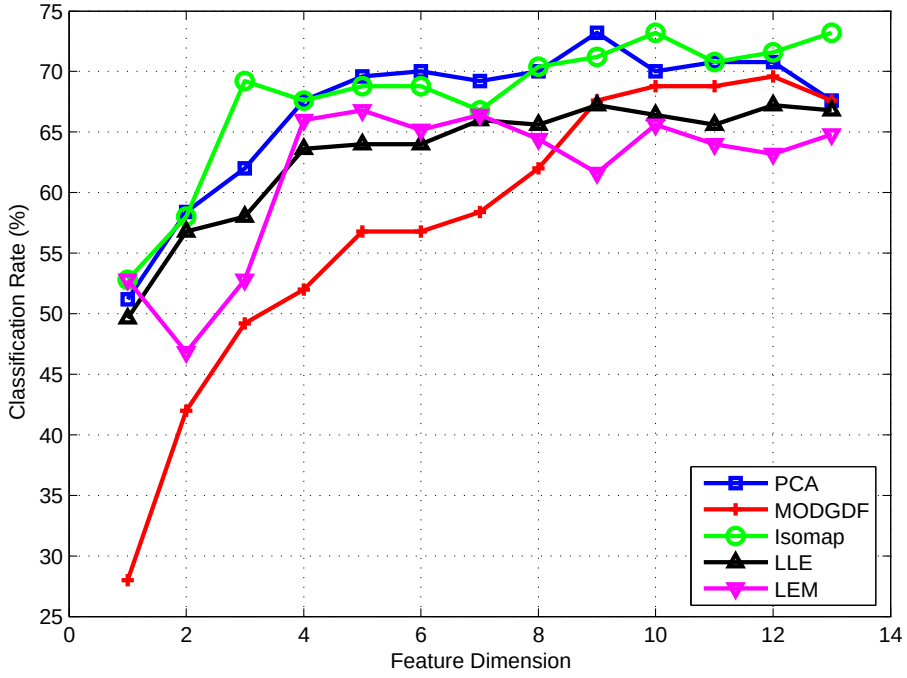


Figure B.4: Phone class classification results for baseline MFCC+Δ, PCA, Isomap, LLE and Laplacian eigenmaps features on data from the TIMIT speech corpus. Evaluation performed on training data.

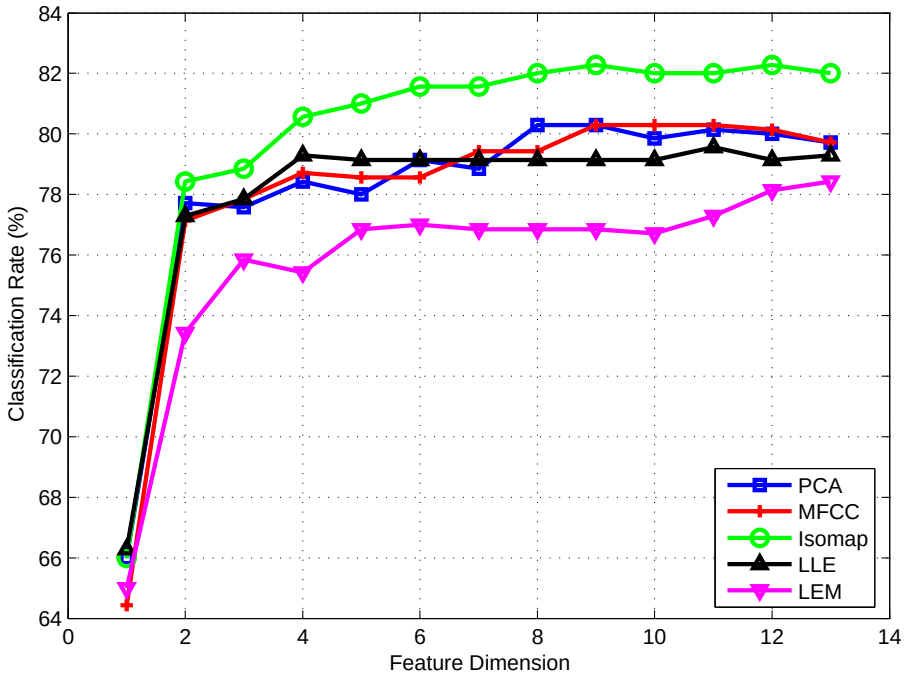


(a) MFCC

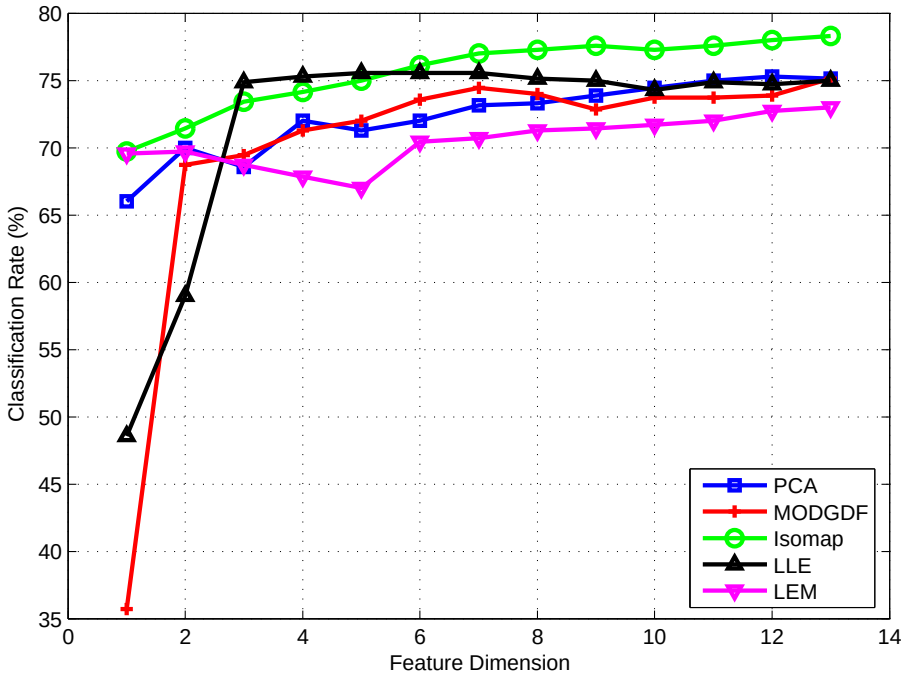


(b) MODGDF

Figure B.5: Five vowel classification accuracy using (a) MFCC and (b) MODGDF features. The performance of each feature after dimensionality reduction by PCA, Isomap, LLE, and LEM is also shown.

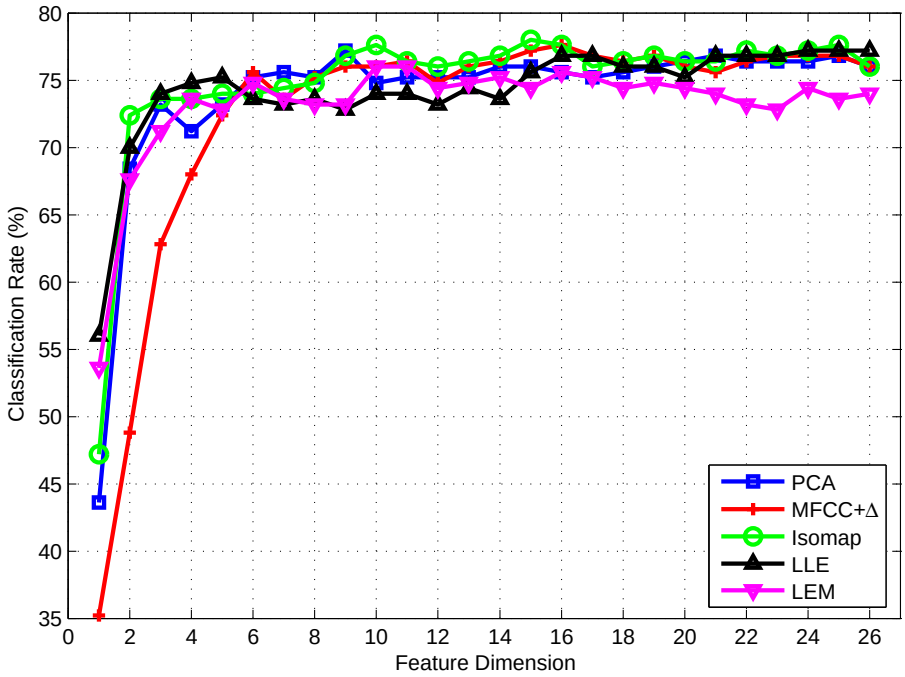


(a) MFCC

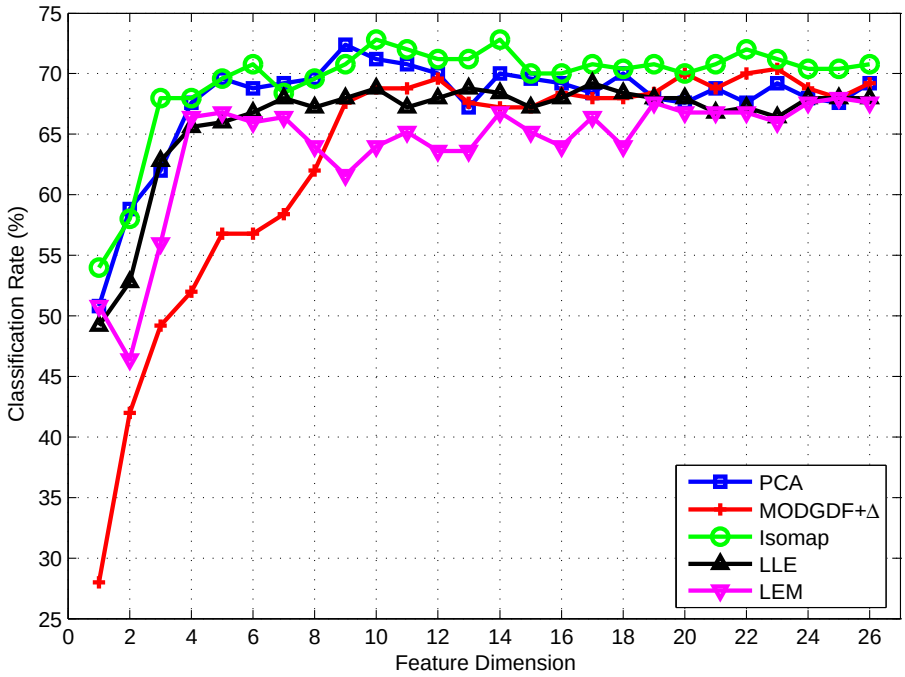


(b) MODGDF

Figure B.6: Phone class classification accuracy using (a) MFCC and (b) MODGDF features. The performance of each feature after dimensionality reduction by PCA, Isomap, LLE, and LEM is also shown.

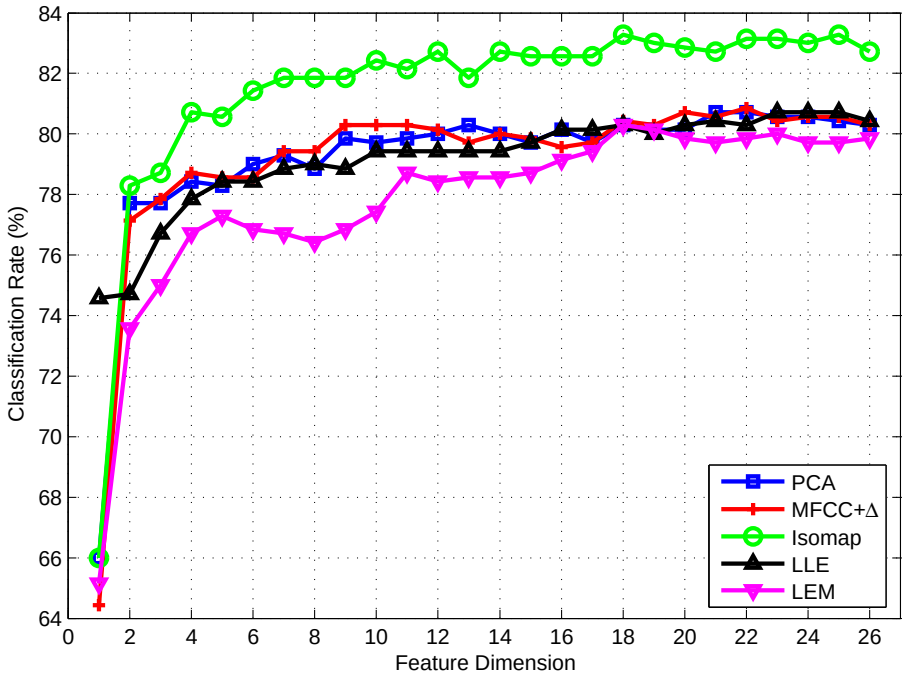


(a) MFCC+Δ

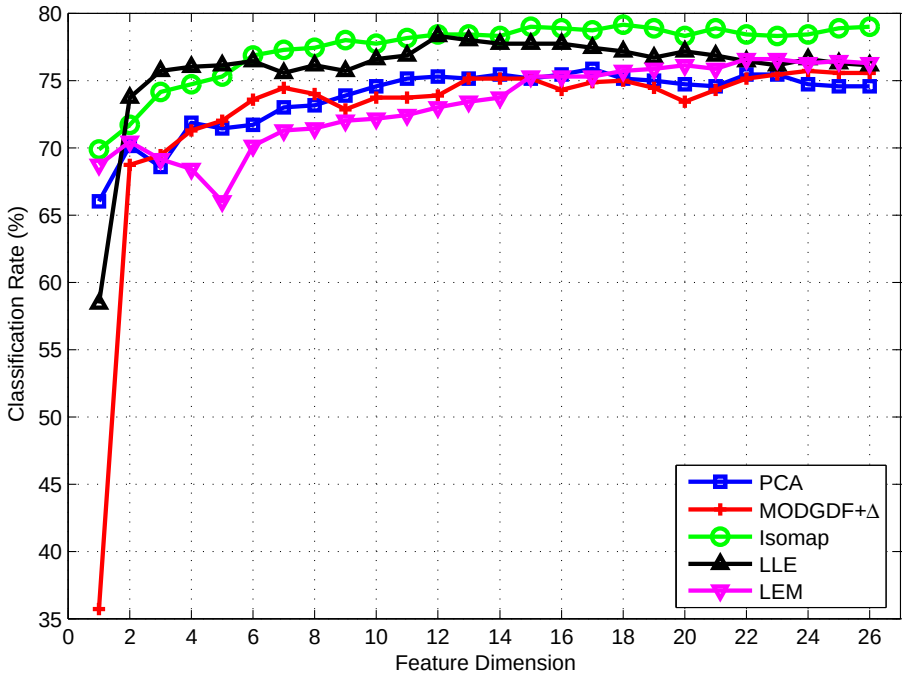


(b) MODGDF+Δ

Figure B.7: Five vowel classification accuracy using (a) MFCC+Δ and (b) MODGDF+Δ features. The performance of each feature after dimensionality reduction by PCA, Isomap, LLE, and LEM is also shown.

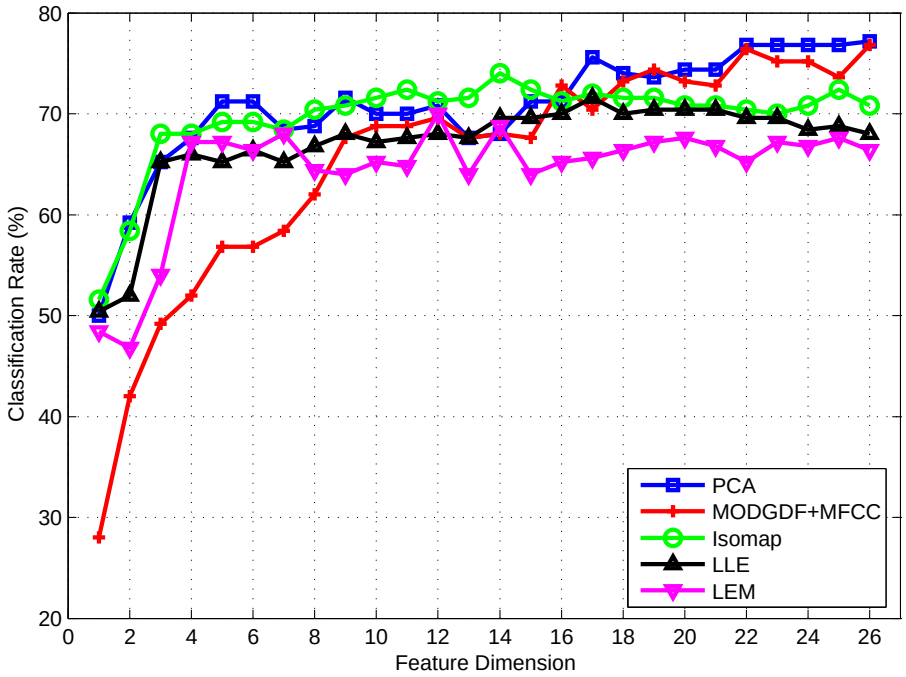


(a) MFCC+Δ

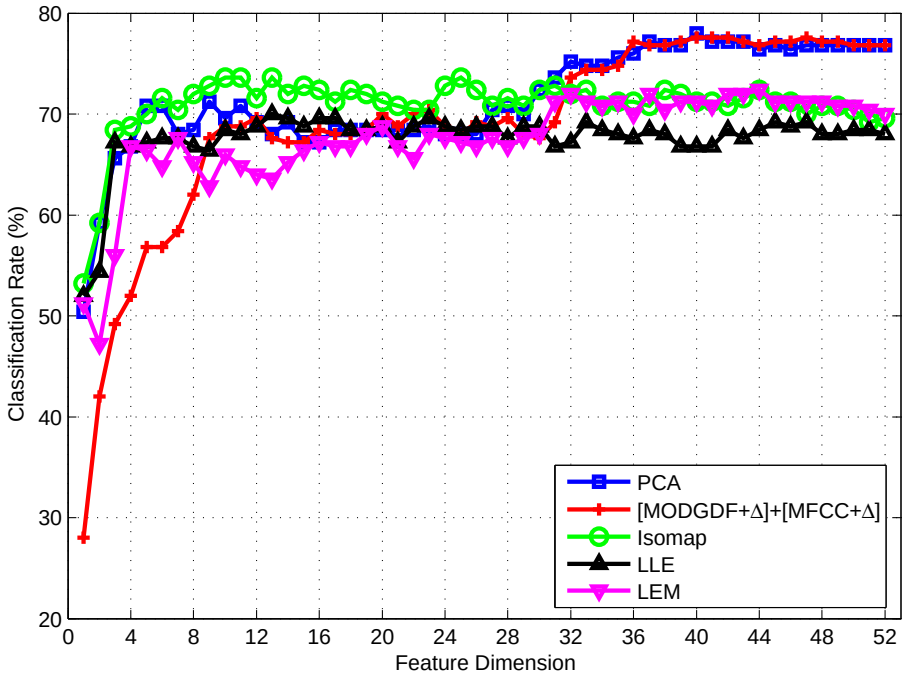


(b) MODGDF+Δ

Figure B.8: Phone class classification accuracy using (a) MFCC+Δ and (b) MODGDF+Δ features. The performance of each feature after dimensionality reduction by PCA, Isomap, LLE, and LEM is also shown.

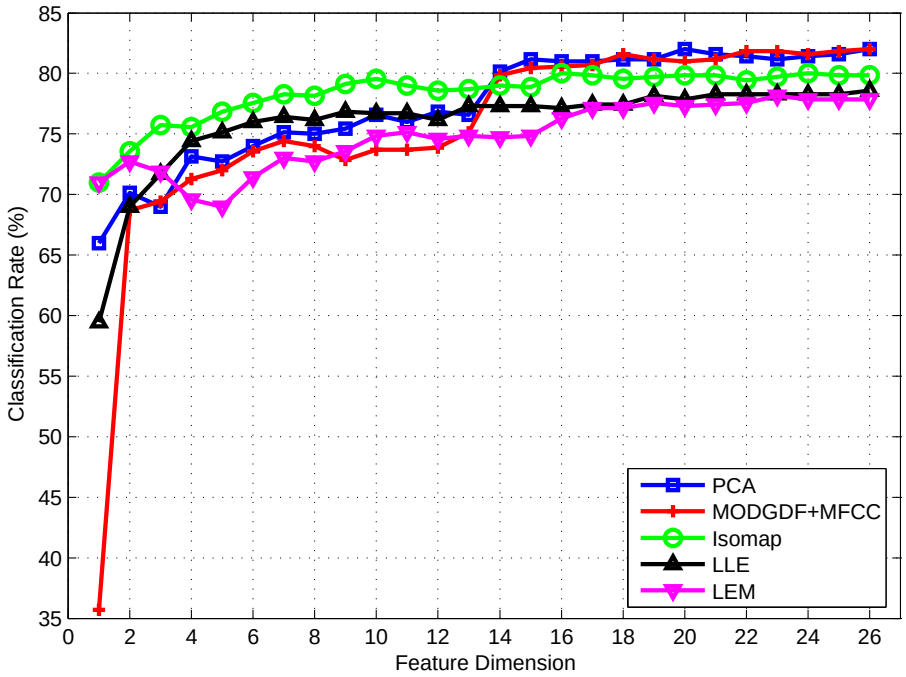


(a) [MODGDF+MFCC]

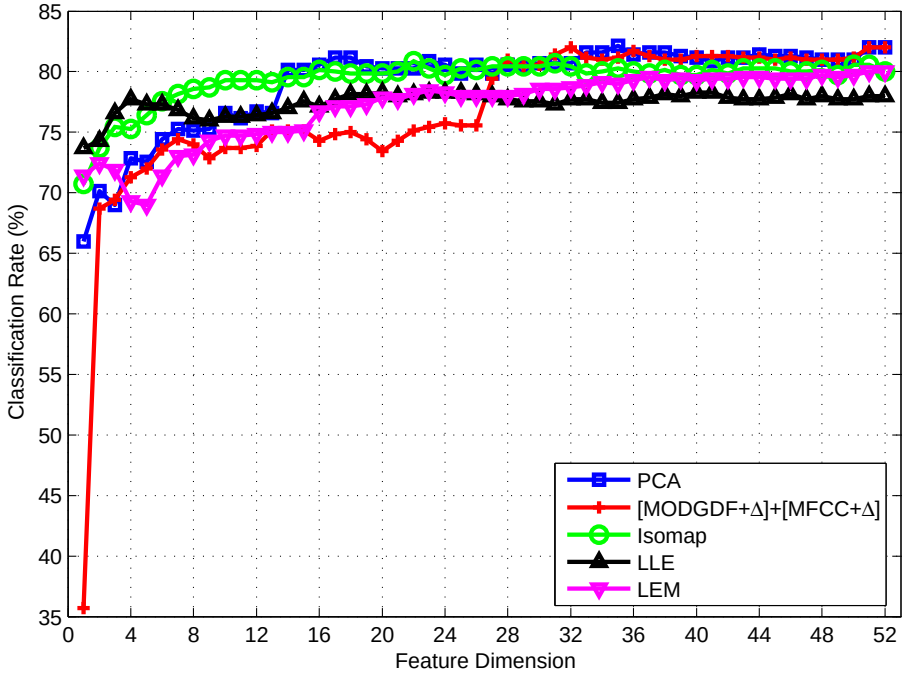


(b) [MODGDF+Δ]+[MFCC+Δ]

Figure B.9: Five vowel classification accuracy using (a) [MODGDF+MFCC] and (b) [MODGDF+Δ]+[MFCC+Δ] features. The performance of each feature after dimensionality reduction by PCA, Isomap, LLE, and LEM is also shown.



(a) [MODGDF+MFCC]



(b) [MODGDF+Δ]+[MFCC+Δ]

Figure B.10: Phone class classification accuracy using (a) [MODGDF+MFCC] and (b) [MODGDF+Δ]+[MFCC+Δ] features. The performance of each feature after dimensionality reduction by PCA, Isomap, LLE, and LEM is also shown.

References

Abarbanel, H. (1996). *Analysis of Observed Chaotic Data*. Springer-Verlag, New York.

Aizerman, M., Braverman, E., and Rozonoer, L. (1964). Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837.

Alder, M., Togneri, R., and Attikiouzel, J. (1991). Dimension of the speech space. *IEE Proceedings-I*, 138(3):207–214.

Alsteris, L. D. and Paliwal, K. K. (2005). Evaluation of the modified group delay feature for isolated word recognition. In *Proc. of the Eighth International Symposium on Signal Processing and Its Applications (ISSPA)*, volume 2, pages 715–718.

Atal, B. S. and Hanauer, S. L. (1971). Speech analysis and synthesis by linear prediction of the speech wave. *J. Acoust. Soc. Amer.*, 50(2):637–655.

Balasubramanian, M., Schwartz, E. L., Tenenbaum, J. B., de Silva, V., and Langford, J. C. (2002). The Isomap algorithm and topological stability. *Science*, 295(5552):7.

Banbrook, M. (1996). *Nonlinear analysis of speech from a synthesis perspective*. PhD thesis, The University of Edinburgh.

Banbrook, M. and McLaughlin, S. (1994). Is speech chaotic?: Invariant geometric measures for speech data. *IEE Colloquium on Exploiting Chaos in Signal Processing*, pages 8/1–8/10. Digest No 1994/193.

Banbrook, M., McLaughlin, S., and Mann, I. (1999). Speech characterization and synthesis by nonlinear methods. *IEEE Trans. Speech and Audio Processing*, 7(1):1–17.

Barney, A., Shadle, C., and Davies, P. (1999). Fluid flow in a dynamic mechanical model of the vocal folds and tract. i and ii. *J. Acoust. Soc. Amer.*, 105(1):444–466.

Bartholomew, D. (1987). *Latent Variable Models and Factor Analysis*. Charles Griffin & Company Ltd., London.

Baydal, E., Andreu, G., and Vidal, E. (1989). Estimating the intrinsic dimensionality of discrete utterances. *IEEE Trans. Acoustics, Speech and Signal Processing*, 37(5):755–757.

Behrman, A. (1999). Global and local dimensions of vocal dynamics. *J. Acoust. Soc. Amer.*, 105(1):432–443.

Belkin, M. and Niyogi, P. (2002). Laplacian eigenmaps and spectral techniques for embedding and clustering. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, pages 585–591, Cambridge, MA. MIT Press.

Belkin, M. and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396.

Bell, A. M. (1867). *Visible Speech: The Science of Universal Alphabetics, or Self-Interpreting Physiological Letters, for the Writing of All Languages in One Alphabet*. London: Simpkin, Marshall & Co., inaugural edition.

Bellman, R. E. (1961). *Adaptive Control Processes: A Guided Tour*. Princeton University Press, Princeton, New Jersey, U.S.A.

Bengio, Y., Paiement, J.-F., Vincent, P., Delalleau, O., Roux, N. L., and Ouimet, M. (2004). Out-of-sample extensions for LLE, Isomap, MDS, eigenmaps, and spectral clustering. In Thrun, S., Saul, L., and Schölkopf, B., editors, *Advances in Neural Information Processing Systems 16*, Cambridge, MA. MIT Press.

Beyerbach, D. and Nawab, H. (1991). Principal components analysis of the short-time Fourier transform. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 3, pages 1725–1728.

Bishop, C., Svensén, M., and Williams, C. (1998). GTM: The generative topographic mapping. *Neural Computation*, 10(1):215–234.

Bo, L., Hongbin, Z., and Wenan, C. (2008). Boundary constrained manifold unfolding. In *Proc. of the Seventh Int. Conf. on Machine Learning and Applications*, pages 174–181, San Diego, CA.

Boehm, J. F. and Wright, R. D. (1968). Dimensional analysis and display of speech spectra. *J. Acoust. Soc. Amer.*, 44(1):386–386.

Bozkurt, B. and Couvreur, L. (2005). On the use of phase information for speech recognition. In *Proc. of the European Signal Processing Conference (EUSIPCO)*.

Bozkurt, B., Doval, B., DAlessandro, C., and Dutoit, T. (2004). Appropriate windowing for group delay analysis and roots of ztransform of speech signals. In *Proc. of the European Signal Processing Conference (EUSIPCO)*.

Burges, C. J. C. (2005). Data mining and knowledge discovery handbook: A complete guide for researchers and practitioners. In Maimon, O. and Rokach, L., editors, *Geometric Methods for Feature Extraction and Dimensional Reduction*. Kluwer Academic Publishers.

Camastra, F. (2003). Data dimensionality estimation methods: A survey. *Pattern Recognition*, 36(3):2945–2954.

Campbell, J. P., Shen, W., Campbell, W. M., Schwartz, R., Bonastre, J. F., and Matrouf, D. (2009). Forensic speaker recognition. *IEEE Signal Processing Magazine*, 26(2):95–103.

Carreira-Perpiñán, M. A. (2001). *Continuous latent variable models for dimensionality reduction and sequential data reconstruction*. PhD thesis, Department of Computer Science, University of Sheffield.

Carreira-Perpiñán, M. A. and Renals, S. (1998). Dimensionality reduction of electropalatographic data using latent variable models. *Speech Communication*, 26(4):259–282.

Casdagli, M. (1991). Chaos and deterministic versus stochastic non-linear modelling. *Journal of the Royal Statistical Society B*, 54(2):303–328.

Chang, H. and Yeung, D.-Y. (2006). Robust locally linear embedding. *Pattern Recognition*, 39(6):1053–1065.

Childers, D. G. (1999). *Speech Processing and Synthesis Toolboxes*. John Wiley & Sons, Inc., New York.

Cox, T. F. and Cox, M. A. A. (2001). *Multidimensional Scaling*. Chapman & Hall, 2nd edition.

d'Alessandro, C., Darsinos, V., and Yegnanarayana, B. (1998). Effectiveness of a periodic and aperiodic decomposition method for analysis of voice sources. *IEEE Trans. Speech and Audio Processing*, 6:12–23.

Davis, S. B. and Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. Acoustics, Speech and Signal Processing*, ASSP–28(4):357–366.

de Lima, C. B., Alcaim, A., and Apolinario, J. A. J. (2002). On the use of PCA in GMM and AR-vector models for text independent speaker verification. In *Proc. of the 14th Int. Conf. on Digital Signal Processing (DSP)*, volume 2, pages 595–598.

de Ridder, D., Kouropteva, O., Okun, O., Pietikainen, M., and Duin, R. (2003). Supervised locally linear embedding. In Kaynak, O., Alpaydin, E., Oja, E., and Xu, L., editors, *Proc. of the Thirteenth Int. Conf. on Artificial Neural Networks*, volume 2714 of *Lecture Notes in Computer Science*, pages 333–341. Springer.

de Silva, V. and Tenenbaum, J. B. (2003). Global versus local methods in nonlinear dimensionality reduction. In Becker, S., Thrun, S., and Obermayer, K., editors, *Advances in Neural Information Processing Systems 15*, pages 721–728, Cambridge, MA. MIT Press.

de Silva, V. and Tenenbaum, J. B. (2004). Sparse multidimensional scaling using landmark points. Technical report, Stanford University.

DeCoste, D. (2001). Visualizing Mercer kernel feature spaces via kernelized locally-linear embeddings. In *8th Int. Conf. on Neural Information Processing*.

Deller Jr., J. R., Hansen, J. H. L., and Proakis, J. G. (2000). *Discrete-Time Processing of Speech Signals*. Wiley-IEEE Press.

Demartines, P. and Herault, J. (1997). Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets. *IEEE Trans. Neural Networks*, 8(1):148–154.

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

Denes, P. B. and Pinson, E. N. (1993). *The Speech Chain*. W. H. Freeman and Company, Oxford, England, 2nd edition.

Dijkstra, E. W. (1959). A note on two problems in connection with graphs. *Numerische Math*, 1:269–271.

Donoho, D. L. and Grimes, C. (2003). Hessian eigenmaps: new locally linear embedding techniques for high-dimensional data. *Proc. of National Academy of Sciences*, 100(10):5591–5596.

Duda, R. O., Hart, P. E., and Stork, D. G. (2000). *Pattern Classification*. Wiley-Interscience, 2nd edition.

Eisele, T., Haeb-Umbach, R., and Langmann, D. (1996). A comparative study of linear feature transformation techniques for automatic speech recognition. In *Proc. of the Int. Conf. on Spoken Language Processing (ICSLP)*, volume 1, pages 252–255, Philadelphia, PA.

Elgammal, A. and Lee, C.-S. (2004). Inferring 3D body pose from silhouettes using activity manifold learning. In *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 681–688.

Errity, A. and McKenna, J. (2006). An investigation of manifold learning for speech analysis. In *Proc. of the Int. Conf. on Spoken Language Processing (Interspeech 2006 - ICSLP)*, pages 2506–2509, Pittsburgh PA, USA.

Errity, A. and McKenna, J. (2007). A comparative study of linear and nonlinear dimensionality reduction for speaker identification. In *Proc. of the 15th Int. Conf. on Digital Signal Processing (DSP)*, pages 587–590, Cardiff, Wales.

Errity, A. and McKenna, J. (2009). A comparison of linear and nonlinear dimensionality reduction methods applied to synthetic speech. In *Proc. of Interspeech 2009 - Eurospeech*, pages 1095–1098, Brighton, UK.

Errity, A., McKenna, J., and Kirkpatrick, B. (2007a). Dimensionality reduction methods applied to both magnitude and phase derived features. In *Proc. of Interspeech 2007 - Eurospeech*, pages 1957–1960, Antwerp, Belgium.

Errity, A., McKenna, J., and Kirkpatrick, B. (2007b). Manifold learning-based feature transformation for phone classification. In Chetouani, M., Hussain, A., Gas, B., Milgram, M., and Zarader, J.-L., editors, *Advances in Nonlinear Speech Processing, International Conference on Non-Linear Speech Processing, NOLISP 2007, Paris, France, May 22-25, 2007, Revised Selected Papers*, volume 4885 of *Lecture Notes in Computer Science*, pages 132–141. Springer.

Fant, G. (1970). *Acoustic Theory of Speech Production*. Mouton, The Hague.

Fant, G., Liljencrants, J., and Lin, Q. (1985). A four-parameter model of glottal flow. Speech Transmission Laboratory Quarterly Progress and Status Report 4, Royal Institute of Technology, Stockholm.

Favella, L. F., Reineri, M. T., and Righini, G. U. (1969). On a mathematical procedure for detecting significant parameters in the classification of a statistical ensemble of phenomena. *Biological Cybernetics*, 5(5):187–194.

Fitzpatrick, R. (2006). *Classical Mechanics: An introductory course*. Lulu.com.

Flanagan, J. L. (1972). *Speech Analysis, Synthesis, and Perception*. Springer-Verlag, New York, 2nd edition.

Floyd, R. W. (1962). Algorithm 97: Shortest path. *Communications of the ACM*, 5(6):345.

Fukunaga, K. (1990). *Introduction to Statistical Pattern Recognition*. Academic Press, Inc., Boston, second edition.

Ganchev, T., Fakotakis, N., and Kokkinakis, G. (2005). Comparative evaluation of various MFCC implementations on the speaker verification task. In *Proc. of the Int. Conf. on Speech and Computer*, volume 1, pages 191–194, Patras, Greece.

Garofalo, J. S., Lamel, L. F., Fisher, W. M., Fiscus, J. G., Pallett, D. S., and Dahlgren, N. L. (1990). *The DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus CDROM*. NIST.

Gerstman, L. (1968). Classification of self-normalized vowels. *IEEE Trans. on Audio and Electroacoustics*, 16(1):78–80.

Glover, J. N. (1989). Detecting folds in chaotic processes by mapping the convex hull. In *Proc. Conf. on Dynamics of Complex Interconnected Biological Systems*, Albany, Western Australia.

Gobl, C. (2003). *The voice source in speech communication*. PhD thesis, KTH, Stockholm, Sweden.

Grassberger, P. and Procaccia, I. (1983). Measuring the strangeness of strange attractors. *Physica*, 9D:189–208.

Hadid, A., Kouropteva, O., and Pietikäinen, M. (2002). Unsupervised learning using locally linear embedding: experiments with face pose analysis. In *Proc. of the 16th Int. Conf. on Pattern Recognition*, volume 1, pages 111–114.

Hamidi, M. and Pearl, J. (1976). Comparison of the cosine and Fourier transforms of Markov-1 signals. *IEEE Trans. Acoustics, Speech and Signal Processing*, 24(5):428–429.

Hauskrecht, M., Pelikan, R., Valko, M., and Lyons-Weiler, J. (2007). *Feature selection and dimensionality reduction in genomics and proteomics*, chapter Fundamentals of data mining in genomics and proteomics, pages 149–172. Springer US.

Hegde, R. M. (2005). *Fourier transform phase-based features for speech recognition*. PhD thesis, Indian Institute Of Technology Madras.

Hegde, R. M. and Murthy, H. A. (2004). Cluster and intrinsic dimensionality analysis of the modified group delay feature for speaker classification. In Pal, N. R., Kasabov, N., Mudi, R. K., Pal, S., and Parui, S. K., editors, *Neural Information Processing - 11th*

International Conference, ICONIP 2004, volume 3316 of *Lecture Notes in Computer Science*, pages 1172–1178. Springer-Verlag.

Hegde, R. M., Murthy, H. A., and Gadde, V. R. R. (2007a). Significance of joint features derived from the modified group delay function in speech processing. *EURASIP J. on Audio, Speech, and Music Processing*, 2007(1):5–5.

Hegde, R. M., Murthy, H. A., and Gadde, V. R. R. (2007b). Significance of the modified group delay feature in speech recognition. *IEEE Trans. Audio, Speech and Language Processing*, 15(1):190–202.

Hegde, R. M., Murthy, H. A., and Rao, G. V. R. (2005). Speech processing using joint features derived from the modified group delay function. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 541–544.

Hirsch, M. W. (1976). *Differential Topology*, volume 33 of *Graduate Texts in Mathematics*. Springer, NY.

Horn, R. A. and Johnson, C. R. (1990). *Matrix Analysis*. Cambridge University Press.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol.*, 24:417–441 and 498–520.

Hsu, C.-W., Chang, C.-C., and Lin, C.-J. (2009). A practical guide to support vector classification. Technical report, Department of Computer Science, National Taiwan University.

Huang, X., Acero, A., and Hon, H.-W. (2001). *Spoken Langauge Processing: A Guide to Theory, Algorithm, and System Development*. Prentice Hall PTR.

IPA (1999). *Handbook of the International Phonetic Association*. Cambridge University Press.

Jain, V. and Saul, L. K. (2004). Exploratory analysis and visualization of speech and music by locally linear embedding. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 3, pages 984–987.

Jang, G.-J., Lee, T.-W., and Oh, Y.-H. (2001). Learning statistically efficient features for speaker recognition. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 437–440, Salt Lake City, UT, USA.

Jansen, A. and Niyogi, P. (2005). A geometric perspective on speech sounds. Technical report, University of Chicago.

Jansen, A. and Niyogi, P. (2006). Intrinsic Fourier analysis on the manifold of speech sounds. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 241–244.

Jin, Q. and Waibel, A. (2000). Application of LDA to speaker recognition. In *Proc. of the Int. Conf. on Spoken Language Processing (Interspeech 2000 - ICSLP)*, volume 2, pages 250–253.

Jolliffe, I. (1986). *Principal Component Analysis*. Springer Series in Statistics. Springer-Verlag, New York.

Jones, D. (1964). *An outline of English Phonetics*. W. Heffer & Sons, Cambridge, 9th edition.

Judd, K. (1989). Chaos in complex systems. In *Proc. Conf. on Dynamics of Complex Interconnected Biological Systems*, Albany, Western Australia.

Kayo, O. (2006). *Locally linear embedding algorithm: extensions and applications*. PhD thesis, Faculty of Technology, University of Oulu.

Kirkpatrick, B., O'Brien, D., and Scaife, R. (2006). Feature extraction for spectral continuity measures in concatenative speech synthesis. In *Proc. of the Int. Conf. on Spoken Language Processing (Interspeech 2006 - ICSLP)*, pages 1742–1745, Pittsburgh PA, USA.

Klein, W., Plomp, R., and Pols, L. C. W. (1970). Vowel spectra, vowel spaces, and vowel identification. *J. Acoust. Soc. Amer.*, 48(4):999–1009.

Kohonen, T. (1995). *Self-organizing Maps*. Springer, Berlin.

Kouropteva, O., Okun, O., and Pietikäinen, M. (2002). Selection of the optimal parameter value for the locally linear embedding algorithm. In *Proc. of the 1 Int. Conf. on Fuzzy Systems and Knowledge Discovery*, pages 359–363.

Kubin, G. (1995). Nonlinear processing of speech. In Kleijn, W. B. and Paliwal, K. K., editors, *Speech coding and synthesis*, pages 557–610. Elsevier.

Kuhn, R., Nguyen, P., Junqua, J.-C., Goldwasser, L., Niedzielski, N., Fincke, S., Field, K., and Contolini, M. (1998). Eigenvoices for speaker adaptation. In *Proc. of the Int. Conf. on Spoken Language Processing (ICSLP)*, pages 1771–1774, Sydney, Australia.

Kumar, A. and Mullick, S. K. (1990). Attractor dimension, entropy and modelling of speech time series. *Electronics Letters*, 26(21):1790–1792.

Ladefoged, P. (1967). *Three Areas of Experimental Phonetics*. Oxford University Press, London.

Law, M. H. C. and Jain, A. K. (2006). Incremental nonlinear dimensionality reduction by manifold learning. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 28(3):377–391.

Li, K. P., House, A. S., and Hughes, G. W. (1968). Vowel classification using a dispersion-analysis method. *J. Acoust. Soc. Amer.*, 44(1):390–390.

Lin, T. and Zha, H. (2008). Riemannian manifold learning. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(5):796–809.

Malayath, N., Hermansky, H., and Kain, A. (1997). Towards decomposing the sources of variability in speech. In *Proc. of the European Conf. on Speech Communication and Technology (Eurospeech)*, pages 497–500, Rhodes, Greece.

Mandelbrot, B. B. (1983). *The Fractal Geometry of Nature*. W.H. Freeman and Company, New York.

Mann, I. (1999). *An Investigation of Nonlinear Speech Synthesis and Pitch Modification Techniques*. PhD thesis, The University of Edinburgh.

Mathworks (2004). *MATLAB ®*. Mathworks, Inc., Natick, MA.

McKenna, J. G. (2004). *Kalman filtering towards automatic speaker characterisation*. PhD thesis, The University of Edinburgh.

McLaughlin, S. and Lowry, A. (1993). Nonlinear dynamical systems concepts in speech analysis. In *Proc. of the European Conf. on Speech Communication and Technology (Eurospeech)*, pages 377–380.

Milner, B. (2002). A comparison of front-end configurations for robust speech recognition. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 797–800, Orlando, FL, USA.

Murthy, H. A. and Gadde, V. (2003). The modified group delay function and its application to phoneme recognition. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume I, pages 68–71.

Narayanan, S. S. and Alwan, A. A. (1995). A nonlinear dynamical system analysis of fricative consonants. *J. Acoust. Soc. Amer.*, 97:2511–2524.

Newell, M. E. (1975). *The utilization of procedure models in digital image synthesis*. PhD thesis, University of Utah.

Nowak, M. A. and Krakauer, D. C. (1999). The evolution of language. In *Proc. Natl. Acad. Sci. USA*, volume 96, pages 8028–8033.

Oppenheim, A. V. and Schafer, R. W. (1975). *Digital Signal Processing*. Prentice-Hall, Englewood Cliffs, NJ.

Paliwal, K. and Alsteris, L. (2005). On the usefulness of STFT phase spectrum in human listening tests. *Speech Communication*, 45:153–170.

Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Phil. Mag.*, 2:559–572.

Peterson, G. E. and Barney, H. L. (1952). Control methods used in a study of the vowels. *J. Acoust. Soc. Amer.*, 24(2):175–184.

Pettis, K., Bailey, T., Jain, A., and Dubes, R. (1979). An intrinsic dimensionality estimator from near-neighbor information. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1(1):25–37.

Pijpers, M., Alder, M. D., and Togneri, R. (1993). Finding structure in the vowel space. In *Proc. of the First Australian and New Zealand Conf. on Intelligent Information Processing Systems*, Perth, Western Australia.

Pitsikalis, V., Kokkinos, I., and Maragos, P. (2003). Nonlinear analysis of speech signals: Generalized dimensions and lyapunov exponents. In *Proc. of Interspeech 2003 - Eurospeech*, pages 817–820, Geneva, Switzerland.

Plomp, R., Pols, L. C. W., and van de Geer, J. P. (1967). Dimensional analysis of vowel spectra. *J. Acoust. Soc. Amer.*, 41(3):707–712.

Pols, L. C. W. (1971). Real-time recognition of spoken words. *IEEE Transactions on Computers*, C-20(9):972–978.

Pols, L. C. W., Tromp, H. R. C., and Plomp, R. (1973). Frequency analysis of dutch vowels from 50 male speakers. *J. Acoust. Soc. Amer.*, 53(4):1093–1101.

Pols, L. C. W., van der Kamp, L. J. T., and Plomp, R. (1969). Perceptual and physical space of vowel sounds. *J. Acoust. Soc. Amer.*, 46(2B):458–467.

Quatieri, T. F. (2002). *Discrete-Time Speech Signal Processing: Principles and Practice*. Prentice Hall PTR.

Reynolds, D. A. (1995a). Automatic speaker recognition using Gaussian mixture speaker models. *MIT Lincoln Laboratory Journal*, 8(2):173–191.

Reynolds, D. A. (1995b). Speaker identification and verification using Gaussian mixture speaker models. *Speech Communication*, 17:91–108.

Reynolds, D. A. (2002). An overview of automatic speaker recognition technology. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 4, pages 4072–4075.

Rose, D. J. and Christina, R. W. (2005). *A Multilevel Approach to the Study of Motor Control and Learning*. Benjamin Cummings, 2nd edition.

Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326.

Roweis, S. T., Saul, L. K., and Hinton, G. (2002). Global coordination of local linear models. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, pages 889–896, Cambridge, MA. MIT Press.

Samko, O., Marshall, A., and Rosin, P. (2006). Selection of the optimal parameter value for the isomap algorithm. *Pattern Recognition Letters*, 27(9):968–979.

Sammon Jr., J. W. (1969). A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, C-18(5):401–409.

Sauer, T., Yorke, J. A., and Casdagli, M. (1991). Embedology. *Journal of Statistical Physics*, 65(3/4):579–616.

Saul, L. K. and Roweis, S. T. (2003). Think globally, fit locally: unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research*, 4:119–155.

Saul, L. K., Weinberger, K. Q., Ham, J. H., Sha, F., and Lee, D. D. (2006). Spectral methods for dimensionality reduction. In B. Schoelkopf, O. C. and Zien, A., editors, *Semisupervised Learning*. MIT Press.

Schölkopf, B., Smola, A., and Müller., K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319.

Schölkopf, B. and Smola, A. J. (2002). *Learning with kernels: support vector machines, regularization, optimization, and beyond*. Adaptive computation and machine learning. MIT Press.

Schroeder, M. R. (1991). *Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise*. W.H. Freeman, New York.

Schuster, M., Hori, T., and Nakamura, A. (2005). Experiments with probabilistic principal component analysis in LVCSR. In *Proc. of Interspeech 2005 - Eurospeech*, pages 1685–1688, Lisbon, Portugal.

Scott, D. W. (1992). *Multivariate Density Estimation. Theory, Practice, and Visualization.* Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, New York, London, Sydney.

Seung, H. S. and Lee, D. D. (2000). The manifold ways of perception. *Science*, 290(5500):2268–2269.

Shao, C. (2008). Incremental selection of the neighborhood size for ISOMAP. In *Int. Conf. on Machine Learning and Cybernetics*, volume 1, pages 436–441, Kunming, China.

Shao, C., Huang, H., and Wan, C. (2007). Selection of the suitable neighborhood size for the ISOMAP algorithm. In *Int. Joint Conf. on Neural Networks*, pages 300–305, Orlando, FL.

Somervuo, P. (2003a). Experiments with linear and nonlinear feature transformations in HMM based phone recognition. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 52–55.

Somervuo, P. (2003b). Speech dimensionality analysis on hypercubical self-organizing maps. *Neural Process. Lett.*, 17(2):125–136.

Steinbach, M., Ertoz, L., and Kumar, V. (2004). *New Vistas in Statistical Physics – Applications in Econophysics, Bioinformatics, and Pattern Recognition*, chapter Challenges of Clustering High Dimensional Data. Springer-Verlag.

Stevens, S. S. and Volkman, J. (1940). The relation of pitch to frequency: a revised scale. *American Journal of Pyschology*, 53(3):329–353.

Sun, X. (2002). Pitch determination and voice quality analysis using subharmonic-to-harmonic ratio. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 333–336, Orlando, FL, USA.

Sweet, H. (1877). *A Handbook of Phonetics*. Oxford: Clarendon Press.

Takens, F. (1981). Dynamical systems and turbulence. *Lecture Notes in Mathematics*, 898:366–381.

Tattersall, G. D., Linford, P. W., and Linggard, R. (1990). Neural arrays for speech recognition. In *Speech and language processing*, pages 245–290. Chapman & Hall, Ltd.

Teager, H. M. and Teager, S. M. (1990). Evidence for nonlinear sound production mechanisms in the vocal tract. In *Proc. NATO ASI on Speech Production and Speech Modelling*, pages 241–261.

Tenenbaum, J. (1998). Mapping a manifold of perceptual observations. In Jordan, M. I., Kearns, M. J., and Solla, S. A., editors, *Advances in Neural Information Processing Systems 10*, pages 683–688, Cambridge, MA. MIT Press.

Tenenbaum, J. B., de Silva, V., and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323.

Theiler, J. P. (1988). *Quantifying chaos: practical estimation of the correlation dimension.* PhD thesis, California Institute Of Technology.

Tishby, N. (1990). A dynamical systems approach to speech processing. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 365–368, Albuquerque, NM, USA.

Togneri, R., Alder, M., and Attikiouzel, J. (1992). Dimension and structure of the speech space. *IEE Proceedings-I*, 139(2):123–127.

Togneri, R., Alder, M. D., and Attikiouzel, Y. (1990). Speech processing using artificial neural networks. In *Proc. of the Third Australian Int. Conf. on Speech Science and Technology*, pages 304–309, Melbourne.

Tompkins, F. and Wolfe, P. J. (2009). Approximate intrinsic Fourier analysis of speech. In *Proc. of Interspeech 2009 - Eurospeech*, pages 120–123, Brighton, UK.

Townshend, B. (1991). Nonlinear prediction of speech. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 425–428, Toronto, Ont., Canada.

van der Maaten, L. J. P., Postma, E. O., and van den Herik, H. J. (2009). Dimensionality reduction: A comparative review. Technical Report TiCC-TR 2009-005, Tilburg University.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory.* Springer, N.Y.

Verbeek, J. (2004). *Mixture models for clustering and dimension reduction.* PhD thesis, Informatics Institute, Universiteit van Amsterdam.

Verbeek, J. J., Roweis, S. T., and Vlassis, N. (2004). Non-linear CCA and PCA by alignment of local models. In Thrun, S., Saul, L., and Schölkopf, B., editors, *Advances in Neural Information Processing Systems 16*, Cambridge, MA. MIT Press.

Wang, J., Zhang, C., and Kou, Z. (2003). An analytical mapping for LLE and its application in multi-pose face synthesis. In *The 14th British Machine Vision Conference*.

Wang, L. (2006). High dimensional data analysis. Technical report, Department of Statistics and Probability, Michigan State University.

Wang, X. and O'Shaughnessy, D. (2003). Improving the efficiency of automatic speech recognition by feature transformation and dimensionality reduction. In *Proc. of Interspeech 2003 - Eurospeech*, pages 1025–1028, Geneva, Switzerland.

Wang, Y., Huang, X., Lee, C.-S., Zhang, S., Li, Z., Samaras, D., Metaxas, D., Elgammal, A., and Huang, P. (2004). High resolution acquisition, learning and transfer of dynamic 3-D facial expressions. *Eurographics*, 23(3):677–686.

Yegnanarayana, B. and Murthy, H. A. (1992). Significance of group delay functions in spectrum estimation. *IEEE Trans. Signal Processing*, 40(9):2281–2289.

You, M., Chen, C., Bu, J., Liu, J., and Tao, J. (2006). Emotional speech analysis on nonlinear manifold. In *Proc. of the 18th Int. Conf. on Pattern Recognition (ICPR)*, volume 3, pages 91–94.

Young, S. (1996). A review of large-vocabulary continuous-speech recognition. *IEEE Signal Processing Magazine*, 13(5):45–57.

Young, S., Kershaw, D., Odell, J., Ollason, D., Valtchev, V., and Woodland, P. (2000). *The HTK Book Version 3.0*. Cambridge University.

Zhang, C., Wang, J., Zhao, N., and Zhang, D. (2004). Reconstruction and analysis of multi-pose face images based on nonlinear dimensionality reduction. *Pattern Recognition*, 37(2):326–336.