Learning Models of Speaker Variation

Michael John Witbrock

July 12th 1996

CMU-CS-96-135 3

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213-3890

*Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy.*

Thesis Committee:

Scott E. Fahlman, Chair
John Bridle, Dragon Systems UK.
Alexander H. Waibel
Sheryl Young.

This research was partially supported by the National Science Foundation under grant IRI-9214873. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. government.

ACM Computing Reviews Keywords: talker variability, speaker variability, speech synthesis, speaker adaptation, FFT, LPC, log area ratio, voice transformation, neural networks, canonical discriminant analysis, principal components, speaker modelling, non-linear discrimination, non-linear compression, second order networks, classification figure of merit.

Carnegie
Mellon

School of Computer Science

DOCTORAL THESIS
in the field of
Computer Science

Learning Models of Speaker Variation

MICHAEL JOHN WITBROCK

Submitted in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy

ACCEPTED:

Scott E Fahlman
THESIS COMMITTEE CHAIR

6 May 1996
DATE

THESIS COMMITTEE CHAIR

6 May 1996
DATE

DEPARTMENT HEAD

15 May 1996
DATE

APPROVED:

DEAN

5/17/96
DATE

Abstract

Speaker based variability is an important component of the speech signal, whether it is regarded as a nuisance, impeding speech recognition, or a goal, improving speech synthesis. Although many speech recognisers attempt to avoid errors caused by speaker variation, and a few synthesisers attempt to produce a wide range of voices, these efforts tend to be narrowly focused on the task at hand, rather than based on a general model of the variation. What work has been done on modelling variability itself, on the other hand, has mainly aimed at understanding specific linguistic events, rather than at providing an implementation that is practical.

This thesis attempts to bridge the gap between these two approaches, by using statistical and connectionist techniques to separate out, and to model, the speaker variability component of the speech signal. A number of these models are built and examined for speaker specificity and speed of convergence. Two applications for speaker models are studied with mixed results: speaker adaptation without parameter reestimation for recognition, and mimicry by transforming the voice personality of synthetic speech.

Table of Contents

Acknowledgments

The author would like express his gratitude to Raymond Watrous, for his valuable efforts in recovering the Peterson-Barney vowel data and making it publicly available, to Scott Fahlman, Alex Waibel, John Mount and the late Dr. Sheryl Young at CMU, to John Bridle at Dragon Systems, and to Gerald Cheang at Yale for helpful discussions, and to Warren Sarle or the SAS institute, for his many informative posts to "connectionists" on statistical matters. Special thanks are due to Vidya Dinamani for rescuing the results of many of the experiments from a particularly voracious file-system at the PSC, and to Arthur McNair and Ricky Houghton for attempting to satiate my inexhaustible appetite for disk space and machine time. Thanks are also due to Russell Leighton, from whose Aspirin/Migraines connectionist simulator the CDA code used for these experiments was obtained, and to Richard Stern and Wayne Ward, for lending me papers, and general helpfulness. The author is indebted to the members of the Informedia digital library research group at CMU for their forbearance during the completion of this document, and especially to Dr. Alex Hauptmann for his generosity in taking on some onerous duties during the final months of writing. Especial thanks are due to the National Science Foundation for supporting this research under grant IRI-9214873 .

Finally a debt that cannot be repaid is owed to Antoine Brusseau, for his support and patience, and for remaining in Pittsburgh and our household as this work extended itself almost indefinitely.

Chapter 1. Introduction to Speaker Variation

In computer science, most of the effort spent studying speech has been directed towards the problem of speech recognition, where the goal is to reduce the speech signal to a finite, invariable symbol set and to discard any components of the signal that do not serve to distinguish these symbols. Much research has been directed at supporting this goal by controlling and decreasing speech variability. The work described by this thesis seeks to do just the opposite - discarding the symbols that carry the meaning of the speech, and instead studying the variability in the way those symbols are produced. In particular, this thesis studies the variability associated with speaker identity.

While this effort to understand the variability of speech is interesting in its own right, it also has practical import for speech-based technologies. In the speech recognition domain, some forms of speaker variation, which are noise with respect to the problem, are not *a priori* recognisable as such, and must be identified by a model of variability before they can be eliminated. Unfortunately, there is presently a lack of such models at any but a very gross level, and speech recognition systems must be trained to treat most such sources of variability simply as noise. In speech synthesis, it is clear that the effort to produce a variety of natural sounding voices would be greatly aided by a study of the variability that underlies that variety.

Many previous speech recognition applications have addressed speaker variability by adapting their parameters over time to each new speaker. Others have sought to identify new speakers with particular members of a small set of speakers on whose speech the system has already been trained. The work described in this thesis takes a novel approach: an explicit model of the dimensions along which speakers can differ from each other is built, and, faced with a new voice, the system identifies where the voice fits within the model. This position in "speaker space" constitutes a speaker code, which, if the model is adequate, represents salient features of voice quality. Applications that need to deal with a variety of new speakers are trained to use this code as a source of information with which they can make speaker-specific adjustments to their processing. By designing the model of variability so that it generalises over speakers, producing reasonable codes for speakers outside the training set, it becomes possible to adjust speech applications to new speakers by simply identifying the position of each speaker's voice in the space of the voice model.

1.1. What the thesis does

This thesis applies neural network and statistical techniques to this task of characterising speaker variability. Models are formed of the variability found in speech segments, and these models are combined to form an overall model of speaker variability. Speaker models formed this way are applied to two tasks:

- improving a speaker independent speech recognition system by allowing it to better handle a variety of speakers' voices, and

- modifying a synthesised voice to more closely match that of a target speaker.

The neural net based recognisers, when applied to a realistic speech recognition task, proved stubbornly unable to use information about speaker identity, whether presented in the form of a voice model or more explicitly. Applying this speaker information to a simplified recognition task from the speaker adaptation literature was more successful. Although experiments aimed at accounting for this performance gap gave some insight into what the important differences were, it remains unclear how sources of speaker information can be made useful to "speaker independent" recognisers working in realistic domains. It is possible that investigating the question using the better-understood variety of speech recognition system based on hidden Markov models will permit this question to be answered in future, but the resources to do so were not available during this thesis.

The second application, to speech synthesis, was more successful at using the information contained in the speaker model, allowing the production of a variety of different voices from the same synthetic speech source. When coupled with improved speech synthesis system, and after refinement of both the speaker modelling technique and the voice transformation systems, this work should provide a solid foundation for future work aimed at improving the naturalness and accuracy of speech synthesis for multi-speaker applications.

1.2. The nature of speaker variability

This introduction will review work on speaker variability to provide a general context for work in speaker modelling. Discussion of the literature that is more closely related to the two applications of speaker modelling to speech recognition and synthesis will be deferred to the chapters concentrating on those applications.

First, the review will cover the sources of variability in the speech signal, distinguishing those that contribute to speaker characteristics from those that do not. Then previous attempts to model that variability will be discussed, sometimes with reference to the target applications. Finally, the studied approach to speaker modelling by positioning speakers in a speaker space will be outlined and contrasted with other approaches. This outline will be expanded in the following chapters.

1.3. Sources of variability in the speech signal

The central difficulty facing those who would do speech recognition or speech synthesis is the fact that the single speech signal carries a great many different kinds of information, and that this information is intermingled in a way that makes it very difficult, and sometimes impossible, to decompose. Roughly, these sources of information can be divided into three classes:

- *Linguistic:* information that conveys the speaker's intent to the listener.
- *Speaker identity:* information that conveys permanent characteristics of the speaker
- *State:* information that conveys transient states of the speaker or environment that

are not relevant to the meaning of the signal.

1.3.1. Linguistic information

Linguistic information is conveyed by the speaker's exerting control over the rate of airflow, the tension in the glottis, and the positions and rates of movement of the jaw, tongue and lips, to produce the periodic voicing or aperiodic fricative noise that gives power to the speech, and to shape the acoustic cavity that filters that sound into a signal representing the intended information.

In a naïve model of spoken language, one might view the speech signal resulting from these processes as representing a stream of discrete units — symbols such as words or phonemes — that convey the information in the signal. However, important information is also conveyed by prosodic effects, such as the rate at which and rhythm with which someone speaks, the relative pitch and amplitude of various parts of the utterance, and even by special effects, such as the use of devoiced (*i.e.* whispered) speech or the deliberate production of a mean fundamental frequency that is lower or higher than that of the speaker's normal voice.

Linguistic information is also conveyed by signal characteristics that might otherwise be interpreted as speaker differences. Abe [abe93] studied a single speaker of Japanese reading samples of text from a novel, an encyclopaedia entry and an paragraph of advertising copy. He found that the speech produced for the different genres of text showed effects on vowel formant frequencies similar to those distinguishing speakers. He also found a strong influence from text type on sentence duration, and on the relationship between fundamental frequency and power. These deliberate changes to speaking style are important not only because they bear information and must be accounted for in any complete speech understanding or synthesis system, but, more immediately, because they blur the boundaries between speech segments. A more detailed review of these effects can be found in Eskénazi's work [eskenazi93] which covers the influence speaking-style has on speech, and Péan's work [pean93] which discusses a database being constructed to investigate these effects further. While it is clear that much has been already been found out about speaking style, the similarity between changes in the speech signal due to speaker differences and those due to speaking style suggests that a fully explicated model of how text genre is conveyed by speaking style will ultimately depend on the construction of a good model of speaker differences.

1.3.2. Speaker Identity

There are several permanent characteristics of a speaker that affect voice quality. Characteristics of the glottis affect both the natural pitch of the voice and the shape of the glottal pulses that drive voiced speech. The range of dimensions that the vocal tract can adopt at the speaker's will, and the dimensions of the tract when relaxed, including the position and size of the tongue and lips, affect the range of harmonics that can be produced from a given driving signal, and the harmonic content that the vocal tract is more likely to produce.

As well as these anatomically derived characteristics of voice personality, there are long term preferences for the allophones a speaker uses to instantiate a given phoneme, or even which phoneme to use, in a given context. These differences may be as glaring as the difference between an English and an American pronunciation of the vowel in *"can't"*, or as subtle as the differences in voicing onset time for vowels following stop consonants for francophones and anglophones speaking English [flege94]. There are also permanent preferences for prosodic characteristics, such as segmental duration[1], and the degree of pitch, amplitude or duration stress used to mark semantic and syntactic events in an utterance.

Unfortunately, the effects of these speaker characteristics on the speech signal qualities are often indistinguishable, at least in the short term, from the effects of the language. What appears, over the short term, to be a high fundamental frequency, characteristic of a speaker with a high pitched voice, may turn out, in the long term, to have been a linguistic effect, such as *sotto voce* speech. And, even more obviously, an American speaker can choose to pronounce the word *"can't"* in the English way.

In general, many of the characteristics that make voices distinctive if they are observed from a speaker over the long term, can be produced for purposes of communication in the short term.

1.3.3. State

It is not just the communicative intent and long-term characteristics of a speaker, described above, that affect voice quality. External or internal events can change the environment in which speech is produced, or the internal state of a speaker, and affect voice quality. Vroomen *et al* [vroomen93] showed that a speaker's emotional state, or affect, was reflected by duration and pitch changes[2] in speech, and that these changes were sufficiently pronounced that even stylised versions were sufficient to convey emotion in synthetic speech. Emotion affects speech in a variety of ways; Murray and Arnott [murray93], in their extensive review of the literature on vocal emotion, identify seven aspects of voice quality, including rate, pitch, intensity and mode of articulation, affected by vocal emotion.

Speakers who are tired, or who are under stress, produce speech reflecting those states [arbe 1980 and Sulk 1977 in murray93], and speakers in very noisy environments produce a characteristic voice quality called "Lombard speech". Junqua *et al.* identified changes in twelve components of voice quality for this latter speech, many of which, like pitch, vowel duration and formant frequency, are also important to voice personality [junqua90].

These state based changes in voice quality may be especially problematic for attempts to rapidly model voice personality. They tend to occur over a reasonably long duration, and to affect many of those voice features, such as pitch and speaking rate, that are characteristic of speaker differences. That this confusion should be present is unsurprising if one accepts the notion of Brown *et al.* [1974 in murray93] that personality itself is simply "the characteristic emotional tone of a person over time". It is likely, then, that a model of speaker differences would be aided by research enabling modellers to make the effects of emotion and other state on speech explicit, and by the collection of speech corpora in which visible components of speaker state were labelled.

[1] Further complicated for languages, such as Japanese or Maori, in which segmental duration is phonemic.

[2] It should be noted that the emotive speech in this study was deliberately produced, and therefore could more properly viewed as linguistic. One hopes that it can be assumed that natural emotions have similar characteristics.

1.4. Models of variability

The foregoing discussion will have made it clear that the desire to separate variation due to linguistic phenomena from that due to speaker characteristics will not be easily satisfied. Sometimes the effects of these sources of variation have exactly the same form. The only consistent difference between the sources is that speaker characteristics are nearly permanent, and should be apparent in the long term statistics of the speech signal from a person, and that state and linguistic phenomena affecting voice quality occur over a shorter time course. The aim of this thesis work is to build a model of a speaker's voice that can, in a sense, be "subtracted" from the speech signal to yield a more consistent signal for speech recognition, or "added" to a synthetic speech to mimic a particular voice. Since it is desirable to characterise a new speaker's voice quickly, it will be assumed that the major factors determining the form of the final speech signal are:

- the phone string to be produced, and
- the talker's voice quality.

Medium term voice changes caused by mimicking speaker differences, or by the speaker's emotional or other state, will be treated as if they were produced by different speakers. It is to be hoped that this assumption will not be harmful to applications. Before describing the method for modelling speaker variation, some previous work on the topic of talker variability will be reviewed.

The literature on speaker variation falls into two main classes:

- explicit attempts, with primarily linguistic motivation, to characterise the variability in particular parts of the speech signal, and
- attempts to deal with variability in the pursuit of some particular task, such as speech recognition.

Although the latter research has not been aimed directly at speaker modelling, an implicit model of speaker variability is often apparent. In this section, both kinds of models will be reviewed.

1.4.1. Explicit Models

Since variability amongst speakers is interesting in its own right, quite apart from its possible application in speech recognition and production systems, there have been a number of studies that have attempted to study this variability. Roughly, these can be divided by the degree to which the variability is expressed in terms of phonological rules, or in terms of statistical variability in the speech signal. Since the system to be described in this thesis relies on a model of the latter kind, it is on these statistical models that the review will concentrate.

Phonological rules for variability

Kimura and Nara [Kimura87] view speaker variability in terms of the choice of a particular set of phonological rules that are applied to an orthographic (spelled) transcription of an utterance to transform it into a string of acoustical templates corresponding to the speech. The rule set they used was developed in the context of a speech segmentation system where the task was division of Japanese speech into a string of symbols representing phoneme variants. During training, the rule set, covering such transformations as palatalisation of consonants, nasalisation of vowels, and so on, was expanded for successive speakers whenever segmentation failed. A final set of 317 rules was sufficient to segment almost all of the speech. Interestingly enough, the rate at which new rules had to be added to the rule set to cover new utterances dropped dramatically after five speakers had been covered, suggesting that the types of phonological variability in the speech signal are reasonably few. Each speaker required about 53% of the available rules to describe his or her speech.

The interesting result in this work is that explicit models of the sources of segmental variability can be built, and that they can attain useful coverage from rules derived from relatively few training speakers. While the fact that each speaker uses a subset of the possible rules is interesting, it is not clear how one could use this characterisation in terms of rules to build a representation of a speaker's voice that could be used in other applications; it might take rather a lot of speech to decide which rule subset should be selected for a new speaker, especially if one wished to obtain probabilities for the application of alternate rules. Although it is possible that use of certain rules entails or predicts the use of others, and that such predictions could be used to group speakers by rule sets they are likely to use, this possibility was not investigated in the paper.

Vieregge and Broeders [vieregge93] looked for similar variability in a much narrower domain. They investigated variability in the realisation of the phonological variable /r/ in Dutch, where /r/ can have a variety of realisations depending on context. They found some talker specificity in the choice of realisation across speakers in some contexts, and also found variability in the degree of intra-talker variation. Unfortunately, insufficient data was available to clearly answer the question of whether even the variability in this single speech sound was regular and predictable from other speaker characteristics.

As will become evident when it is described, the model of speaker variation adopted for this thesis assumes that it is possible to identify which speech segments — typically, which phones — a speaker has used when producing an utterance, or that it is possible to chose correctly which phones to synthesise for a speaker. Although the research to date has not produced good algorithms to guide these choices, these studies of variation at the segmental level, since they can help provide a basis for these decisions upon which the model depends, will be very important at improving performance in the future.

Statistical characterisations of variability.

Although the statistical models in the literature tend to have a narrower scope than one might like, often concentrating on a limited set of phones or speakers, they do have the advantage over a purely rule based characterisation of variability that they can be learned fully automatically. They are also, of course, more directly comparable to the work reported in this thesis. Most directly comparable, perhaps, are members of a set of neural network models of speaker variation, which will be described in the next subsection.

One of these narrowly focused statistical models is found in the work of Heuvel *et al.* [heuvel93], who investigated the sources of variability in steady state portions of the three

Dutch vowels /a,i,u/ in C-V-C-ə context (e.g. "tatə"). They performed a discriminant analysis on bark scale spectra for the steady state portions in 10 repetitions each, by 15 male speakers, of these three vowels in 8 consonant contexts (/p,t,k,d,s,m,n,r/). The aim was to discover where in the vowel spectra the speaker-distinguishing information lay. Their conclusion was that most of this information was to be found adjacent to the spectral peaks in the speech; that it was chiefly formant shape, rather than formant position or some quality of the spectral troughs, that distinguished speakers, and that, moreover, these distinctions were captured by around four discriminant functions. Principal components analysis required more functions to capture the same amount of variability. Although there seem to be some problems in the interpretation of these results based to the difficulty of distinguishing the effect of a small formant shift on the variability of the speech signal near the spectral peak from that of a change in spectral shape, these experiments are interesting for two reasons: The first is that the technique of using discriminant functions to highlight variability is one that will be used in building the model described in this thesis. The second is that the limited number of discriminant functions required to model the speaker variability in this admittedly limited domain gives some hope that a reasonably compact, and therefore easily estimated, statistical speaker model might be obtainable.

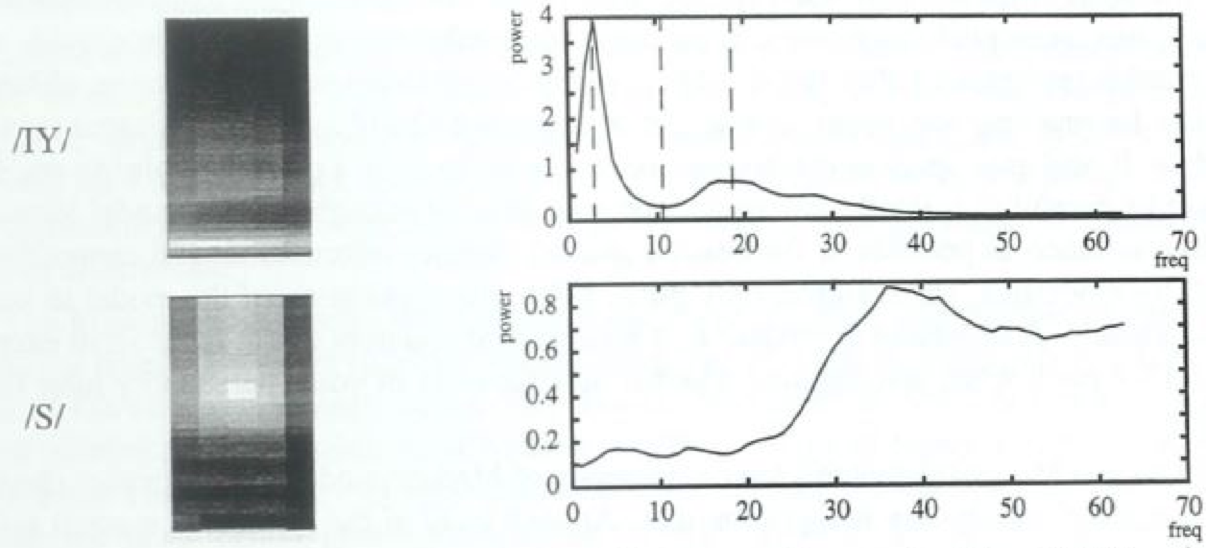


Figure 1: An example of the spectrum of two speech sounds, /S/ and /IY/ as in "see" (although the sounds displayed were not excerpted from an utterance of that word). The illustrations on the left have time on the x axis, frequency on the y access, and power represented as brightness. Each phone has been normalised to take up five time slices, or "frames". The plots on the right are the middle frame from each, with frequency on the horizontal and power on the vertical axis. For the vowel, /IY/, the middle three frames are nearly the same; this is the "steady state" portion of the phone. On the right, lines have been drawn to mark spectral peaks and troughs in this steady state portion. The frequency of the *n*th spectral peak is called the *n*th "formant".

Ward and Gowdy [ward89] used even simpler acoustic measurements to distinguish speakers, in this case for an application to speaker verification. Pitch at three points in the vowel of the word "stop" was measured, along with the duration of voicing for that vowel. Even with this simple model, voices were somewhat separable: a 70% correct speaker identification rate was obtained when the acceptance threshold, based on Mahalanobis distance[3], was set at a point where the numbers of false acceptances and false rejections were equal. It is not surprising, of course, that pitch is an important source of information for distinguishing speakers. Perhaps more interestingly, for male but not for female speakers in this study, the timing information also aided speaker discrimination. While Ward and Gowdy do not believe that mean pitch, on its own, is sufficient for speaker discrimination, the speaker discrimination performance they did manage to obtain using rather simple measures, although too low for practical use, points to a danger in attempts to build more sophisticated speaker models: if one uses speaker discrimination as a goal to base a model's training on, one should be careful to make the discrimination task difficult enough that the model learns to capture as much as possible of the desired speaker identity information, and cannot "succeed" by modelling, for example, only pitch. When the application of the model to voice transformation is discussed in chapter 6, it will become apparent that it is far from easy to control for pitch when investigating whether other sources of voice personality have been modelled.

Mathan and Miclet [mathan90] built a hierarchy of Markov models to do speaker clustering on a small-vocabulary recognition task. At each level of the hierarchy, a small set of adaptation words was used to chose which of two subordinate trees of models to use. The models at the leaves corresponded to speaker clusters. While this clustering technique is quite sophisticated, allowing modelling of both acoustic and timing variability, the authors reported that results were disappointing. Performance was better than for a system using a single Markov model for all speakers, but only insignificantly better than a more sophisticated recogniser (a "bi-model" recogniser) using two Markov models run in parallel for each word. Moreover, this insignificant improvement was only possible after fifteen adaptation words had been uttered. They note that performance of the bi-model recogniser was improved if, during training, words from a particular speaker were used to train only one of the models. This paper reflects a general trend in the literature of disappointing recognition results using speaker clustering techniques, and, disappointingly in terms of efficiency, better results from the use of parallel recognition models. More work on speaker clustering will be reviewed in the next section on neural networks.

Tishby [tishby88] derived a mathematically sophisticated framework for describing the effects of known contributors to the speech signal in terms of a combination of a prior model, such as the state means and covariances used in a standard model, and a set of constraints describing observables, such as speaker identity, related to the new information to be modelled. The aim was to use these constraints to select the one probability distribution, out of the set of possible distributions satisfying the prior model, that also predicted the observables, such as speaker means, describing the new information and that had minimum cross entropy with the prior model. In setting the parameters to achieve this minimisation, a representation of the observations was formed that could be used for clustering. As a demonstration, the technique was used to extend a prior model that divided a set of speakers by sex for a set of acoustic states within digits into one that described speaker means for these states. Clustering in the parameter space used for the transformation distinguished speakers better than a system trained from scratch to do speaker identification. While the technique was applied to speaker identification in [tishby88], it is possible that with a large set of training

[3] A measure of separation of distributions in a space that will be discussed later.

speakers, the parameter space formed could be used as a basis for a predictive speaker model. Once simpler models of speaker contributions to the speech signal of the sort described in this thesis have been successfully applied to a practical task, it may be worthwhile to investigate whether techniques such as Tishby's can be usefully applied to the general speaker modelling problem.

Lamel and Gauvain [lamel93] approached the problem of variability by training independent Markov models for each speaker, or in another case, for each sex. The problem of speaker variation was viewed as applying to the entire speech signal, and an entire Markov model used to model each speaker. In the final model, speakers were, in a sense, viewed as independent from each other, and no attempt was made to take advantage of regularities in voice differences.

The Markov models were run in parallel during recognition, and the speech labelled as coming from the speaker or group whose model had the highest probability of having produced the observed acoustic string. The problem of sparse training data was alleviated by first training a speaker independent Markov model and adapting copies of it to the individual speakers. The technique was very successful when applied to the tasks of speaker, sex and language identification, giving, for example, a text independent speaker identification rate of 98.3% after 2.5s of speech, for models adapted to TIMIT testing speakers starting from a seed model trained on the entire training set.

While this technique of running full Markov models in parallel is obviously worth considering if one wishes to do speaker ID from a known set of speakers, and while it clearly allows one to build a good model of each voice, it suffers from some deficits as a general speaker model. It is, of course, computationally expensive to run even a single Markov model recogniser. Running many in parallel compounds this cost. Moreover, the models must be pre-trained for speakers, and do not satisfy the criterion that speaker models should enable generalisation across speakers. Despite these difficulties, the success of this technique of running parallel specialised models makes finding generalisations that alleviate the problems an attractive prospect. Although doing so is outside the scope of the work discussed in this thesis, some approaches that might be taken will be discussed in the chapter on conclusions and future work (chapter 7).

A non-segmental model

The majority of models of speaker variation have attempted to characterise segmental variation, but this is not the only component of speaker difference. Itahashi and Tanaka [itahashi93] viewed the prosodic contour as an important component of variation due to dialect differences in Japanese. In particular, they examined f_0 contours for fourteen male speakers, each of whom represented a different Japanese dialect, reading a well known Japanese short story. These contours were approximated using a piecewise linear function. Eighteen aggregate statistics were calculated over parameters, such as starting F_0, slope and power, of the line segments. The resulting 18-element vectors were subjected to principal components analysis (PCA)[4]. The authors plotted the vectors for the fourteen dialects pro-

[4] A technique for characterising variation that will be described in detail in later chapters.

jected onto pairs of the first five principal components and observed that in some of these projections the speakers appeared to group into linguistically plausible dialect clusters.

While one might have wished that the data had been gathered over a larger set of speakers, the paper serves the useful function of pointing out that prosodic contours are an important component of speaker variability, including variability between speaker classes. Explicitly modelling variation in the prosodic contour in this way would be well worth pursuing further, especially if a more sensitive model than aggregate statistics can be constructed.

Statistical models of variability such as those reviewed above reflect a useful approach to understanding speaker differences. However, they have generally been either too narrowly or two widely focused. A concentration on particular linguistic phenomena can be so narrow that it is of little obvious use in applications dealing with multiple speakers. Or a model can capture a great deal of the variability in a small set of speakers at the cost of a loss of computational tractability or the ability to generalise to new speakers. The aim of this thesis is, of course, to build a model that is sufficiently expressive to capture a useful amount of the available information about speaker variation, but simple enough to be practical. Statistical techniques, such as PCA, used in some of the previous work will be among the tools applied to this task.

Neural Network models of variability

More closely matching the initial intention of this work, there have also been a number of attempts to use neural network models to characterise variability. Although neural network techniques are, in some respects, very similar to the statistical models described above, there are important differences. The first of these is motivation: while the aim of building an explicitly statistical model is generally to understand the variability in the signal itself, neural net models are frequently construed as offering a model of how human information processing works. Neural net models may sometimes be built in the hope of understanding how people represent speaker variability, but it is not clear that this hope is warranted. The second main difference is that a neural network generally represents a larger class of possible modelling functions than a particular statistical technique. This may or may not be an advantage; in applying a particular statistical model to a set of data the modelling assumptions — the form one expects the data to take — are usually explicit, and the causes of failure or success of the modelling effort are generally understandable. The operation of neural network models tends to be more difficult to analyse. On the other hand, by representing a larger class of possible models, neural networks may have a greater chance of initial success.

Artières and Gallinari [artières93] looked to a neural network non-linear auto-regressive model, where speech frames f_{t-1} and f_{t-2} are used to predict frame f_t, to improve performance at speaker classification over that of similar linear auto-regressive models that had been previously used [bimbot92]. Speaker identification was done for fifteen speakers from the TIMIT database. Separate model networks were trained for each speaker.

Like the multiple Markov model system described above [lamel93], this system attempted to model each speaker's voice separately, in this case as a 1st order autoregressive process. No representation of the voice was formed, except in the weights of the network used to model it. Interestingly, from the point of view of building speaker models on the basis of speaker discrimination, a more accurate model of the speech signal from each particular speaker did not necessarily translate into better speaker discrimination. They attempted to improve the extraction of speaker distinguishing information from the networks by learning to identify inputs that would generate correct speaker classifications. This boosted speaker recognition accuracy somewhat, but at a cost of requiring more input for a classification decision, since much of the information is thrown away. Nevertheless, the improvement is interesting, since it implies that appropriately chosen subsets of the speech signal can be used to improve the representation of speaker characteristics.

Konig and Morgan [konig93] constructed a rather simple model which viewed speakers as belonging to one of between two and five speaker clusters. The speaker code used was the long term average of cepstral parameters. These clusters were either supervised to distinguish men from women, or formed by an unsupervised *k*-means clustering in speaker code space. Neural nets were trained to classify incoming data into the clusters, and the binary decisions made by these networks used during training as an input to a phoneme classification network. In recognition, all cluster inputs were tried, and the one with the highest decoding probability was used for the whole utterance.

Results were disappointing, with not even the supervised clustering into males and females producing a significant performance improvement over the baseline. This disappointing result for speaker information added to the input of a single recogniser, as compared to schemes using separate recognisers for each group is consistent with the findings reported in chapter 5 of this thesis. The model explored by the Konig paper is one in which speaker differences divide speakers into acoustic clusters based upon long term spectral characteristics. Unfortunately, the assumption that the identity of these clusters can readily be used as additional information to neural net phonetic classifiers appears to be problematic, even when, as in this case, the system has the opportunity to try using all groups during recognition. Existing classifiers, at least, do not seem to be equipped to make good use of this type of speaker information.

Blackburn *et al.* [blackburn93] concentrated on speaker differences due to accent. They trained neural network classifiers to distinguish between Arabic-accented, Chinese-accented and unaccented Australian English when given features extracted from segmented phonemes as input. Separate networks were trained for stops, voiced and unvoiced phones, and energy dips, and their results combined over time to give an accent classification. Although classification error rates were not detailed in the paper, except by giving segment by segment accent confusion rates, the authors claimed that the system classified accents as rapidly as a trained phonetician. The model of variation implied by this work is, of course, that speakers fall into classes with transparent descriptions, and that these accent classes can be identified and used. If one is dealing with speakers with a variety of strong accents, it seems natural to assume that preclassification into these accent classes is likely to be useful, before attempting to form a more finely-grained speaker space, although, as the preceding paper showed, applying these classifications may still be problematic. What is even less clear is whether simple acoustic features such as those employed by Blackburn *et al.* are sufficient to describe more subtle forms of accent variation, such as those distinguishing speakers from different regions within the United States.

1.4.2. Implicit models

There have been few, if any, applications of explicit descriptive models of speaker variability to actual speech recognition or synthesis systems. Some systems, however, can be viewed as having a model of the nature of speaker variation that is implicit in their choice of a method for dealing with variability. Most of these systems are speaker-independent or multi-speaker speech recognisers, but such implicit models are also to be found in synthesis systems that allow user control of parameters meant to affect voice quality.

Most speaker adaptation schemes applied to speech recognition have involved partially retraining the system, before recognition, using a set of "adaptation" samples of the new speaker's voice. In essence, these systems have used a variety of methods to look for corresponding frames in the new and originally trained speakers' speech and to attempt to find a function that maps between them. When this mapping is found, it is either added to the system as a preprocessing stage used during recognition to relate codebook entries for the new speaker to those for the original speaker or speakers, or the training samples for the old speaker are converted into the voice of the new speaker via the mapping, and the recognition system is retrained with this new larger synthetic training set.

A number of recognition systems have used this adaptation by retraining scheme, with some variation in implementation. Furui [furui89] formed hierarchical trees, based on an inter-frame distance measure, for frames from both the reference and the new speaker, and then used distance measures computed between nodes of these tree structures to learn a transformation from position in the new speaker's tree into position in the reference speaker's. The system was designed to map between corresponding spectral clusters. Because this technique was based only on spectral structure, it could be performed on unlabelled, unprompted speech, and could be carried out during recognition. A disadvantage of the technique is that it required a large amount of training speech to estimate the cluster positions. In this model, voices were regarded as similar in structure, but different in realisation. While the acoustic frames emitted by a given speech state might differ, the relationship between states was regarded as consistent across speakers, enabling a correspondence to be found between trees.

Rigoll [rigoll89] adapted the IBM speech recognition system to a new speaker by having the speaker utter a subset (25% or five minutes) of the sentences that had been used initially to train the system. A mapping function was generated between speaker specific codebooks by time aligning the data from the old and new speaker, transforming the remaining 75% of the training data into the estimated templates of the new speaker, and retraining the system on the new synthetic and natural speech. Similarly, but using mapping during recognition, instead of during training, Nakamura and Shikano [nakamura89] had their system learn a mapping between a fuzzy labelling[5] of frames for the new speaker and a fuzzy labelling for the reference speaker in a standard hidden Markov model system. They defined a fuzzy labelling as a scheme where frames are represented by the set of probabilities that the frame was generated by each label.

Watrous [watrous91a] was one of the first to suggest the use of neural network models to separate out the effects of different sources of variation on the speech signal. He suggested

[5] This system is called a fuzzy vector quantization in the paper.

that this variability was best modelled by regarding each phoneme as having a canonical form that is modified as each form of variation, such as loudness, context, or speaker, is introduced. He suggested, moreover, that these transformations were reversible. In experiments done using the Peterson and Barney database [peterson52,watrous91b], which will be described in detail later in the chapter on speaker adaptation in this thesis, Watrous showed that normalisation of formant frequencies, by specialised neural nets using multiplicative connections, significantly reduced classification errors. Inter-speaker variance was also reduced for phonemes from the TIMIT database, although the effect of this reduction on classification accuracy was not specified [watrous90,91a,93]. It was Watrous' early success with improving recognition of vowels from formant pairs by using speaker information, together with Cottrell's work on modelling variability in faces using compression networks [cottrell90], that initially motivated the work reported in this thesis. However, the technique used by Watrous required the use of labelled speech from the new speaker to train the input transformation, making it somewhat difficult to apply in many contexts where adaptation would be desirable, and required training of the transformation network for each new speaker, a relatively time consuming process. The aim of the work contained herein was to find adaptation techniques that did not share these faults. Unfortunately, the successes of Watrous' pilot experiments, and of the replications of them described in chapter five, were not matched when the same sort of normalisation techniques were applied to more ambitious tasks. It remains, therefore, a matter of controversy whether the effects of various kinds of variability on the speech signal are separately reversible.

Hernandez-Mendez and Figueras-Vidal [hernandez-mendez93], developed speaker models based on performing a "non-linear principal components analysis" of acoustic frames using neural networks specialised for each speaker. The speaker classifications produced by these models were used to combine the results of speaker-specific recognisers in a spoken digit recognition task for four male and two female speakers. Five repetitions for each of the ten digits were used for training, and five more repetitions, collected after a one month delay, were used in testing. High energy frames, representing voiced segments of the digits, were extracted and modelled by projecting them onto a lower dimensional space using self organising feature maps, radial basis functions, a variety of back-propagation networks, and principal components. Unlike the models described in [witbrock92] and in this thesis, models were not specialised for phonological units - frames for all digits were modelled in a single network.

Speaker identification performance using single digits from the test set ranged from 25% to 29% error for the network speaker model, and 35% error for the principal components model. Discriminant training, using null targets for frames from other speakers, improved speaker recognition performance by about 20%.

Results for applying the speaker discrimination from these speaker-identifying networks in a digit classifier were mixed. While choosing the best speaker on a digit by digit basis improved performance relative to choosing frame by frame or utterance by utterance, best recognition for training speakers was obtained by running all speaker dependent models in parallel. For novel speakers, merging speaker dependent recognisers using speaker models did not produce any additional gain in performance improvement when more than four speaker-specific recognisers were combined, and in general did not improve on the performance of a single multi-speaker recogniser.

In this paper, Hernandez-Mendez *et al.* modelled a speaker's voice quality by building specialised filters for their voices, and speaker similarity was judged by the amount of distortion introduced by these filters. A new speaker was modelled by using a mixture of these filters. Although some success from such filters might be expected in multi-speaker domains, even ones more realistic that the digit recognition task used, the models did not generalise well for new speakers. It is this failure to generalise that the explicitly speaker-independent voice models described in this thesis were intended to address.

Cox and Bridle [cox89,90, bridle91] described neural network based systems that, like Watrous' were based on the notion that there is a canonical form for speech that is affected by speaker dependent transformations to produce the final speech signal. Given a speech stream from a new speaker, their system simultaneously searched for the most probable sequence of recognised symbols and the most probably correct speaker dependent transformation function, according to these three ways of modelling the speaker effect:

- As a spectral bias - each speaker added a constant bias to each of twenty-seven spectral frequency channels.
- As a variable shift plus bias - each speaker added a fixed bias vector, plus a weighed combination of a three channel window to form a channel. The weights for the combination were fixed. This model represented a shift of up to one channel up or down.
- As a similar transformation, except the weights could vary across the spectrum, allowing variable but limited spectral shifts, expansions and compressions.

In these three models, the speaker differences were purely acoustic and uniform across time. There was no provision for varying the transformations depending on phone class, for example. On the other hand, since there were few parameters to be estimated, the small speaker adaptation effect available from these modelled speaker differences could be obtained with relatively little adaptation speech, and in relatively little time, event though back-propagation was used to do the search for speaker specific parameters.

Hampshire and Waibel's meta-pi network adapted a recogniser to new speakers by selecting amongst multiple speaker specific recognisers on a frame by frame basis, using a learned weighted average of the individual recognisers' outputs to make a final decision. This network was found to perform significantly better than a single recogniser trained on the multiple speakers [hampshire92]. This model differed from that of [hernandez-mendez93] in that the speaker model was the weighting between the recognisers, and was optimised on a case by case basis for each speaker, rather than being computed by an independent network. Speaker modelling was implicit in the way the recogniser operated, rather than being performed in a separate subsystem. The advantage of networks such as this, which are constructed to group parameter subsets by speaker, over a single multi-speaker recogniser, may lie in the ability of the speaker specific models to more sharply cover their input space, rather than modelling the between speaker variance. If this is the case, the model of speaker variation implied is one in which individual speakers have substantially less acoustic variability than multiple speakers, and where speakers cluster to a sufficient extent that using a particular speaker's models — or a weighted combination of a set of speakers' models — is closer to the truth about a new speaker than is an overall mean representation derived from all speakers' speech.

More conventional — non connectionist — recognition systems have done something with similar effect: by increasing the number of codebook entries[6] they use, they can divide the input space into output classes more accurately. This sort of technique, of course, is limited, in that it selects between codebook entries derived from different speaker classes on the basis of the single frame of input speech being processed. Each of these frames must still contain all the information needed for its classification. Such systems cannot use information from previous frames to improve performance, as human beings have been shown to do [mullenix89]. One could imagine a variation on this scheme in which the system is encouraged to use the same speaker model for contiguous frames, switching between reference speakers more slowly. In the systems reviewed this was done in one respect: Some systems, such as Sphinx [hwang94] have maintained both male and female models, and chosen the best one to analyse an entire utterance. While this does model a characteristic of the speaker, the technique is not easily extended to a wider set of speaker characteristics. The attempt has been made, of course. The system in [lamel93], described above, does the same thing in a much more dramatic way, maintaining parallel Markov models for all speakers.

All these systems share the characteristic that they effectively retrain the system, or at least a subset of the system, to be a speaker dependent recogniser for the new speaker. One can draw a distinction between this learning-based "adaptation" to a new speaker, and a system that is able to collect information about a speaker to make a transient adjustment in its recognition strategy. It is the latter strategy that is pursued in this thesis.

1.5. A space of speakers

The models of speaker variation evident in the work described above are limited in one respect or another. If they are explicit, they are designed to cover variation in a narrow phenomenon. This limits their applicability in two ways: it limits the amount of the speech signal that the model can make use of, and it may limit the applicability of the model to parts of the speech signal not explicitly modelled. For implicit models, one is limited to trying to build the complete model for a speaker only with the speech one has already heard from that speaker, or to regarding the speaker as identical to its nearest neighbour in some small set. The former limit makes it difficult to apply models to applications where limited speech is available from the target speaker. The latter limit makes it hard to model novel speakers with any fidelity.

1.5.1. What people can do

Human beings show a remarkable ability to handle speaker variation. Despite voice differences, the average person is able to understand utterances from a wide variety of speakers correctly and with little or no apparent effort. There is substantial experimental evidence showing that human listeners use information from earlier utterances to influence the processing of later input; the identification of vowels by human listeners is more accurate when the stimuli are drawn from a single speaker than when they are drawn from multiple speakers, for isolated vowels [assman82], and for consonants [Fourcin, 1968 in mullinix89]. Sim-

[6] A codebook entry is a kind of "reference template" for part of the speech signal. The signal is modelled by comparing it to a sequence of these codebook entries.

ilar effects apply for recognition of monosyllables [creelman57]. These effects of talker change on human speech recognition accuracy apply strongly and consistently to whole word recognition tasks as well [mullenix89, mullenix90]. Since the effects include changes in processing latencies for words, the existence of a system for low level mandatory processing of speaker differences is implied. On the other hand, there is some dispute over how important these effects are in everyday performance [e.g. creelman57] and even whether they are important at all [verbrugge76]. More research is warranted in this area to fully understand the nature and role of speaker adaptation in human performance. Nevertheless, since, currently, speech recognition systems trained speaker-independently do suffer performance decreases relative to speaker independent systems, it is desirable for them to be able to improve their performance by adapting to the voice of the speaker, as humans do in at least some cases.

1.5.2. The model

What this thesis proposes is that there are underlying regularities in the way speakers' voices differ, and that these regularities can be used to amplify the usefulness of a small sample of a speaker's voice. Using data from a large number of training speakers, models are constructed for the variability of in each of a number of speech segments.[7] Using knowledge of which speaker said each segment, these segmental models are used to construct an overall model across speakers of the relationships between segmental pronunciation within a speaker. This overall voice variation model can be used with information, extracted from the appropriate segmental model, about the pronunciation of a given segment to make predictions about the pronunciation of unheard segments from the same speaker. Once it is trained, the model can be run in an entirely feed forward manner, allowing it to be applied in a straightforward manner to speech tasks that might benefit from speaker specific information. The feed-forward mode of operation also ensures that use of the model imposes a very limited computational demand.

1.5.3. Comparison with others

Earlier, speaker models were divided into two classes for review: explicit models that attempted to form a representation of vocal personality as such, and implicit models, that used some representation of voice personality formed in the course of trying to perform a task.

Since the model developed here contains an explicit representation of a speaker space, and can place speakers within it independent of any particular task, it is an explicit model. However, it shares, in some respects, more of the qualities of some of the implicit models. In particular, it is closely related, in application, to the models of Cox and Bridle, and of Watrous. The difference is that the speaker representations in their models are formed as the result of parameter optimisation in the course of a speech representation task, the current model is trained to produce a voice personality representation for each of a large set of training speakers in a manner designed to permit generalisation to new speakers. Beyond that needed

[7] In fact, the models built in this thesis have always covered the variability in phonemes, but that is for practical, rather than theoretical reasons.

to form the model from the initial training set, no further parameter optimisation should be needed to handle new speakers.

1.5.4. Applications

The main aim of this thesis has been the construction of statistical and neural network models of speaker variation. Ideally, one would evaluate such models by measuring their ability to capture salient differences between speakers. Unfortunately a well articulated set of parameters for describing voice differences remains to be discovered. This makes the evaluation task more complicated. Although, for the sake of comparison, two measures of speaker discriminability are applied to the models to approximate a measure of speaker model quality, it must be noted that increasing the system's ability to distinguish speakers is not necessarily the same thing as increasing its utility in describing their voices.

Since a direct measure of model quality is lacking, the model has had to be evaluated in the context of specific tasks for which voice differences are important, at the cost of not inconsiderable extra effort. If the model can provide the information about speaker voice needed by some realistic applications where speaker differences matter, it will have shown its virtue as a representation of a speaker's voice quality.

Recognition

One of the motivations for this research was the reported and evident ability of human beings to use rapid adaptation to new speakers to improve speech recognition accuracy. This, together with the extensive literature showing improvements in recognition accuracy for speech recognisers adapted to new speakers led to the belief that a connectionist speech recogniser, given an adequate description of a speaker's voice, would be able to use the information to adjust its classification surfaces, leading to improved recognition performance. Regrettably, it turned out to be the case that even providing a perfect speaker description, in the form of speaker ID, to such nets, had little effect on recognition accuracy. Although it was possible to get some information about why this may have been the case, the fact remained that this task was not serving well as way to measure model quality. In future work, it is hoped that there will be opportunity to apply speaker models similar to those described here to a hidden Markov Model (HMM) based speech recognition system. The explicit statistical representation of the spectral characteristics of the input signal used by these models, by allowing an understanding of the relationship between the position of a speaker in speaker space and the acoustics of the speech signal they produce, may allow an explanation of why the neural net based recognisers were unable to make use of this information. It is to be hoped that this understanding will permit the design of recognisers that are able to use speaker identity information, in a single recogniser, without retraining.

Voice transformation

Since speech synthesis based on the model would permit direct perception of the effects of different speaker descriptions, voice mimicry was selected as an alternative application domain. Using networks trained to convert SoftTalk[8] speech into the voice of the speaker described by the speaker voice description, it is possible to produce speech that, although not of terribly high quality, is similar in sound to that of the modelled speaker. The low quality of the converted speech is at least in part due to the use of synthetic speech as the source signal that is converted into the voice of the target speaker. Further quality deficits were attributable to the fact that it was necessary to work entirely with speech represented in a reasonably low quality linear predictive coefficient (LPC) encoding. The performance of the model and of the voice conversion system was evaluated on the basis of perceptual experiments using human subjects.

1.6. Outline of chapters

The next chapter gives an outline of the speaker model, and the speaker voice descriptions it yields. Chapters 3 and 4 describe the model in more detail: the first of these describes the phone pronunciation models from which the overall model is built, and compares a variety of techniques that were tried for forming these phone models; chapter 4 describes the overall speaker model, again comparing a number of methods for obtaining it. Chapter 5 describes the experiments with speech recognition, including some reasonably extensive work exploring the circumstances under which a recogniser would use a speaker model. Chapter 6 describes the application of the speaker model to mimicry synthesis by voice transformation, both comparing competing voice models, and various methods of achieving the voice transform for quality. Finally, chapter 7 draws overall conclusions from the presented experimental work, and suggests future directions to take, both with voice modelling, and with mimicry synthesis.

[8] A commercial speech synthesis system produced by Digital Equipment Corporation.

Chapter 2. Measuring Voice Characteristics

2.1. Introduction - Speaker Models

The most important quality of a useful model of speaker variation is productivity. The model must produce a representation of previously heard speech that allows it to make predictions about future speech from the same speaker. For speech synthesis, the aim is to alter synthesis parameters and produce novel utterances in the speaker's voice. For speech recognition, the aim is to alter a system's expectations about such things as phoneme boundaries in acoustic space and the timing of forthcoming speech, so as to improve its recognition performance.

As a degenerate case, a set of labels uniquely identifying all speakers is a suitable speaker model, when used with a application system that has been trained on the same speakers. A recognition system using such a label model must have heard enough speech from a given speaker to estimate the speaker specific parameters it uses to set classification boundaries. A similar synthesis system must have learned a suitable set of synthesis parameters for each speaker. In this degenerate scheme of using labels as a model, the modelling system would have the task of identifying the speaker and assigning the appropriate label.

In the more interesting case that speech from novel speakers is to be recognised or mimicked, or where a reliable method of identifying speakers is not available, it is necessary to look elsewhere for a model. Since there is available no *a priori* knowledge about whom the speech comes from, a model must describe a means of extracting a set of features, describing the speech personality of any particular speaker, from the speech signal itself. Henceforth, such a set of features extracted for a particular speaker will be called a Speaker Voice Code (SVC). Since it is desirable to form this SVC as rapidly as possible, and to have the SVC easily useable by application systems, one criterion for choosing the model's features is that they should be as stable as possible within a speaker. Of course, the features must also successfully distinguish what *is* distinct about the speech of different speakers, preferably in an application independent manner.

If the parameters are chosen appropriately, so that speaker class characteristics can be exploited, this sort of general speaker model should allow better estimation of adaptation or synthesis parameters, where limited training data is available for each speaker, than is available from speaker identity alone, since parameter estimates can be smoothed by those of other similar speakers.

2.2. Elements of a model

For speech technology applications, at least, the most important aspect of the speech signal is its symbolic content: a string describing *what* is said. In speech recognition, the aim is to transcribe or otherwise identify this string, or at least some useful part of it. In synthesis, the aim is to produce speech corresponding to the string. Although the assumption is not completely warranted, for the current purposes it can be supposed that the content of the speech is independent of who is speaking. In building the speaker voice model, it is clearly not desirable to capture variation due to what is being said. In fact, the goal is to avoid doing so, by separating this variation from the rest of the signal, and only modelling what is left. How each of the audible symbols (lexemes, for example, or smaller "segments" such as syllables or phonemes) making up the meaning of an utterance is represented in the speaker's voice is the essence of speaker variation. The elementary modelling technique used in this thesis is that of holding symbolic content constant, modelling the variation *within* acoustic symbols, and then combining the representations of this variation into an overall model of the speaker's voice. The choice of symbols will be discussed more thoroughly later, but generally a subset of phonemes will be used.

It is, of course, the case that the meaning of an utterance is also partially conveyed by prosody. Pitch, amplitude, rate, presence of voicing and other components of the speech signal can be varied at both the segmental and suprasegmental level to change the meaning of the string. Moreover, differences in the way these prosodic effects are applied are an important part of voice personality - distinguishing English from American voices, for example. Lacking an adequate method for separating prosody from other components of the signal, or the personality related aspects of prosody from the semantic, there is little choice but to treat prosody as simply a source of noise in the speaker model.

The technique, then, is to segment phones from the speech signal using a phoneme labeller,[1] to model the variation between examples of the same phone uttered by different speakers, and to combine these models into an overall speaker model. Inconveniently enough however, acoustic symbols vary greatly in duration, whereas, in general, the available techniques for modelling variation require that their input be a set of fixed length vectors. As a first step, therefore, it is necessary to produce fixed length vectors from the phoneme segments in the speech signal. This will be done by simply applying a linear transformation to stretch or shrink the phoneme segments extracted from utterances to constant duration. Other methods of fixing the length of speech symbols will, however, be discussed in a later section (3.5.).

The application of these speaker models, or rather of the SVCs produced by them, depends on it being possible to train the target system (a synthesiser or recogniser, for example) with the SVCs from some limited set of training speakers, and on the system being able to generalise to using SVCs from novel speakers to aid it in its task. For this hope to be justified, the space of SVCs has to be reasonably well populated by the training speakers, so the target application can learn a speaker adaptation that reasonably interpolates between training speakers. Two conditions must be met: 1) the number of training speakers must be reasonably large, and 2) the parameters making up the SVC must be relatively few, limiting the dimension of the space that must be filled.

Introspection about the experience of listening to voices suggests that it should be possible to find a relatively small number of parameters that represent voice differences with adequate fidelity. Since representation vectors derived from speech units by signal processing tend to be rather long, statistical and neural net techniques will be used to find a lower dimensional subspace onto which these phone based vectors can be projected, while still

[1] Although for all the experiments reported here, an oracle in the form of a pre-labelled database was used.

retaining the important voice quality information. First, descriptions for speakers vocal quality when producing individual speech units (the phoneme /IY/ for example) will be found, and then these phone descriptions will be combined into an overall speaker model. For consistency, these reduced segmental representations will be called Phone Pronunciation Codes, or PPCs. The next chapter (3) will describe the experiments that were done to explore and evaluate different kinds of PPC.

The final stage in building the speaker model is to combine the models of pronunciation of individual symbols into an overall voice model. This model is intended to capture the regularities that exist in relationships between the pronunciations of different phones by a single speaker, for example, the way the pronunciation of the phone /ix/ varies with the pronunciation of /ah/. This model will be relied upon to predict the sounds of unheard speech units from the sounds of previous heard units from the same speaker. Like the symbol codes (the PPCs), this speaker model should produce SVCs with as low a dimension as possible. Its derivation and use should also be robust in the face of missing inputs, since it is not possible to rely on having a complete speech symbol inventory for a speaker during training, let alone from the short segments of speech the speaker modelling system should be able to make use of when applied. Again connectionist and neural net techniques will be used to build a variety of candidate models. These experiments are described in Chapter 4.

2.3. A small example model

The requirements described above define a class of voice models. Before the discussion continues, in later chapters, to cover a comparison of some of the members of that class, the next few pages will be used to describe a particular instance from this class. This should serve to illustrate each of the steps involved, and provide a framework for the comparison between models in later chapters. In this illustrative example, attention will be paid to the degree to which the model satisfies the criteria for the utility of models that have been have set above, mainly with an eye to comparison with later models.

As is the case with nearly all of the work described in this thesis, the model was trained and tested using the TIMIT acoustic phonetic corpus [fisher86, lamel86]. This corpus is described in some detail near the beginning of the next chapter (§3.1), but for this illustration it should be sufficient to note only that it contains about thirty seconds of phonetically labelled speech from each of 630 speakers from eight "regions" of the United States. For this model speech from only regions 1, 2 and 3 (New England, Northern, North Midland) was used. For training, the speakers from the "train" subset of the database were used, and for these speakers, only speech from the 5 "sx" (phonetically compact[2] sentences per speaker was used. There were 190 speakers in the training subset used, and a total of 950 utterances.

For the sake of speed in training this model, use was only made speech from the following ten phonemes,[3] which occurred most frequently in the studied section of the database: /ix/, /s/, /n/, /iy/, /tcl/, /r/, /l/, /kcl/, /dcl/, /k/.

[2] These sentences were designed to provide good coverage of pairs of phones, and to include extra occurrences of phonetic contexts thought to be difficult or of particular interest by the corpus designers.

[3] A guide to the representation used in this document for phones and phoneme is contained in Appendix A.

2.3.1. Signal Processing

Using the phoneme level labels available for all sentences in the TIMIT database, each example of any of the selected phones was excerpted from the speech recordings in the database. The digital recording of the excerpted phone was zero padded to ensure that it was a multiple of 128 samples long, and an FFT power spectrum computed on non-overlapping 128-sample frames, yielding 64 power values per frame. These variable length collections of frames were used to build phoneme models. Further details on these signal processing steps can be found in §3.3 to §3.5, to which a reader unfamiliar with speech processing techniques may wish to refer.

2.3.2. Phoneme models

All FFT spectral frames for each phone were warped, using a linear distortion (§3.5), to a constant five frame duration. Each of the phones (which will be indexed with i) was, at this stage, represented by a vector $\vec{p}_i$ of 320 spectral coefficients. This fixed-length real-number representation is suitable for further processing by connectionist networks or multivariate statistical analysis.

As an indication of the amount of training data used for building these phone models, the phoneme /kcl/, one of the less frequent of those selected, was represented by 1058 of these 320 element vectors, or an average of 5.6 occurrences per training speaker.

As noted above, it is desirable for phone models to be compact, consistent within a speaker, and distinct across speakers (who are indexed here by j). The technique from multivariate statistics that is generally used to achieve these goals is Canonical Discriminant Analysis (CDA) [james85, dennis91]. Although it will be described more thoroughly in the next chapter, briefly, CDA involves computing the eigenvectors of the ratio of the between groups[4] and within groups[5] covariance matrices. The data is then projected onto the eigenvectors corresponding to the largest eigenvalues of the ratio matrix. Putting aside the technical description, the important thing to note is that this projection turns out to be the linear transformation of the original data that maximises the discriminability of the groups, while resulting in vectors having the desired dimension. This analysis was applied separately to each of the ten phonemes. In this application, since the aim is to distinguish pronunciations of a phoneme from different speakers, the goal of discrimination is to separate phone instances into groups by speaker identity.

A modified[6] version of the Aspirin CDA software [dennis91] was used to compute canonical discriminants for speakers over the training vectors for each phone, and the data were then projected onto the first eight of these discriminants. After this projection, each phone pronunciation was represented by those eight linear combinations of the original 320 components in the $\vec{p}_i$ vectors that maximally distinguished between speakers. The choice to

[4] The within groups covariance matrix is the covariance matrix of sample vectors after group (speaker) mean vectors have been subtracted, and measures the variability of samples of a phone within a single speaker.

[5] The between groups covariance matrix is the covariance matrix of group (speaker) mean vectors, and measures the amount of dispersion of speakers with respect to a given phone.

[6] The modifications were simply made to increase the size of the data that could be handled, and to provide more flexibility in the way results were output. No substantial algorithmic changes were made.

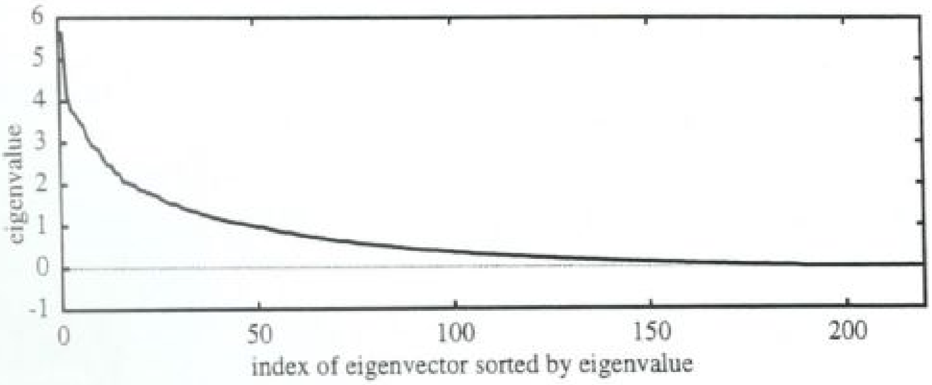


Figure 2: Eigenvalues for the phoneme /iy/, sorted by size. Since they fall off smoothly, there is no obvious choice of the dimensionality at which the important variation in the phone has been accounted for. A projection onto eight eigenvectors was chosen for the phoneme models, to keep the representation reasonably compact.

make the dimension of these PPC vectors eight was taken rather arbitrarily; there are 189 eigenvectors[7] resulting from the analysis and any set of the first *n* of them, ranked by eigenvalue, could have been chosen. As Figure 2 (for the phoneme /iy/) shows, the eigenvalues fall off quite smoothly, offering no strong guide to where the useful variation in the phone has been accounted for in a projection. The chief reason for the decision to use only eight dimensions to represent each phoneme was a desire to limit the size of the input to the speaker modelling phase. Improvements in the time warping used could reduce a source of irrelevant variation in the phone vectors, and make the eigenvalues fall off more sharply.

Characteristics of the Phoneme Codes

To show what the PPCs look like, and to demonstrate that they are at least somewhat consistent with a speaker, PPCs were extracted for the phone /iy/ for the three male and two female training set speakers who used the phone at least six times.[8] Since these speakers used the phone different numbers of times, the number *m* of PPCs for each speaker differed. These PPCs were then divided into two groups for each training speaker, derived from the first *m/2* utterances of /iy/ from the speaker, and the second *m/2*, respectively. The plots in Figure 3 are designed to give a graphical representation of first and second groups of PPCs from each speaker. Since these are two dimensional plots, only the first two components of the PPC are used. There is considerable variation amongst the PPCs from a given speaker, as might be expected, considering the number of effects on the PPC that could not be controlled for. Nevertheless, relationships among the speakers and position are largely preserved. Despite the variation in the pronunciation of the phone within each speaker, maximising the discriminant function has located the speakers within a subspace of speaker space for the phone, and it is this information that will be combined with similar information about other phones to form the speaker model. Plots for the other phones have similar characteristics.

[7] The number of groups (i.e. speakers) less one.

[8] The fact that there were only this many speakers with enough data to compute variances for halves of the phone instances is surely a sign that the TIMIT database is less than ideal for this sort of research.

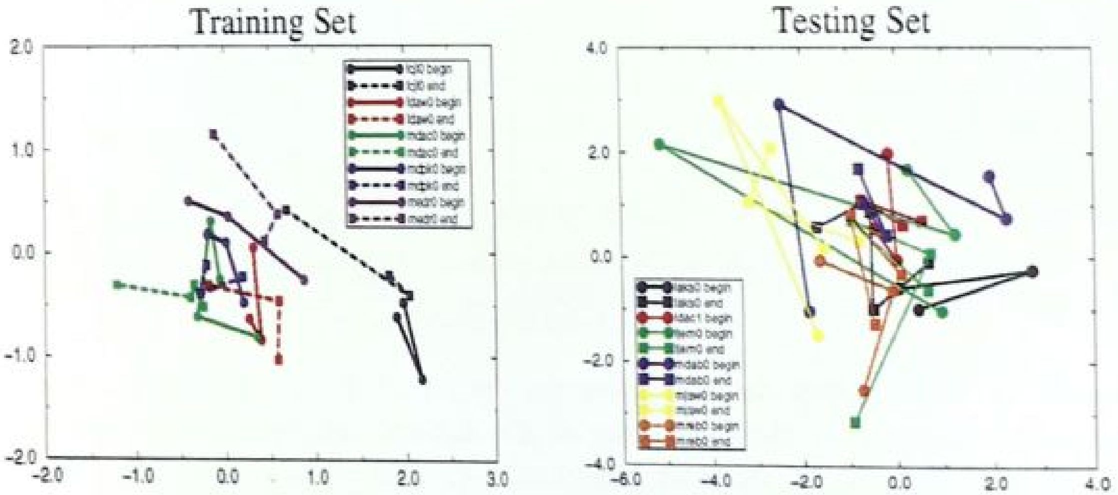


Figure 3: The first two components of the Models of /ix/ pronunciation for first and second half of the set /iy/ pronunciations available from each of a subset of speakers. While the speakers could not easily be identified on the basis of these phone codes alone, the codes from a given speaker are clearly located in nearby regions of the speaker space for /iy/. This position will be combined with evidence from other phones to form the speaker code. The within speaker variation of the speakers from the testing set is even greater, but there is still visible clustering of points by speakers.

Another measure of code quality.

Table 1 gives the values of a discriminant measure designed to measure the relative dis-

Phone	/dcl/	/ix/	/iy/	/k/	/kcl/	/l/	/n/	/r/	/s/	/tcl/
J	8.85	3.55	5.50	5.92	6.12	5.26	4.49	6.14	4.36	4.31

Table 1: Discriminant measure on PPCs for small example phone models. A description of the measure is given in the accompanying text; it is based on the relative separation of PPCs for different speakers when the ratio of this separation to that of PPCs within a speaker has been maximised.

persion of groups — in this case of speakers — in the transformed space that the CDA produces and in which the PPCs lie. This measure, the square root of the trace[9] of the ratio matrix referred to above, will be discussed in more detail in the next chapter;[10] for now it will suffice to note that larger values of the measure indicate better clustering of PPCs within a speaker. The PPC for the phone /iy/, for which phone models were shown in Fig. 3, is in about the middle of the range, the other phones provide a similar amount of information to distinguish speakers. It is also interesting and somewhat surprising to note that, as far as this model is concerned, consonants are as good for discriminating speakers as vowels are.

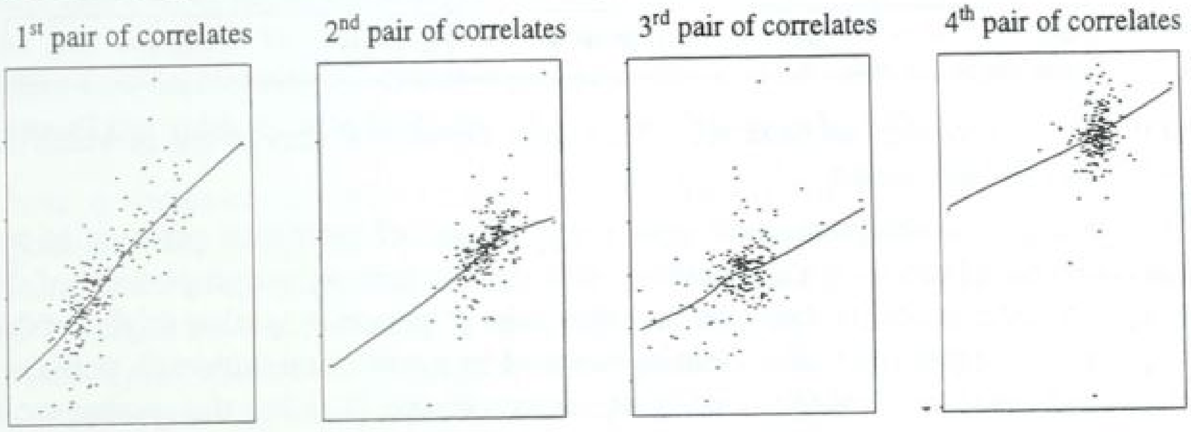
[9] the trace of a matrix, the sum of its diagonal elements, is equal to the sum of its eigenvalues, its square root is similar to the diameter of a sphere with a similar volume to the space containing the projected vectors.

[10] In section 3.7 on page 53.

2.3.3. Speaker model

Correlations between phone models

Now that the PPCs for the example have been found, it is possible to proceed to combining them into a speaker model. Before doing so, though, it is worth verifying that they have the qualities that are needed. The previous paragraph showed that the phone models locate a speaker within their particular space with some exploitable stability. If these PPCs are to be useful in a speaker model, however, they must bear relationships to each other that can be exploited to make predictions about unheard phones from heard ones. If it is possible to predict the PPC for one phone from that of another in a pairwise fashion, then it is reasonable to expect that an underlying variable, the SVC, can be found that enable one to make all such predictions simultaneously. To demonstrate this prediction, mean PPCs for each phoneme were calculated for each speaker,[11]and canonical correlation analysis [becker88,manley86] applied between pairs of these means, across speakers. This analysis finds a set of pairs of linear combinations of the components of the PPCs, such that the first pair has the highest possible correlation, the second has the highest correlation among variables uncorrelated with the first pair, and so on. Figure 4 shows the first four of these eight pairs, for the phones /ix/ and /iy/ from the training set plotted with the appropriate projection for /ix/ on the ordinate, and for /iy/ on the abscissa[12]. The values of the correlation coefficients, r, for these four pairs are r_1=0.798, r_2=0.636, r_3=0.486 and r_4=0.287 respectively. Using the Bartlett test given below, from [manley86], it is possible to calculate the probability that each of these measured correlations r_j is greater than or equal to its given value, under the null hypothesis that there is no correlation between the /ix/ and /iy/ vectors (n is 190, the number of speakers, and p, q, and r, the width of the /ix/, /iy/ and correlate vectors, respectively, are equal to



1st pair of correlates 2nd pair of correlates 3rd pair of correlates 4th pair of correlates

Figure 4: Scatter plots of the first four canonical correlates between /ix/ and /iy/ for the training speech. These correlates are pairs of projections of the PPCs that have maximal correlation, such that successive such pairs are orthogonal. The projection for /ix/ is given on the x-axis, and for /iy/ on the y-axis. The lines through the data are locally linear fits using the loess method, and are included to give an impression of the degree of correlation and how linear the fit is.

[11] overall means for the phoneme were used when the speaker did not utter a phone.

[12] The graph was produced using the SPlus scatter.smooth function. Interested readers are referred to the documentation for that program, and to [chambers93].

$$\left(\phi_j^2 = -\left(n - \frac{1}{2}(p+q+1)\right)\sum_{i=j}^{r} \ln(1-r_i^2)\right) \sim \chi^2(p-j, q-j)$$

eight). The results of this analysis are laid out in Table 2; the first three correlates are highly significant, suggesting that there are at least three orthogonal dimensions in which the PPCs for /ix/ and /iy/ are related.

Table 2: Significance tests for the first four canonical correlates between the /ix/ and /iy/ vectors across speakers using test due to Bartlett. The first three correlations are highly significant, and are almost certainly not due to chance.

index (*j*)	correlation	ϕ_j^2	$pr(\chi^2(p-j, q-j) \ge \phi_j^2)$
1	0.798	363.45	$<< 0.0001$
2	0.636	179.37	$<< 0.0001$
3	0.486	85.42	$<< 0.0001$
4	0.287	36.497	> 0.06

The assumption behind speaker modelling on basis of segmental pronunciation — the approach taken here — is that there are similar relations, although perhaps of different strength or dimensionality, between the other pairs of phones a speaker might produce, and that, moreover, these correlations can be captured by a projection (although, perhaps, a non-linear projection) onto a single underlying vector variable. Building the speaker model consists of finding this projection of the phone pronunciation codes described above onto the single underlying variable, or Speaker Voice Code (SVC).

Training the speaker model

In this example of speaker modelling, instead of having the system explicitly model the correlations by trying to produce predictions of the values of PPCs from a subset of them,[13] it is assumed that the predictions made by the PPCs are useful only in as much as that they serve to distinguish speakers. The SVC will be generated using a non-linear projection of the PPCs onto a single vector. The projection will be found by optimisation performed with the goal of learning this speaker distinction. The model of speaker variation, in this case, consists of the neural net shown in Figure 5. This network attempts to do a non-linear discrimination between speakers, using the information available from a subset of the ten chosen phones.

During training, phones codes were randomly chosen from those available for the target training speaker so that, on average, 5.75 of the ten sets of eight-component phone inputs had PPC data on them. Inputs for which a PPC had not been chosen were set the overall mean of the appropriate PPC. A variety of widths was used for the bottleneck layer in which the speaker model was formed, producing a set of speaker space models of differing dimen-

[13] Although that too will be done, in a later chapter.

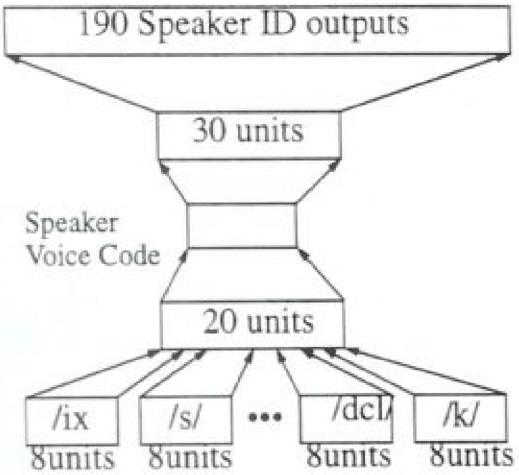


Figure 5: **A connectionist discriminator that can be trained to form a low-dimensional speaker model. Phone codes from a speaker are placed on the appropriate sets of input units to the network, and the network is trained to output his or her identity. In doing so, the network produces a vector of hidden-unit activities in a bottleneck layer, and these activities can be used a speaker code. The purpose of the network is to form the SVC in the bottleneck layer.**

sion. Training was carried out with a learning rate of 0.001 and momentum of 0.5 for a thousand epochs. At the completion of training, for the network with a two unit bottleneck, approximately 3% of the generated training input patterns resulted in correct speaker identification. It should be noted that this is an extremely difficult task, there is a great deal of noise in the input patterns, and to make it easy to plot, the network is trying to encode the speaker information into a very narrow, two unit bottleneck. Despite this, its classification performance is almost six times greater than the rate that would be expected by chance.

After training, the SVCs were generated (again for the training speakers in this example) by using input in the order that it had appeared in the original speech. For each speaker, all of the banks of eight phone code units were initialised with zeros. As phone codes were read from the file, they were placed on the appropriate input bank, replacing whichever value had previously been there.

The hidden unit activities, constituting the SVC are plotted in Figure 6 for the two unit bottleneck. SVCs for four female and four male speakers are shown at four points in time: after each of the first 15 phones had been placed on the input, after phones 11-25, phones 21 to 35, and after phones 31-45. Over time, the speaker codes move towards final position in the speaker space, and increase their within speaker clustering and between speaker separation. Not all speakers have easily identified positions; speaker *fjcf0*, for example, appears to vacillate between two positions. A more detailed analysis of the speaker codes, including their degree of clustering, and the time course of their formation, is deferred to chapter 4.

2.4. Summary

The example illustrates that, given an adequate division and classification of speech into segments, it is possible to construct models of the segments that are consistent within speakers, that differ between speakers, and that predict each other's values. To take advantage of

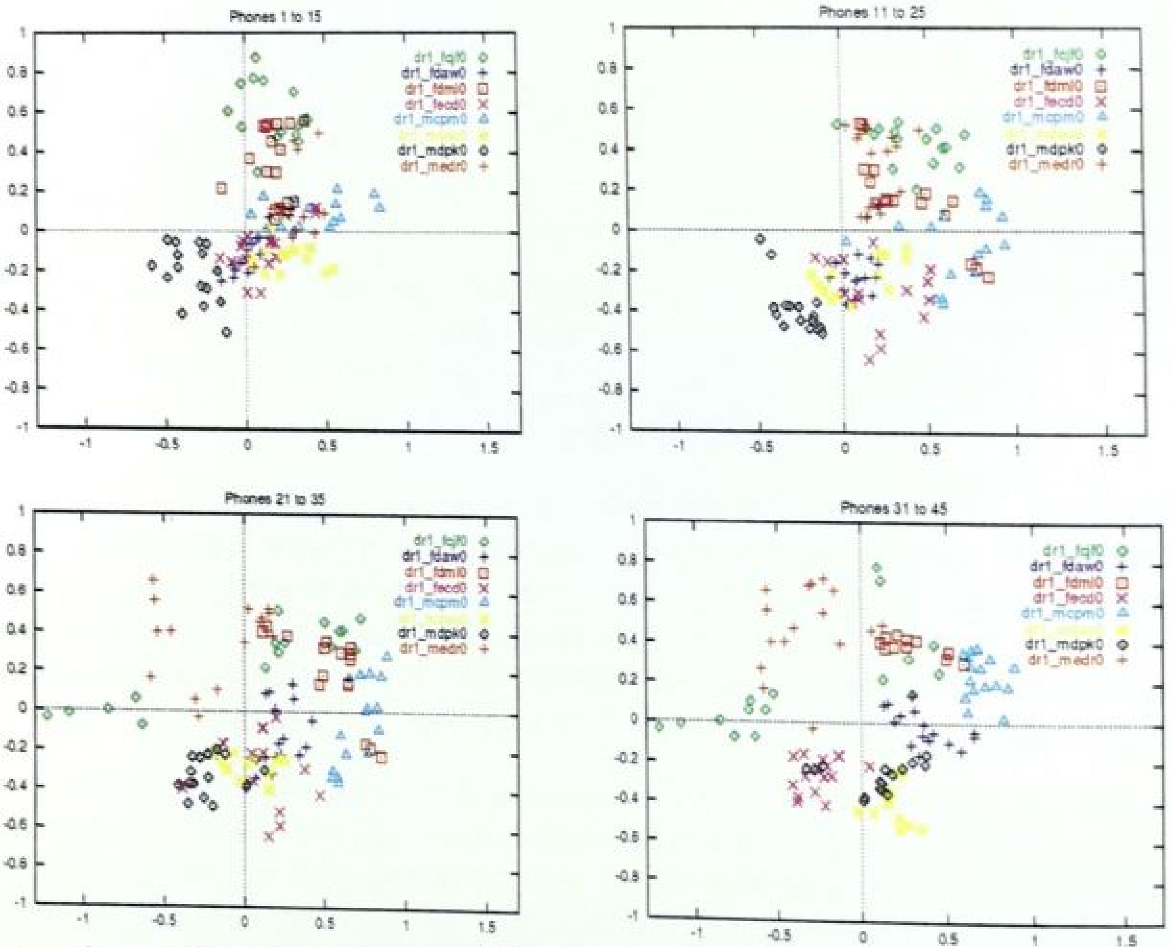


Figure 6: Evolution of speaker models. The four graphs represent speaker codes formed after the addition of phones 1-15, 11-25, 21-35 and 31-45 of those spoken by the first' four female and male training speakers. While there is considerable variation within speakers, and overlap between speakers, there is a clear distinction by sex, and clear differentiation between speakers. Over time, the codes become more separated between speakers and more tightly clustered within speakers.

this ability to predict the PPC of one segment from that of another, speaker models can be built, designed to capture the relations between segmental pronunciations. The representation that one of these speaker models forms of the set of segments it has heard from a speaker, at some given time, constitutes at least a partial characterisation of the speaker's voice — a Speaker Voice Code.

Of course, there is more than one way to derive a PPC from a speech signal, and more than one way to combine these PPCs in an SVC. The following two chapters will explore these alternatives for the PPC and SVC respectively, after which two speech tasks will be used to examine the utility of this partial characterisation of voice quality.

Chapter 3. Variation within Phones

The previous chapter contained an overview of the form the speaker models would take, and illustrated that form with a particular instance of such a model. The next two chapters will describe possible variants of that general form in detail, and describe the experimental work that was done to select among them. The current chapter will concentrate on the construction of models of the variation in individual speech units, and the following chapter will cover the combination of the outputs of these segmental models into overall representations of a speaker's voice.

3.1. Database

Good data for speaker modelling work is not easy to obtain. Since it is desirable to attain reasonably good coverage of the space of speakers, speech samples are needed from a large number of people. Since speaker models are being built up out of models of the variability within phones, it is necessary to estimate both how pronunciation of each phone varies between speakers, and how it varies within a speaker. To accurately estimate the latter, within speaker variability, it would be useful to have available an amount of speech from each speaker sufficient to contain several examples of each of the phones composing the model. Unfortunately, it was not practical to gather such a huge, specialised database solely for the purpose of supporting this thesis work. Moreover, even with the considerable computational resources available to the CMU Neural Network Speech Group, many of the experiments described here would have been computationally infeasible if done on a larger database.

With these constraints in mind, a subset of the data in the TIMIT (Texas Instruments/Massachusetts Institute of Technology) database [fisher86,lamel86] was chosen as the training and test set for the speaker and phone models. This database contains recordings of 6 300 sentences, ten sentences uttered by each of 630 speakers from eight dialect regions in the United States. Because of computational and storage constraints, the experiments in this thesis used data only from speakers who had been raised in the first three of these regions, labelled dr_1 (New England), dr_2 (Northern) and dr_3 (North Midland) in the database. These groups contained, respectively, 49, 102 and 102 speakers, of whom 18, 31 and 23 were women. Of the ten sentences spoken by each speaker, two were "dialect sentences" designed to highlight dialectical variation, these sentences, sa_1 and sa_2 were identical for all speakers. Three of the sentences were "diverse" sentences, selected from two existing corpora with the aim of maximising the range of "allophonic" contexts of the phonemes used. These sentences (si_n) were different for every speaker. The remaining five "compact" sentences (sx_n) from each speaker were each spoken by a total of seven speakers, and were designed to give good diphone coverage, with a concentration on contexts thought to be difficult, or of particular interest, by the database designers.

The material on the database CDROM is divided into training and test directories, and this division was used in the reported experiments,[1] rather that the division suggested in the doc-

[1] This was done merely for convenience; the recommended division should be used for further experiments.

umentation contained on the disc. During training, the $sa_{1,2}$ sentences were also excluded, so that they could be used as reference material in a comparison of the results of voice conversions performed using speaker models from different speakers applied to the same text.

3.2. Symbol set

If every speaker could be constrained to utter the same utterance, over the same duration, the voice modelling task would be relatively straightforward. A fixed set of these utterances could be collected from a great many speakers, and a model of the variation in them, of chosen dimension, could be estimated by a technique such as principal components analysis or compression in a bottleneck network.

In fact, for rapid, natural adaptation, no constraints can be placed on the speech uttered. Instead, a set of speech units must be chosen within which voice variation can be modelled, and into which the speech can be divided for analysis. To the extent possible, the realisation of these units should be constant within a speaker but vary between speakers. Additionally, the units must occur sufficiently frequently in speech to make them useful for modelling; learning to extract information from a unit of speech is of little use to a system if that unit is almost never used by speakers. This choice of a set of symbols for modelling depends on satisfying two, mutually antagonistic goals:

1. It is desirable to minimise the number of symbols used to describe the meaning of the speech signal, so that there will be enough samples of each symbol available to provide a reasonably dense coverage of the space in which it varies. This is essential to producing a model that makes useful predictions, since otherwise it will not be possible to obtain reliable estimates for the parameters of a phone model
2. Since the speech associated with each of the symbols is to be used only to model speaker differences, it is desirable to minimise the amount of the variation in the instances of each symbol that is unrelated to speaker characteristics. In particular, it is necessary to minimise the influence of phonetic and contextual variation.

A natural unit to choose as a symbol is the phoneme, since it provides a balance between frequency and consistency within speakers. It is also attractive, since it is generally the unit of recognition or synthesis. Information extracted by modelling differences in its pronunciation is likely to be useful to a phoneme based recognition or synthesis task, and the phoneme units required should be reasonably easy to extract from utterances, since the identification of these units is often a component of the target application.

Of course, there are disadvantages in the choice of phonemes as a basis of modelling — chiefly the amount of acoustic variation in phones due to context. Since much of this variation is due to immediate phonetic context, a large source of non-speaker-related variation could be eliminated if it were possible to use triphones[2] as the modelling unit. Unfortunately the number of triphones is so large, and the available resources of training data, computation and storage space so limited, that this sort of modelling is not tractable. Although the technique was not applied to the main body of experimental work reported here, an initial

[2] Triphones, also called PICs (phones in context) are units consisting of the realisation of a phoneme in the context of a particular preceding and succeeding phoneme.

approach to the problem to reducing allophone variation, without modelling triphones separately, was made, and will be described in §3.11.

In most of the experiments reported, the only source of phonetic variation that was controlled for was the difference between phones, and this control was achieved by separately modelling variation within each of a set of phones. Since uncommon phones are, in general, unlikely to be particularly productive in predicting future speech, since they are not usually available, and since this scarcity also makes it hard to get reliable estimates for the parameters on which they vary, infrequent phones were not modelled. The choice of which phones to use and which to omit was made with the assistance of the data shown in Figure 7, a graph of the frequencies of phones used by the speakers in the first three geographic regions (dr1, dr2 and dr3) of the timit training set, sorted by frequency. The great majority of the phones in the database are covered by the first thirty of these phones, and it is these phones that were used to build models.

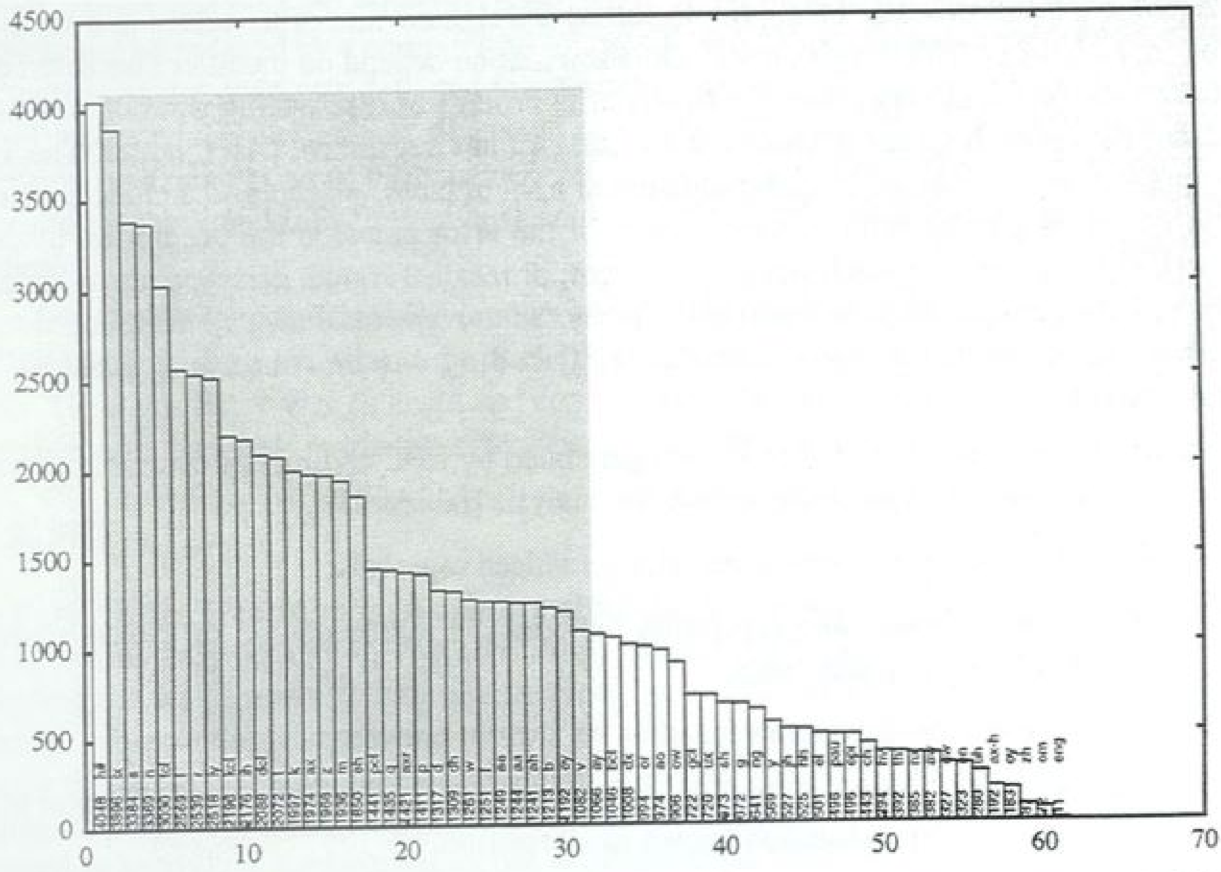


Figure 7: Frequency distribution for phones in speech from speakers frm the TIMIT *$dr_{1,2,3}$* regions, excluding *sa* sentences, for all speakers. The shaded region indicates the thirty frequent phones used for the main experiments. These account for 78.3% of the phone occurrences.

3.3. Analysis method

Once the set of symbols had been chosen, a representation of the raw speech signal was selected for use as a starting point for building the phone models. The two candidate representations were suggested by the target applications. Speech recognition systems typically use a spectral representation of the speech signal, mimicking the signal analysis done by the basilar membrane in the human ear, so it was natural to consider building the model with FFT filterbank coefficients. Voice Transformation is performed, in the system described later in this thesis, in a representation consisting mainly of LPC log area ratio coefficients, which are related to a speaker's vocal tract dimensions, so it was also natural to consider using that representation for building speaker models. Both these representations are briefly described below, followed by an experiment that was done to compare their suitability for voice modelling.

3.3.1. LPC log area ratios

LPC (linear prediction coefficient) coding is derived from the observation that the speech signal at a given time can be approximated by a weighted sum of its values at a small number of past times. The weights used in this summation depend on the filter characteristics of the vocal tract, and vary relatively slowly. The process of discovering a set of weights that describe a speech signal is known as building an autoregressive (AR) model. The speech signal can be represented by a combination of a set of these weights and a crude approximation, such as a pulse train or white noise, of the error between the prediction of the AR model and the actual speech signal. This error, or residual signal, corresponds to the excitation signal generated by the vocal cords or by the movement of air past obstructions in the throat and mouth. Far more detail on LPC coding can be found in [rabiner93] and [markel76].

The compact representation of speech generated by LPC coding has desirable properties as a representation of speech for speech for analysis [rabiner93]:

- It models the speech well, especially for voiced segments.
- It provides a reasonable separation between the representation of glottal source and vocal tract characteristics.
- It is computationally tractable, and
- It has tended to work well in recognition applications, usually after conversion to a quasi-spectral representation known as a cepstrum.

The LPC representation also has the very desirable property, from the point of view of speech synthesis, that it is very straightforward to reconstruct a good quality speech signal from the LPC representation

The raw LPC representation of speech is not ideal for speaker modelling purposes. These models require learning to place speech from different speakers at different points within a space. During training, the modelled position and the desired position of the speech in this space must be compared using a distance metric. With raw LPC "reflection" coefficients it is not appropriate to use the sort of Euclidean distance measures appropriate for neural network training [rabiner93]. Fortunately, it is straightforward to convert the reflection coefficients generated by LPC analysis into a more appropriate representation.

The vocal tract can be viewed, approximately, as a concatenation of p fixed length cylindrical sections of cross-sectional areas $A_i, (i = 1, 2, ..., p)$. Starting from reflection coefficients, one can calculate coefficients that are each the log of the ratio between the areas of successive cylindrical sections, starting from the lips, i.e. $l_i = \log(A_{i+1}/A_i)$. These Log Area Ratio (LAR) coefficients have two desirable qualities from the point of view of speaker modelling: sensible Euclidean distances can be calculated on them [rabiner93], and they represent directly a fairly good [kuc87] approximation to the vocal tract shape differences between speakers that the models are trying to capture.

While linear predictive encoding is conceptually straightforward, producing a reliable LPC encoder/decoder is not an entirely simple matter. Despite some misgivings about its quality, a decision was made to use the freely distributable version of the government standard LPC-10 coder [tremain82]. This coder represents speech as a series of 22.5ms frames, each of which consists of a pitch value, a RMS power value, two boolean voicing decisions for half frames, and ten LPC reflection coefficients. In normal operation, bandwidth requirements would be reduced by a complicated bit encoding scheme detailed in the reference, but this section of the code was defeated for the experiments described in this thesis, and the frames produced were exactly as described above. For the majority of the experiments the reflection coefficients were transformed into the Log Area Ratio (LAR) representation before further use, and transformed back into reflection coefficients for the purposes of resynthesis.

Readers intending to use the LPC encoder should note that it introduces a two frame delay in the speech stream, and that it drops the last two frames, making the delay difficult to detect. Until this delay was detected and compensated for[3], aligning the frames the encoder produced with the labels in the timit database had not been successful.

3.3.2. FFT

The Fast Fourier Transform is an efficient algorithm for computing the Fourier decomposition of a time varying signal. The algorithm, which is explained in detail in many places including the well known "Numerical Recipes" [press88], takes an array of floating point numbers representing a signal and returns an array of complex numbers representing that specify the phase and amplitude of a set of sine waves. These sine waves, at equally spaced frequencies between 0 Hz and half the sampling frequency of the original signal, can be summed to reproduce the original signal, or, more importantly for the present purpose, the amplitude at each of these frequencies can be extracted to provide the power spectrum of the signal. This representation, which is computed in approximation both by the basilar membrane of the inner ear, and, it seems, by the auditory cortex, displays useful information about a speech signal. Notably, this information includes the spectral peaks, or formants, in the signal resulting from vocal tract resonances controlled by the position of articulators during vowel production, and the shape of the filter applied to the noise of turbulent airflow during consonant production.

Since in speech processing, one is interested in the time course of the speech signal, and not just its overall spectral characteristics, the FFT is computed on fixed duration "windows" on the speech signal. The duration of the windows is chosen to be short enough to

[3] By finding a minimum in the alignment error between encoded and unencoded speech, a process that the author feels compelled to mention not because it is particularly interesting, but simply because of the effort it involved.

capture important changes in the speech signal while still providing enough frequency resolution too reveal important spectral peaks and troughs. Typically, and in the work reported here, the frames contain 128 samples from a signal sampled at 16kHz, giving a time resolution of 8ms and a frequency resolution of 125Hz. The resulting series of power spectrum vectors are referred to as "frames" of speech.

It is common practice to smooth the signal by using overlapping windows, applying a windowing function to the signal to reduce boundary effects, and by reducing the dimension of the frames by combining contiguous frequencies into a smaller set of logarithmically spaced bins. In the work reported here, however, the power spectrum of the raw signal was typically used directly.

Figure 11 shows an example of an spectrogram computed from the utterance sa_1, *"She had your dark suit in greasy wash-water all year."*, spoken by a male speaker (mpgh0) from the New England (dr_1) region of the TIMIT data-set.

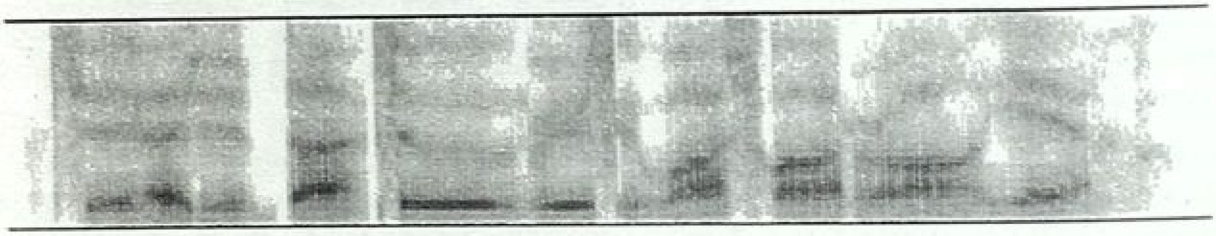


Figure 8: An example spectrogram generated using the Fast Fourier Transform (FFT). The sentence *"She had your dark suit in greasy wash-water all year."* was spoken by a male speaker.

3.4. Experiment: Choice of Analysis Method

Although both the LPC and FFT methods of extracting spectral information had been widely used in speech research, it was not known which, if either, was a better representation to use for modelling speaker differences. In this experiment, phone models built using both were compared for speaker discriminability.

3.4.1. Materials

The models compared were produced from speech from the ten phones /dcl/, /ix/, /iy/, /k/, /kcl/, /l/, /n/, /r/, /s/ and /tcl/, represented using both LPC LAR coefficients and FFT filterbank coefficients. In each case, the model was "trained" on the '*sx*' sentences for "training" speakers from dr_1, dr_2 and dr_3, and "tested" on the '*si*' train, '*si*' test, and '*sx*' test sentences for speakers from the same region. This testing data included speech from the same speakers used to develop the phone model, and from different speakers, to permit measurement of the degree of overfitting, if any, in the model.

In the case of LPC coefficients, the phone boundaries available in the TIMIT database were used to excerpt frames directly from precomputed LPC-LAR versions of the speech files. The LAR coefficients were used directly, and the two voicing decisions were multiplied by 1.0, and the pitch and gain by 0.01 and 0.005 respectively, to convert them to floating point numbers. The frames making up each instance of a phone were then warped to a fixed length of ten frames using the linear warp described below, yielding a 140 element vector per phone.

If the case of FFT coefficients, the raw speech from the TIMIT database was read in, and converted to floating point values in the range [-1:1]. The speech for the target phone was excerpted, and FFTs calculated on non-overlapping 128 sample segments. The final segment was zero padded to 128 samples, if necessary. The FFT analysis yielded frames of 64 power values for each segment. The frames for each phone instance were linearly warped to a fixed set of five frames, yielding a 320 element vector per phone.

3.4.2. Procedure

All vectors from both encodings and for the four data-sets (sx_train, si_train, sx_test, si_test) for each phone were projected on to the first eight principal components for the training data, yielding eight unit phone codes. The discriminability measure J, described in section 3.7, was calculated for each phone and for each of the eight conditions.

3.4.3. Results

Table 3 shows the mean value across phones, and the variance, of the discriminant measure for the two representations, for both training and test speakers. This measure describes how closely clustered phone codes within a speaker are, compared to the spread of the phone codes between speakers. In all conditions the LPC-LAR representation produced slightly more tightly clustered phone models than the FFT representation, but the difference was clearly not statistically significant, since the differences between means are close to a single standard deviation.

Table 3: Discrimination measure J calculated for two candidate input representations to phone modelling. The phone models in question are generated by PCA. The shaded entries are calculated from phone instances used in training.

Representation	Speakers	sx	si
		mean / s.d.	mean / s.d.
LPC-LAR	train	2.05 / 0.43	2.36 / 0.40
	test	2.06 / 0.35	2.35 / 0.35
FFT	train	1.76 / 0.27	2.17 / 0.39
	test	1.79 / 0.35	2.28 / 0.46

3.4.4. Conclusion

There was no strong reason to choose one representation over another, although there was a nonsignificant tendency for the LPC to perform better. In the end, the FFT representation was chosen for use in further phone modelling experiments, in part because this representation is a more familiar one to workers in speech, and because the spectral representations it produces are more readable.

3.5. Fixed Length Phonemes

Phonemes present a problem as far as modelling is concerned. They occupy a variable duration. Moreover, the relative starting times within the phoneme of the acoustic states that make them up vary between different instances if the same phoneme. If the aim were simply to try to model vocal tract characteristics, this temporal variability would need to be eliminated, as far as possible, from the modelling process. For the most part, in this thesis, it is acceptable to accommodate this variability, since the dynamic aspects of speech production are also important to voice quality. In this latter case, the linear warp described below suffices to put the speech into a form suitable for model building. However, in recognition of the fact that for some potential applications of speaker modelling, particularly applications to recognition, reducing this variation will help highlight relevant speaker differences, an algorithm for doing so will also be described.

3.5.1. Linear warping

The most straightforward method of fixing the length of FFT analysed phonemes for further analysis is the linear time warp. This technique is rather straightforward; if the length of the sound is s and the target length t, then the phone is divided into t sections s/t frames long. One can either take a "representative" frame from the middle of each section, in an attempt to reduce spectral smear, or produce the new frame by calculating the mean vector across the section, with frames from section borders being appropriately weighted. This process is described in figure 9. In cases where the input sound sample is shorter than the target dura-

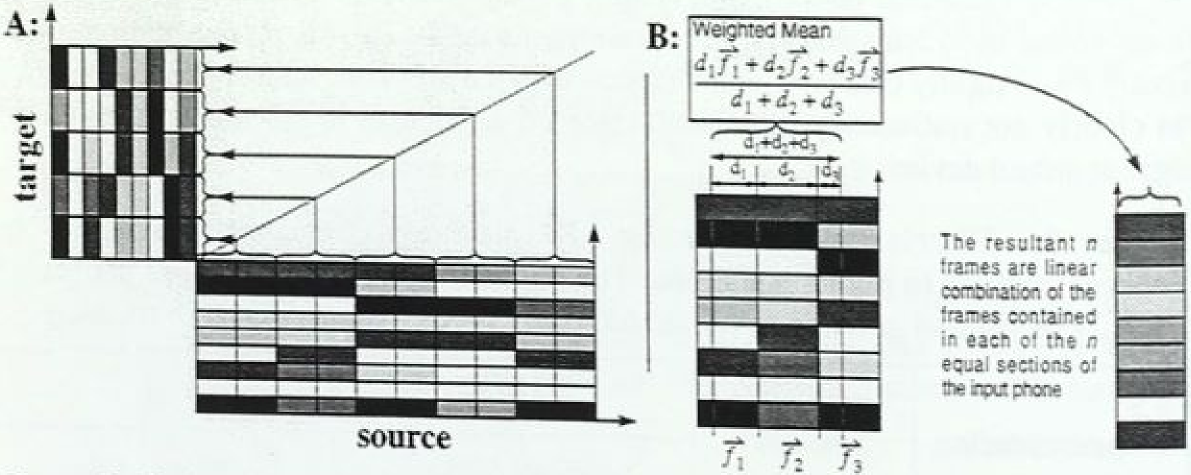

A:
target
source
B:
Weighted Mean
$\frac{d_1\vec{f_1}+d_2\vec{f_2}+d_3\vec{f_3}}{d_1+d_2+d_3}$
$d_1+d_2+d_3$
d_1
d_2
d_3
$\vec{f_1}$
$\vec{f_2}$
$\vec{f_3}$
The resultant n frames are linear combination of the frames contained in each of the n equal sections of the input phone

Figure 9: Fixing the length of a phoneme by linear warping. Part A of the diagramme shows the general scheme: Multiple frames in the source spectrum are copied onto single frames in the target in the case where the source length is a multiple of the target. Where this is not the case, target frames are linear combinations of source frames, as shown in part B of the diagramme. If the source spectrogram is shorter than the target duration, its frames are replicated until it is not, and the process shown above is applied.

tion, the length is doubled by frame replication until this is no longer the case, and then the linear warp is done. If the target length is sufficiently long, the speech can be recovered with reasonable fidelity after inverting the time warp, providing an invertible encoding, such as LPC, has been used to produce the starting vectors. In the case of FFT power spectra, this inversion is more difficult, but can still be done in approximation [Alex Waibel, personal communication].

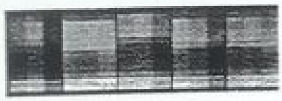
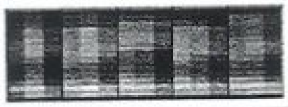
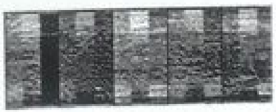
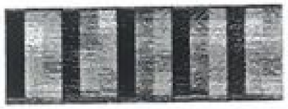
This linear warping technique results in fixed length vectors, but retains information about timing within the phones. This information may not be particularly useful if used to construct phone models used, for example, to adapt a recogniser based on frame labelling.

3.5.2. Iterative alignment

Although this technique was not used in later experiments, in part because it is much more time consuming than linear warping, some pilot work was done to develop an algorithm to time align the excerpted phones used as input to the phone models. If an accurate recogniser were available, of course, one could eliminate temporal distortions by using the states from the alignment path generated during recognition. Speech information from a given recognition state would be inserted into the appropriate frame of the fixed length input vector.

In the absence of such a recogniser, an alternative method of identifying acoustic states within excerpted phonemes needs to be used. One such method is to reduce temporal distortions using a variation on the *k-means* clustering algorithm (described, for example, in [rabiner93]). For each speech sound, a fixed number of templates are used. After initialisation, involving assigning every *k*th instance of a phone to the kth template, an iterative procedure is used. Each example of the speech sound is aligned by dynamic time warping (DTW) alignment [ney84] against each template, and is assigned to the template with the best alignment score. After all samples have been treated in this manner, the frames in the templates are replaced by the mean of the frames in the samples of speech that aligned to them. This procedure is applied repeatedly until the total Euclidean distance between the input phones and the templates they align to reaches a stable minimum.

Table 4: These are the five, three frame long, template vectors for two vowel phones /IY/ and/EY/ and two consonants, /S/ and /K/ for the RMSpell database, at the end of the iterative time alignment procedure has been performed.

Vowels: /IY/ and /EY/		
Consonants /S/ and /K/		

3.5.3. Warping Chosen

The end result of either of these alignment procedures is a set of fixed length vectors capturing the acoustical features of the speech unit. These fixed length vectors can be used to build models of variation in pronunciation of the units they represent.

To accentuate the variation amongst instances of the same unit, the overall mean vector for the unit can be subtracted from each sample, and the resulting vector can be divided by the standard deviation, yielding vectors with zero mean and unit variance. This makes no difference to the statistical modelling techniques described later, but this sort of normalisation is widely believed to speed neural net learning, since there is no need to learn an offset for the output units.

Although the iterative warping method probably produces cleaner spectral estimates for sections of a phone, if the phone is thought of as consisting of a sequence of spectral states, it is not clear that this is the most desirable thing to do when building general models of voice variation. An important part of voice quality may be contained in the relative durations of these states within a phone, and the alignment would lose this information[4]. For this reason, and for practical reasons of computational load, a linear warping to five frames was used for the major experiments that will be described in the rest of this document. After this linear warping had been done, each phone was represented by a 320 element vector.

3.6. Capturing the Variation

After the length of the phone exemplar has been fixed, and any methods designed to correct for context effects have been applied, the next step is to build a representation of the variation in phone exemplars that is as parsimonious as possible. It is this representation that will be used as input to the system that combines the descriptions of variation in individual phonemes into an overall model of speaker variation. Four such dimensionality reduction techniques, two statistical and two using neural-networks, were investigated, with the aim being to retain as information as possible from the phone, in a representation of the lowest possible dimensionality. Figure 10 outlines the techniques.

	Variational	Discriminant
Linear	Principal Components Analysis	Canonical Discriminant Analysis
Non-Linear	Bottleneck Neural Net Encoder	Bottleneck Neural Net Discriminator

Figure 10: Methods for reducing the dimensionality of data. Variational methods select the directions of maximum variation in the input data. Discriminant methods choose directions of maximum variation relevant to a classification task.

The "variational techniques" involve forming a reduced dimension representation of a set of data that aims simply to retain as much information as possible about the variation in the

[4] It might, however, have been the best choice or the actual test applications built, since in neither case was timing important to the system. However, improved voice transformers, for example, should adjust timing, and the model of voice should not discard this information without good reason.

data, whatever that variation might be. The linear variational technique, Principal Components Analysis (PCA) [james85, duda73, dennis91], finds a linear projection of the data onto a fixed, smaller, vector, that if inverted, most closely matches the original data set. Experiments with PCA are described in § 3.6.1. In the non-linear version of PCA [sarle94], a network with a bottleneck layer of the desired dimension is trained to match its outputs to its inputs. While it is not clear that there is any way to be certain that the encoding found is optimal, such networks can, as shown in § , exhibit impressive power when compared with linear methods. § 3.6.2 describes the application of these methods to the speaker modelling task.

Unlike the variational methods, which make no assumptions about the meaning of the data, discriminant methods assume that the data can be divided into relevant classes, and that, in fact, these classes are known at training time. These methods try to retain only variation that distinguishes the classes, and to discard that which distinguishes the members within each class. In the case of the experiments described here, the groups of observations to be distinguished are those coming from different speakers. The linear technique, Canonical Discriminant Analysis [james85, duda73, dennis91] (CDA)[5], does this by maximising the ratio of variation between groups to that within groups following the projection. Experiments using this technique are described in §3.6.3. The non-linear technique, again, uses a bottleneck network, but this time it is trained to label the speakers. Experiments with such a network are described in §3.6.4.

3.6.1. Statistical Dimensionality Reduction

If one is unbiased about which dimensions of variation in the original signal are important, principal components analysis gives the optimal linear projection of the original phone space onto a subspace of limited dimension. It retains as much of the information about the variability of the original space as can be retained in a linear subspace of the chosen dimension.

Since the phones are all same length at this stage, it is possible to use this standard technique to find a subspace describing their variation. This is done projecting the set of vectors on to an appropriately sized subset of the eigenvectors of their covariance matrix. By restricting this projection to the m eigenvectors with the largest eigenvalues, one chooses a set of m dimensions along which the original phone vectors vary the most. This is almost exactly what is wanted. This projection has the additional advantages of not being terribly expensive to compute and of being invertible. If the original analysis technique is invertible, as is the case with LPC coding, it is even possible to recover the speech, more or less accurately, by inverting the projection, reversing the time warp, and performing, for example, LPC synthesis on the resulting frames of estimated LPC coefficients.

3.6.2. Connectionist Dimensionality Reduction

One of the widely touted advantages of neural networks trained with backpropagation over other statistical methods is that they can learn non-linear transformations of their inputs. An example of such a transformation is the one implemented by the network in Fig-

[5] Sometimes also known as Linear Discriminant Analysis (LDA).

ure 11. In principle, this ability gives neural nets far greater power than linear models

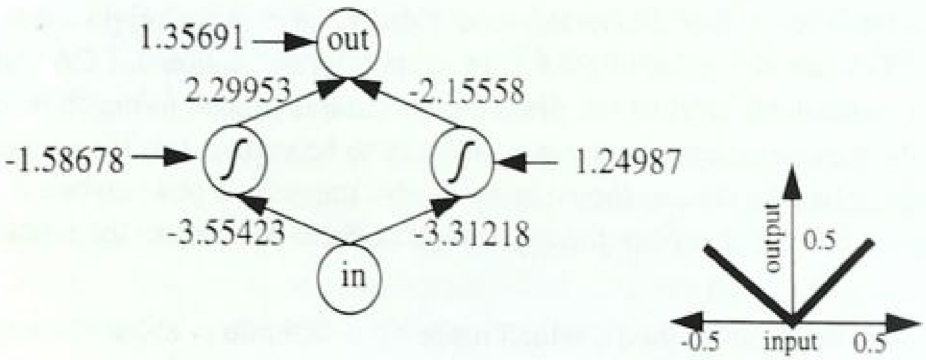


Figure 11: A trivial example of a nonlinear function learned by a neural network, $out = |in|$, or, more precisely {(-0.5,0.5), (0,0), (0.5,0.5)}. Weights are shown on the links connecting nodes, biases are the values next to each non-input node. The input-output fuction computed by this network is plotted at the right.

[minsky88], both as function approximators and as classifiers, if the appropriate weight settings can be learned. What these networks are doing, in effect, is multivariate multiple nonlinear regression [sarle94]. It is the increased complexity in the regression function that the nonlinearity provides, that gives these networks the potential to more closely match the target values.

The Power of Bottleneck Networks

The idea of non-linear dimensionality reduction using neural networks is to use a network like that of Figure 11 to produce a non linear encoding of the input, and a second, similar network, concatenated with the first, to invert the encoding. The layer at which the encoding and decoding networks coincide is called a bottleneck[6], and the hidden unit activations (or outputs, if preferred) in this bottleneck layer constitute a reduced dimensionality representation of the inputs. Cottrell [cottrell90] used networks with a 64x64 grid of input and output units, representing greyscale pixels, and 80 (or fewer) hidden units in a single hidden layer to form reduced representations of face images. He then used these images as input to networks designed to extract features such as sex. The use of such networks is, however, somewhat controversial; in a very instructive paper, Boulard and Kamp [boulard88] showed that for "standard" three layer networks, using their one hidden layer as a bottleneck, the representations learned are at best equivalent to a subset of the principal components of the input. In fact, they showed the more general result that no matter how many layers the network has, *if* the bottle neck layer is the penultimate layer, then the representation learned can be no better that the optimal linear subspace found by principal components analysis[7].

There is a danger that this result will be seen as being more discouraging than it should be. One widely used connectionist text [hertz91], for example, seems to imply, although it does not state, that the Boulard and Kamp paper showed non-linear compression to be a hopeless

[6] The first use of the term "bottleneck", or "goulot d'étranglement" in French, has been ascribed to Yann le Cun, but a suitable reference could not be located.

[7] It is noted that this optimal linear subspace is, in itself, a rather good reduced representation for faces, and one that has been widely used in the literature [valentin94].

prospect for all networks trained with backpropagation. This is not the case; while the number of layers required may seem daunting, especially if one is not used to using networks with bypass (or short-cut) connections [lang88], networks with a layer (or layers) between the bottleneck and the output units can learn non-linear encodings of considerable complexity. In fact, DeMers and Cottrell [demers93] repeated the experiments in the earlier paper [cottrell90], compressing the first fifty principal components of the images to five dimensions using such a five layer autoassociative net[8].

To drive home the point that these networks can achieve better-than-linear performance, the two networks shown in Figures 12 and 13, each with a hidden layer following the bottleneck, were trained, using backpropagation with momentum, to encode increasingly complicated non-linear functions in a single dimension, by reproducing the input function on the output of a network with a single unit in its bottleneck layer.

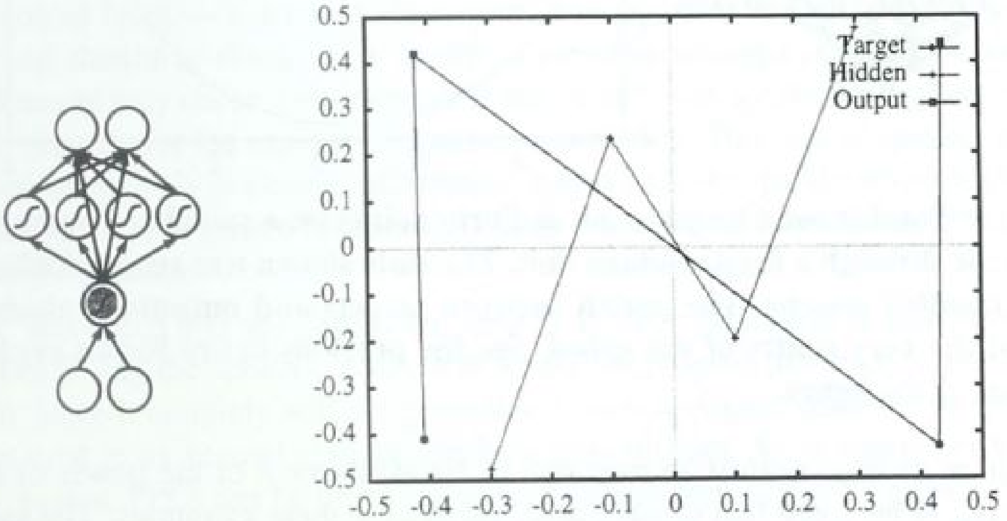


Figure 12: Non Linear Compression 1: four points on a square are passed without significant error through a single hidden unit. The state shown was reached after three hundred thousand training epochs. The outputs match the targets so well that their positions are indistinguishable on the graph. For the input and output activations, the dimensions of the graph correspond to the two input and output units, respectively. For the hidden unit activations, successive exampes of the four input patterns are spaced evenly across the *x* axis, and the y axis is the unit's output.

The first figure (12) shows a net that was trained to implement one of the simplest non linear compressions possible. After training, four input points on the corners of a square in 2-space are encoded as four distinct points on a line by the single unit in the network's bottleneck. The four input tuples are reproduced almost exactly on the network's outputs after decoding in the second hidden layer.

The second figure (13) shows an example of a non-linear problem familiar from the connectionist literature [lang88] adapted to this context. A spiral, which goes through nearly 2 complete revolutions, is encoded, again, by a single hidden unit. It is perfectly evident, in this case, that thanks to the hidden layer that follows the bottleneck, it has been possible for

[8] They did not, regrettably, compare the distortion of images compressed this way with those compressed by projection onto five principal components.

the representation in the bottleneck to be something much more complicated than a projection onto the first principal component of the training set.

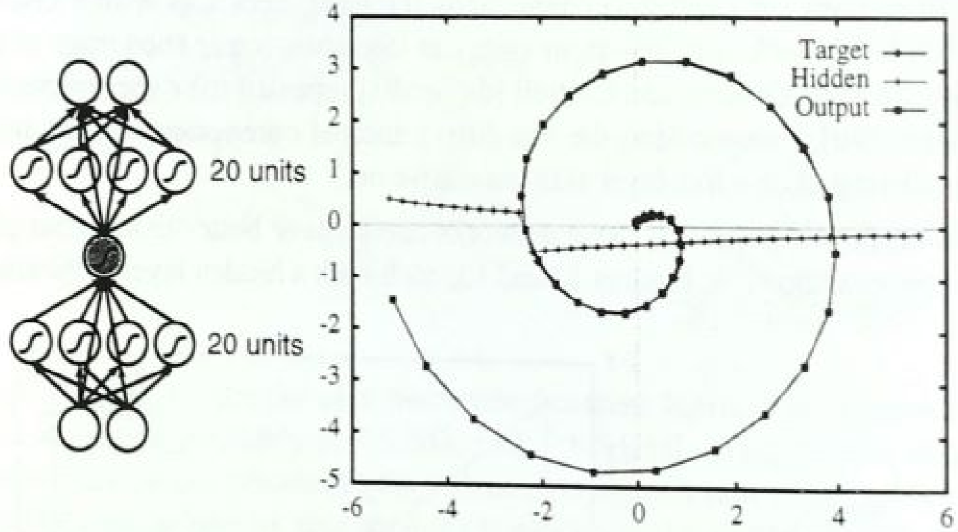


Figure 13: Non Linear Compression 2: Forty points on a spiral are passed with only slight error through a single hidden unit. The state shown was reached after fourteen million training epochs. The match between targets and outputs is almost perfect, except at the very centre of the spiral. See the previous figure for an explanation of the layout of the figure.

As a note of caution against an over optimistic assessment of the power of neural networks, it should be noted that it was not trivial to train these examples. The author is not confident that the examples given could have been learned without the use of the combination of short-cut connections outside the bottleneck, and comparatively large initial weights[9]. Even with these conditions satisfied, the networks took an extraordinarily large number of training passes[10] to move beyond a "linear" hidden representation, and thence to the performance shown.

Earlier in this chapter, models formed by projecting the vectors derived from each phone onto their first few (n) principal components were discussed. If, instead, bottleneck networks with the same number, n, of hidden units were trained on the same data, it is reasonable to hope that they would form a more compact representation that retains more information about the phones in the same sized representation. Given the difficulty of learning these nonlinear representations, though, it is hard to predict how much of an advantage the representations formed in bottleneck networks will provide.

In the light of this discussion, the experiments reported below, which compare the two methods, are interesting in two separate respects. They serve both as a demonstration of extracting speaker information from phones, the ostensible and primary purpose, and as an experiment to compare the usefulness of PCA and bottleneck networks as dimensionality reduction tools on a real-world task.

[9] It seems, perhaps, that the large initial weights (of the order of 2.0) prevent the network from too easily reaching local minima that involve setting the non-shortcut weights to zero, or making the weights symmetrical.
[10] Fourteen million, in the case of the spiral problem.

3.6.3. Statistical Discrimination

Principal components analysis can reduce the dimensionality of speech segments by finding a linear projection of those segments that retains as much as possible of the variation in the segments, but in the lower dimensional space. Compression neural networks with a narrow bottleneck in one hidden layer can perform a similar reduction in the dimensionality of speech samples, either by finding an approximation of the principal components, or, if necessary, by learning a more complex non-linear encoding and a matching decoding.

Compressing the data in this way, however, is not necessarily the best choice if the goal is speaker modelling. While the projections will capture variation, they do not care where the variation comes from — there is no constraint that says that variation within a speaker is irrelevant and should be discarded in favour of variation amongst speakers. A purely variation based model may chose a representation that retains non-speaker dependent variation in the speech segments at the expense of speaker information. This loss of speaker information and addition of possibly irrelevant information may reduce the speed with which the speaker model reaches an appropriate location in speaker space, and the stability of that position within a single speaker.

Earlier it was noted that one degenerate form of speaker model would consist of a 1-from-n vector identifying the speaker. While one would not propose to use such a representation as a model, since it certainly will not generalise to new speakers, there are other techniques that can be used in an attempt to make speakers more distinct. As an alternative to the variational techniques, PPCs can be based on representations formed during attempts at classification. While the 1-from-n representation that classification learns for training speakers will not be used, it is to be hoped that the internal representations of such classifiers will separate these training speakers well, and that they will also distinguish novel speakers. It is these internal representation that will be used as PPCs.

The first of these classification methods is linear discriminant analysis (LDA). In this technique a linear transformation is learned that projects a database labelled by groups onto a linear subspace which maximises the ratio of between groups variation to within groups variation. In the speaker modelling application, of course, the technique is used to find the projection of the speech signal that maximises the distance between samples from different speakers, while minimising the distances between samples from a single speaker.

The projection matrix used in LDA is made up of a chosen number of the eigenvectors, with highest corresponding eigenvalues, of the ratio covariance matrix $\Sigma_B \Sigma_W^{-1}$, where Σ_B is the between groups covariance matrix (i.e. the covariance matrix of the group mean vectors), and Σ_W is the within groups covariance matrix (i.e. the covariance matrix of all the vectors, after the appropriate group means have been subtracted). Projecting onto these eigenvectors produces a set of vectors that maximise the amount of retained variability that is due to group differences in the original vectors, and minimises that due to within group variability.

3.6.4. Connectionist Discrimination

Perhaps the most popular use of connectionist networks is in pattern classification. If the data fall into n classes, the network is trained to produce a distinct output value on the one output unit, out of n, corresponding to the class of the input pattern. As was the case with the compression networks above, the activations of units in a hidden bottleneck layer can be used as a reduced dimensionality representation of the input, although not, this time, one that can depended upon to be invertible. In this case, these hidden unit vectors should be similar within speakers, and different between speakers, since they are the support for an output representation that, if training is successful, will be almost identical within speakers, and perfectly distinct between them. Webb and Lowe [webb90] show that, for a somewhat simpler network architecture using linear output units connected to the hidden layer, the network maximises a network discriminant function

While the same networks are employed, and they are still doing a sort of multivariate multiple non-linear regression, this use of neural networks differs somewhat from their use as a functional approximator described above. Different sorts of output error are acceptable in the two cases. While, in a function approximator, the output vector of the network should match the target vector as exactly as possible, indicating that training should be done with an error function that is proportional to the distance to be minimised, in classification, it does not matter what the output values are, exactly, as long as the target class's unit is the supremum of some chosen function. Usually one wants the target unit to have a higher value than all the non target units, but that is all that is required. John Hampshire [hampshire90] suggested an "error" function, called a classification figure of merit, based on this goal of minimising misclassifications. Using this error function, error is only backpropagated to the target unit, and to the non-target unit with the highest output. In some of the experiments reported here, an error function similar to Hampshire's CFM error measure[11] was used, as follows.

$$j = \underset{i}{\operatorname{argmax}}(t_i) \qquad k = \underset{i \neq j}{\operatorname{argmax}}(o_i)$$

$$cfm = \left(o_j - o_k + \frac{1}{2}\right) - \frac{1}{9}\log\left(1 + e^{9\left(o_j - o_k - \frac{1}{2}\right)}\right)$$

$$cfm' = \frac{1}{1 + e^{9\left(o_j - o_k - \frac{1}{2}\right)}}$$

The value cfm' is backpropagated into the highest non-target unit, k, and value $-cfm'$ into the target unit, j.

While this CFM does tend to speed convergence and decrease error on the training set, it isn't certain that it is best thing for speaker modelling purposes. While learning to classify the training speakers, the hope is that the network will produce a hidden unit representation that distinguishes speakers, while retaining the similarities that exist between similar speakers. CFM expends more of its effort than the usual squared difference measure in forcing

[11] When the neural net code used here was being written, John Hampshire's function was, apparently, the subject of a patent application. For this reason, a function was independently produced that was intended to have similar properties.

similar speakers (the target, and its closest competitor) apart. While this will improve classification performance, it may decrease the utility of the hidden representation for speaker modelling purposes, by interferring with the aim of producing a topographic representation of speaker similarity in the representation space.

In fact, this is a general danger, not only with this particular measure, but with using discrimination of any sort to train the models. While discriminative training does provide an incentive for the model to concentrate on those features of the speech that distinguish speakers, there is a risk that the parts of the signal that best distinguish the training speakers do not include the features a human listener would consider important components of voice personality. It is to be hoped that the balance between these possibilities lies in favour of successful speaker modelling, and that the speakers will be separated as meaningful groups, before they are separated as individuals.

3.7. Measuring Performance

As various speaker models are developed, an objective measure under which they can be compared is necessary. An ideal such measure could have two forms. If the model is designed for a particular task, then, of course, the ideal performance measure is the change in a performance measure specific to that task, if, indeed, such a performance measure exists. If the model is intended to be task-neutral, the ideal speaker space would have the same topology as a human speaker space. That is, speakers perceived as being more similar by human listeners should lie closer together in this space. Unfortunately, obtaining similarity measures of any reliability from human subjects on a speaker set of any size would be a considerable research project in itself, and not one that could have been completed for this thesis.

Despite the *caveat* in the previous section against blindly separating speakers without regard for the larger speaker groups of which they are a part, a measure of model quality can be approximated by measuring the degree to which a candidate model distinguishes different speakers from each other.

Asoh [asoh90] used a discriminant criterion $J = \mathrm{tr}(\Sigma_B \Sigma_W^{-1})$, where Σ_B is the between groups covariance matrix (i.e. the covariance matrix of the group mean vectors), and Σ_W is the within groups covariance matrix (i.e. the covariance matrix of all the vectors, after the appropriate group means have been subtracted), so $\Sigma_B \Sigma_W^{-1}$, is the ratio covariance matrix used in canonical discriminant analysis. The function tr() is the trace function, the sum of diagonal elements.

However, the trace of a matrix is equal to the sum of the eigenvalues of the matrix and, following [freidman86], these eigenvalues can be regarded as mean squared length. The square root of the trace can consequently be regarded as an overall radius for the data in the discriminant space, and is a figure of merit for classifiability. Consequently, in comparing reduced data sets, the following measure will be used.

$$J = \sqrt{\mathrm{tr}(\Sigma_B \Sigma_W^{-1})}$$

The values of J reported with experiments should be used to compare them with other similar experiments, rather than regarded as an absolute measure. Since J increases with increasing dispersion of the group means relative to the dispersion within groups, larger values of J suggest a better speaker or phone model, or, at least, a model that better serves to distinguish speakers.

After experiments comparing models of speaker variation in phones have been described in the following sections, the results of measurements of both J and a more direct measure of classifiability will be compared in §3.13 to discover how well the measure predicts classification accuracy.

3.8. Experiment: Comparing Dimension Reductions

Since speaker models based on PPCs derived by all four of the dimensionality reduction techniques could not practically be produced, an experiment was performed to compare these PPCs in isolation from the rest of the system. Although the measure J described above is not entirely satisfactory, it served as a practical objective measure on which a comparison could be based. Since the main danger with the measure is a loss of speaker groupings, the techniques were also compared with respect to their retention of the difference between men and women, since this is the only obvious voice personality grouping for which have class information was available in the database.[12]

3.8.1. Materials

As explained at the beginning of this chapter, phone models were trained using speech only from regions 1, 2 and 3 (New England, Northern, North Midland) of the TIMIT database. For training, the speakers from the "train" subset of the database were used, and for these speakers, only speech from the five "sx" and three "si" (phonetically compact[13] and diverse, respectively) sentences per speaker was used. There were 190 speakers in the training subset used, and a total of 950 utterances.

Before the dimension reduction techniques were applied, the speech was preprocessed as follows: All instances of each of the phones ("ix", "s", "n", "tcl", "l", "r", "iy", "kcl", "ih", "dcl", "t", "k", "ax", "z", "m", "eh", "pcl", "q", "axr", "p", "d", "dh", "w", "f", "ae", "aa", "ah", "b", "ey" and "v") were excerpted from the files of digitised speech using the phone label files provided. In order that the original phone ordering could be reimposed on the separated phones later, indexing information, specifying the start and end time of each phone, and the sentence from which it was excerpted, was retained for each phone. The excerpted speech was zero padded at the end to a multiple of 128 samples, and analysed using a FFT on 128 sample non-overlapping windows, yielding 64 filterbank power coefficients per 128

[12] Geographical region, while it is distinguished in the database, has its main affect on accent, which the current work does not attempt to model. Pilot analyses looking for modelled differences between the regional groups did not produce positive results.

[13] These sentences were designed to provide good coverage of pairs of phones, and to include extra occurrences of phonetic contexts thought to be difficult or of particular interest by the corpus designers.

sample frame. Frames inside phones were linearly warped to a constant five frame length as described earlier in this chapter, yielding a single 320 coefficient vector per phone. Speaker identity, regional identification, and phonetic context information was stored with these vectors. These 320 coefficient vectors extracted from each phone were subjected to the dimension reduction methods being compared.

3.8.2. Procedure

Although the input and the PPCs formed by each of the four techniques were identically represented as vectors of floating point numbers, the way they were processed differed somewhat. The method of construction for each of the models will be described before their performance is compared. In all cases, the set of vectors representing a single phone were modelled separately. For example, when the text describes the training of neural nets using phone vectors, it means that thirty such nets were trained, each of these nets being trained and tested only on the subset of vectors corresponding to a particular one of the thirty phones.

For each dimension reduction technique, models producing PPC vectors of length 1, 2, 3, 4, 5, 10 and 15 were trained. In total, 210 (seven PPC lengths by thirty phones) models were built for each modelling method.

Projection onto Principal Components

Using a version of the PCA program described in [dennis91] that had been slightly modified to allow it to handle larger vectors, eigenvectors of the covariance matrices for the phone vectors were calculated. These eigenvectors were sorted by decreasing eigenvalue, the eigenvectors with the *n* largest eigenvalues being the first *n* principal components. Projections of phone vectors onto the first *n* principal components were calculated, for the seven values of *n* listed above, and these were used as PPCs of dimension *n*.

Neural net compression

Phone instance vectors were compressed by neural networks with the five-layer topology shown in Figure 14. Bypass connections are present between every pair of layers, except

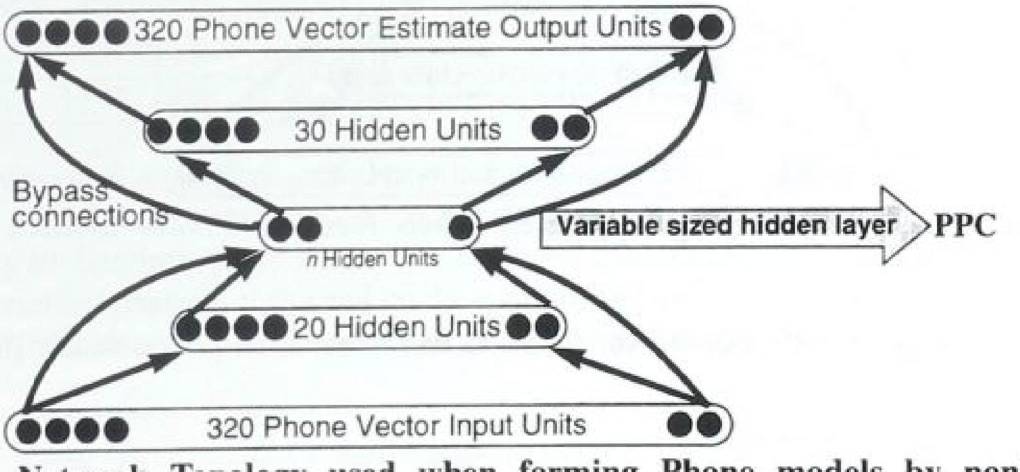

320 Phone Vector Estimate Output Units
30 Hidden Units
Bypass connections
n Hidden Units
Variable sized hidden layer
PPC
20 Hidden Units
320 Phone Vector Input Units

Figure 14: Network Topology used when forming Phone models by nonlinear compression.

where such connections would bypass the bottleneck in the second hidden layer. The number of units in the two fully hidden layers were chosen on the basis of a combination of the results of pilot experiments, the belief that the decoding task is more difficult than the encoding task, and the need to control the amount of computation required - there are probably many other choices for the size of these layers that would do just as well. During training, the same phone vector was used as input and target. These networks were therefore trained as constrained function approximators, where the function being approximated was the identity function. In learning to reproduce their inputs on their output units[14], by changing their weights to minimise the mean squared error between their outputs and the target, the networks had to form a representation of the input that could be contained in the outputs of the *n* units in the 2nd hidden layer (shown in red). These *n* hidden layer outputs were collected for each phone vector, and were used as PPCs.

Projection onto canonical discriminants

Using a version of a program (CDA) described in [dennis91, again slightly modified to allow larger vectors, eigenvectors of the between group/within group ratio covariance matrices[15] for the phone vectors were calculated. These eigenvectors were sorted by decreasing eigenvalue, the eigenvectors with the *n* largest eigenvalues being the first *n* canonical discriminants - the directions that maximally separated the speakers while keeping the utterances of a phone by a single speaker tightly clustered. Projections of phone vectors onto the first *n* canonical discriminants were calculated, for the usual values of *n*, and these were used as PPCs of dimension *n*.

Neural Net discriminator

Neural networks having the five-layer topology shown in Figure 14 were trained to identify which of the 189 training set speakers had uttered the phone vector presented to the

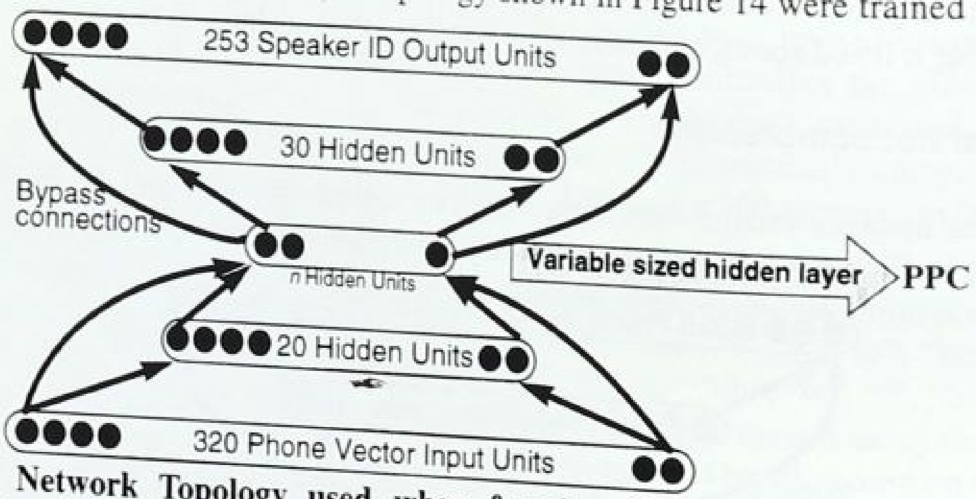

253 Speaker ID Output Units
30 Hidden Units
Bypass connections
n Hidden Units
Variable sized hidden layer
PPC
20 Hidden Units
320 Phone Vector Input Units

Figure 15: Network Topology used when forming Phone models by nonlinear discrimination.

[14] Training parameters for the networks are given in Table B-1 in Appendix B.

[15] The reader may wish to refer back to the description of CDA given earlier in this chapter.

input[16]. Again, the vector of outputs of the units in the second hidden layer was used as the PPC for the phone instance whose vector had produced them.

To investigate the role of the training criterion in deciding speaker modelling performance, these experiments were done using both mean squared error and the previously described approximation to Hampshire's [hampshire90] CFM as the error function for the network during training.

3.8.3. Results

Once the PPCs for the thirty phones had been calculated for all four of the modelling methods, the corresponding speaker labels were used to permit calculation of the measure J for the PPCs. These values are a measure of the relative linear discriminability of the speakers based on the PPCs. There is a strong relationship between the measure J and actual discrimination scores that will be described later in this chapter. Full tables of the discriminant measure, calculated on training set data, by phone and PPC size for the four techniques, PCA, NNCompress, LDA and NNDA are given in Appendix D. as Tables D-1, D-2, D-3 and D-4 respectively. Tables 5 and 6 and summarize these results and similar results for test set data for consonants and vowels respectively, by giving the mean value of J across all PPC sizes.

Table 5: Discriminant measure (J) for the four types of PPC, averaged across all PPC sizes, for vowels. The discriminant models separate speakers better than the variational models. The clear advantage of LDA in separating speakers in the training set is lost for the test set, on which the neural net discriminator has slightly better performance.

	Method	Phone										Mean
		ix	iy	ih	ax	eh	axr	ae	aa	ah	ey	
Train	PCA	1.10	1.30	1.29	1.20	1.42	1.40	1.72	1.72	1.55	1.78	1.45
	NNCompress	1.05	1.24	1.16	1.15	1.31	1.36	1.53	1.58	1.45	1.61	1.34
	LDA	2.37	3.07	3.28	3.08	3.74	4.42	5.55	5.88	5.66	5.84	4.29
	NNDA	2.09	2.45	2.59	2.00	2.78	2.17	3.26	3.03	2.81	3.25	2.64
Test	PCA	1.05	1.34	1.18	1.15	1.42	1.11	1.71	1.59	1.53	1.80	1.39
	NNCompress	1.00	1.26	1.07	1.13	1.22	1.11	1.52	1.41	1.37	1.61	1.27
	LDA	1.57	1.69	1.81	1.22	1.76	1.10	2.09	1.48	1.45	1.96	1.61
	NNDA	1.80	1.99	2.00	1.51	2.16	1.39	2.62	2.02	2.04	2.47	2.00

Surprisingly enough, considering the importance that has been ascribed to vowels in determining voice personality, vowel models are not strongly favoured over those for consonants in their ability to discriminate speakers. For principal components, for example, the mean pooled discriminant measure, measured on the training set, for vowels was 1.45 and for consonants 1.34. Since the standard deviations (0.22 and 0.28 respectively) are > 0.2, this differ-

[16] There are 253 output units for convenience in dealing with both training and testing speakers, to avoid the necessity relabelling the speakers. The 64 targets corresponding to test speakers were not used during training, and one would not expect them to be meaningful during testing.

Table 6: Discriminant measure for the four types of PPC, averaged across all PPC sizes, for consonants.

	Method	Phone																				Mean
		s	n	tcl	l	r	kcl	dcl	t	k	z	m	pcl	q	p	d	dh	w	f	b	v	
Train	PCA	1.88	1.42	0.78	0.95	1.02	1.04	1.14	1.47	1.29	1.84	1.63	1.20	1.17	1.49	1.38	1.32	1.31	1.60	1.34	1.54	1.34
	NNCompress	1.75	1.42	0.80	0.89	0.95	1.04	1.14	1.47	1.30	1.84	1.62	1.17	1.19	1.48	1.38	1.31	1.29	1.61	1.35	1.54	1.33
	LDA	2.92	2.82	1.96	2.61	2.71	2.75	2.85	3.08	3.02	3.58	3.77	4.74	3.97	4.35	4.21	5.00	4.84	5.64	9.91	6.63	4.07
	NNDA	2.65	2.27	0.90	1.93	1.91	1.11	1.32	2.34	1.98	2.64	2.30	0.97	1.93	1.97	2.00	1.80	1.93	1.86	1.75	1.88	1.87
Test	PCA	1.75	1.35	0.84	0.89	0.96	1.07	1.25	1.41	1.31	1.88	1.50	1.20	1.18	1.37	1.55	1.28	1.30	1.69	1.41	1.59	1.34
	NNCompress	1.66	1.36	0.85	0.84	0.90	1.09	1.25	1.41	1.31	1.89	1.49	1.16	1.16	1.39	1.57	1.29	1.27	1.70	1.41	1.57	1.33
	LDA	1.64	1.66	0.84	1.22	1.14	1.03	1.06	1.15	1.12	1.47	1.62	1.26	1.23	1.10	1.18	1.23	1.24	1.25	0.99	1.31	1.24
	NNDA	1.78	1.87	0.86	1.33	1.39	1.10	1.30	1.40	1.48	1.88	1.88	0.92	1.31	1.60	1.64	1.27	1.51	1.70	1.42	1.71	1.47

ence is not significant (t_{28} =1.03 P(equal means) < 0.31). Visual inspection suggests that across all techniques, there might be slight trend toward vowels distinguishing speakers more easily than consonants do, although statistical tests do not provide any compelling confirmation of this trend. Vowels from the training set, pooled across all techniques have J measures with a mean of 2.43 and s.d of 1.39. Consonants have a mean of 2.15 and s.d of 1.47. This difference is not significant (t_{118}=0.996, P(equal means) < 0.32).

The same measure of speaker discriminability for each PPC size, pooled across all thirty phones is given in Table 7 and plotted in Figure 16. The "variational" PCA and NNCom-

Table 7: Discriminant measure (J) for PPCs of various dimensions, for the four techniques. Each cell represents an average across all phones.

	Method	Width							Means
		1	2	3	4	5	10	15	
Train	PCA	0.6661	0.9520	1.1328	1.2575	1.4271	1.9218	2.2713	1.3755
	NNCompress	0.6394	0.9068	1.0804	1.2203	1.3732	1.8688	2.2378	1.3324
	LDA	2.3270	3.0750	3.5700	3.9689	4.3052	5.4825	6.2636	4.1417
	NNDA	1.1295	1.6258	1.8922	2.0941	2.2495	2.8063	3.1099	2.1296
Test	PCA	0.6390	0.9255	1.0989	1.2210	1.4077	1.9114	2.2829	1.3552
	NNCompress	0.6169	0.8912	1.0523	1.1968	1.3459	1.8500	2.2169	1.3074
	LDA	0.7190	0.9711	1.1491	1.2800	1.4095	1.8395	2.1721	1.3629
	NNLDA	0.8179	1.2218	1.4707	1.6145	1.7368	2.2069	2.4499	1.6455

press methods appear to be similarly effective, with the linear, PCA, method having slightly better performance. The linear discriminant model separated training speakers substantially better than the other methods. It has a significantly higher J measure than NNDA, the next best (t_{12} = 3.50, P(means equal) < 0.005). However, this very high value of the discriminability metric for LDA on training speakers is, perhaps, to be expected, since it is precisely this J measure that linear discriminant analysis attempts to maximise. On testing speakers the method performed less well - separating speakers no more than the variational methods did. For test data, the neural net discriminator did better, outperforming the linear and variational methods, although the difference was not significant (t_{12}=0.9975, p(equal means) < 0.34).

The second trend, displayed in Figure 16, is that, as one would expect, the discriminability

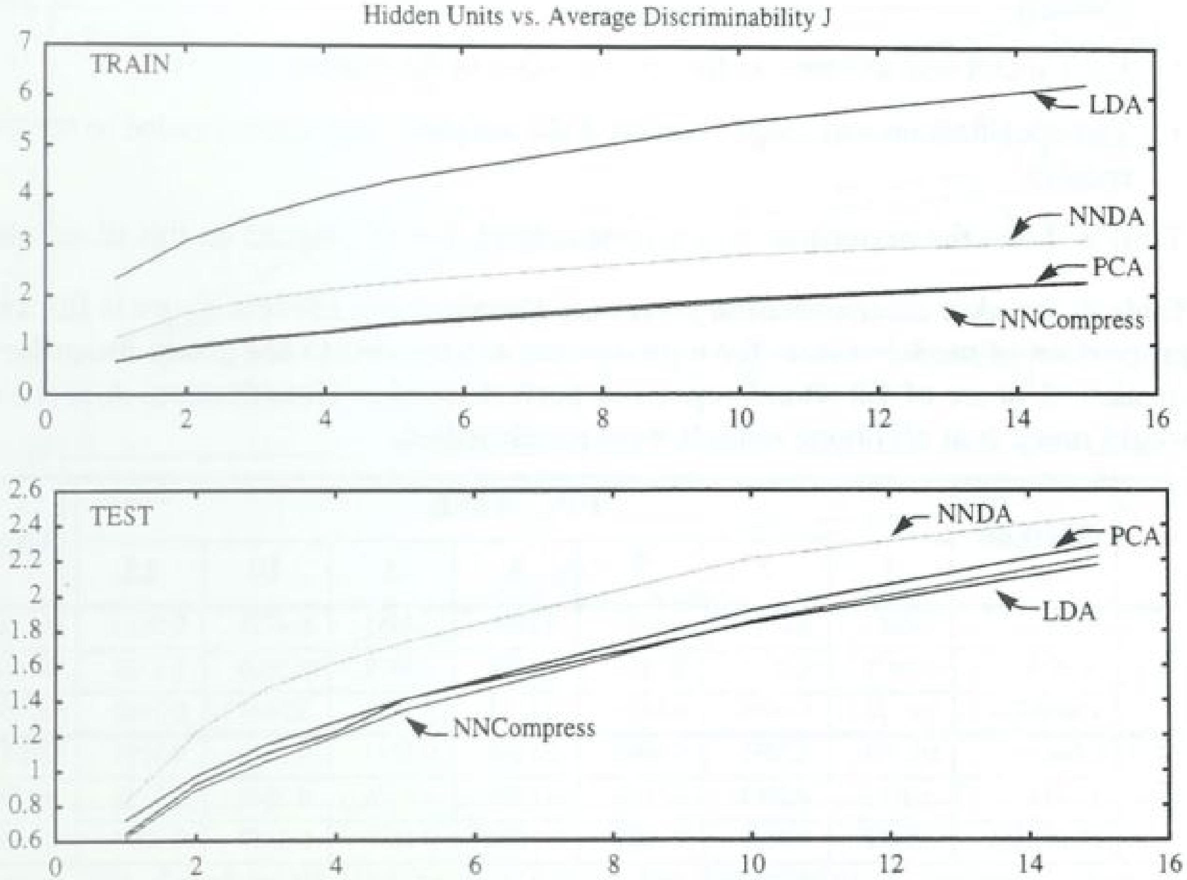


Figure 16: Discriminability of speakers on the basis of phones projected onto PPCs of various dimensions, for the four techniques, each pooled across phones. Linear discriminant analysis produce the most separated speaker codes for the training speakers, but for the test set, this advantage was lost, and the neural net discriminator was most successful.

of speakers in each model increases with the number of parameters in the PPC devoted to modelling each phone, with diminishing returns from extra hidden units. Clearly the amount of speaker information increases with the size of the PPC. Since the curve was levelling off above this point, PPCs with ten components seemed a reasonable choice to carry forward for use in building speaker codes.

Performance measured directly on a discrimination task

Since the training technique used for the NNDA model was trying to achieve good discrimination performance, rather than match the LDA model in maximising the separation of speakers in the space whose volume is approximated by the J measure, it was important, for comparison purposes, to see how well the NNDA networks and LDA performed on the discrimination task itself. To this end, a more direct measure of discriminability was calculated for each condition. For each speaker, the centroid for all the PPCs associated with a given phoneme was calculated. These centroids where used to calculate speaker classification rates, using nearest centroid classification for both the output vector of the NNDA model network and the PPCs formed in the hidden units, and for the LDA PPCs. The classification was done as follows:

- Within each phoneme, means were calculated for the set of vectors from each speaker.
- Each vector was assigned to the speaker class of the nearest mean vector.
- The classification was judged correct if the assigned class corresponded to the true speaker.

Table 8 shows the performance (0 = none correct, 1 = all correct) on this discrimination task for LDA PPCs (LDA), the outputs of the NNDA network (NNDA) and the PPCs formed in the hidden units of that network (HidNNDA). The complete tables of results for the discrimination tasks are given in Appendix D. as Tables D-5, D-6 and D-7.

Table 8: Speaker discrimination scores for Discriminant models. Score is the average proportion of model vectors for a phone that are nearest to the group mean for their speaker. A score of 1.0 would represent perfect speaker classification. A score of 0.0 would mean that all phone models were misclassified.

	Method	PPC Width							Means
		1	2	3	4	5	10	15	
Train	LDA	0.0291	0.0780	0.1355	0.1963	0.2581	0.4753	0.5933	0.2522
	NNDA	0.0437	0.0546	0.0886	0.1188	0.1478	0.2666	0.3323	0.1504
	HidNNDA	0.0226	0.0491	0.0841	0.1153	0.1435	0.2444	0.2862	0.1350
Test	LDA	0.0383	0.0697	0.0960	0.1219	0.1547	0.2539	0.3273	0.1517
	NNDA	0.0713	0.0891	0.1250	0.1568	0.1876	0.2881	0.3359	0.1791
	HidNNDA	0.0441	0.0839	0.1227	0.1520	0.1803	0.2682	0.2994	0.1644

As one would expect, the more units that were used, the better the discrimination. Reflecting the results with the J measure, LDA based PPCs serve to discriminate the training set somewhat better than those based NNDA ($t_{12} = 1.34$, P(LDA<HidNNDA) < 0.102), but this advantage is not robust. For test speakers, the hidden representations formed by the neural net discriminator are more distinct than the LDA representations[17] although this difference is not significant ($t_{12}=0.24$ p(means equal)<0.814).

Although, again, the difference is not significant, the discrimination score is higher for the output units of the NNDA than for the hidden units, suggesting that the previously stated reservations about the use of the J score in this case were not entirely unfounded - the internal representations formed are somewhat more distinct with respect to a non-linear classifier than the linear classifiability measure J would lead one to expect. The measure seems to slightly underestimate the quality of PPCs formed in non-linear models.

3.8.4. Conclusions

Since the vowel and consonant models have similar power to distinguish speakers, the choice of the most frequent phonemes regardless of class, as a basis for building the models was confirmed as a reasonable one.

[17] In fact, the NNDA does slightly better for test speakers than for training speakers. This is probably coincidental.

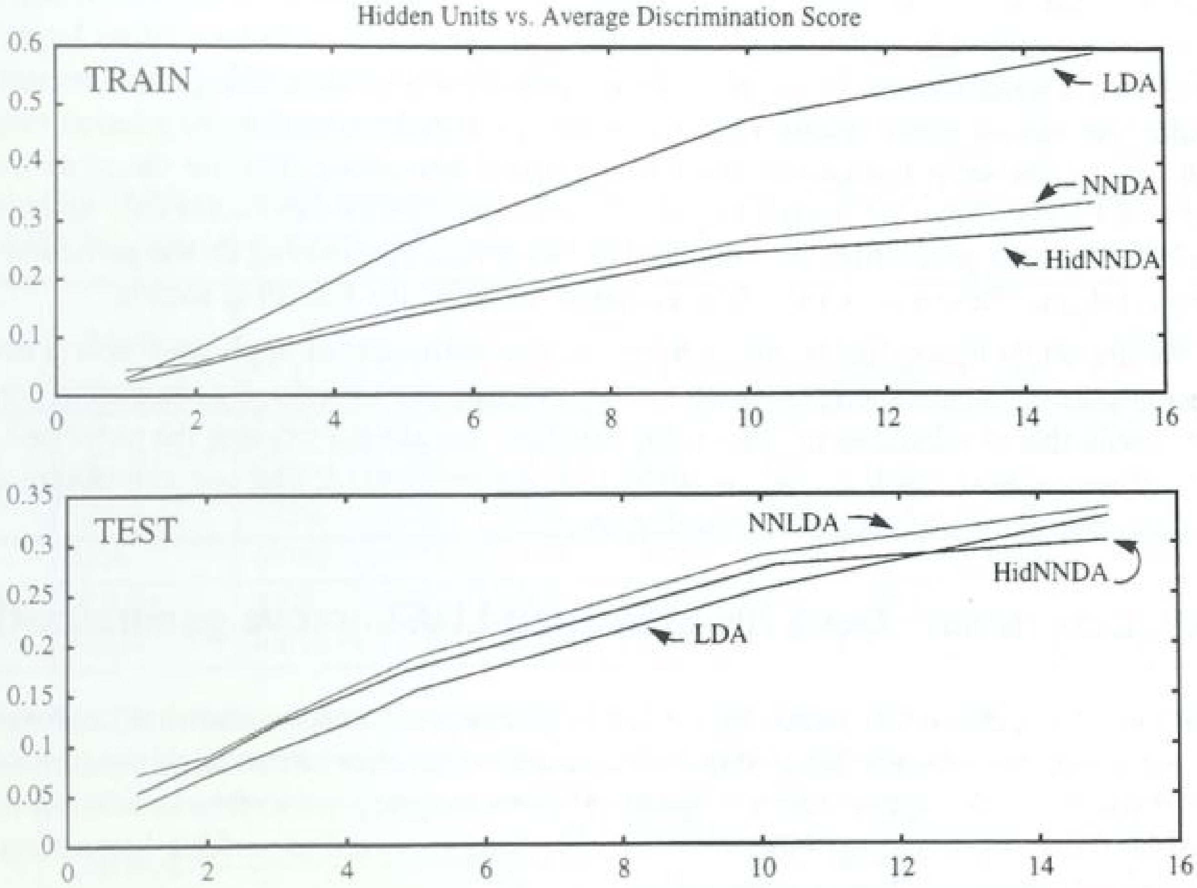


Figure 17: Speaker discrimination scores for Discriminant models. Score is the average proportion of model vectors for a phone that are nearest to the group mean for their speaker. Again, in this case, the training set advantage for LDA is lost on testing data, but here there is no clear advantage for the neural net during testing.

For the variational techniques, the principal components analysis and the neural network compressor performed nearly identically. This suggests that these networks, like the ones used in Cottrel's [cottrell90] early work on face compression, were simply calculating principal components of the phone example vectors, and very inefficiently at that. When the training of example compression networks was described earlier in this chapter, it was pointed out that it took a very long time, even with the very clean data used, to escape from the linear approximations of the solutions that the networks learned initially. Even if the networks could have learnt a non-linear compression from the noisy data used here, it is far from clear that the number of training epochs used was sufficient, despite the fact that this amount of training took a vast amount of computation when summed over the 210 networks trained. If bottleneck compression networks are to be a useful technique for training non-linear encodings, then it is clear that specialised methods for improving the training of such networks need to be found. One possibility, that there was not sufficient time to explore, would be to preload the network with weights derived from principal components of the training data, ensuring that the network's training is focused entirely on learning a non-linear component of the encoding.

In the discriminant case - the more usual application of neural networks - there was weak evidence that the neural nets performed somewhat better than the linear discriminant analysis on testing data. Since the LDA discriminated training speakers better, it is likely that this