An Investigation of Manifold Learning for Speech Analysis

Andrew Errity and John McKenna

School of Computing
Dublin City University, Dublin 9, Ireland
{aerrity, john}@computing.dcu.ie

10.21437/Interspeech.2006-628

Abstract

Due to the physiological constraints of articulatory motion the speech apparatus has limited degrees of freedom. As a result, the range of speech sounds a human is capable of producing may lie on a low dimensional submanifold of the high dimensional space of all possible sounds. In this study a number of manifold learning algorithms are applied to speech data in an effort to extract useful low dimensional structure from the high dimensional speech signal. The ability of these manifold learning algorithms to separate vowels in a low dimensional space is evaluated and compared to a classical linear dimensionality reduction method. Results indicate that manifold learning algorithms outperform classical methods in low dimensions and are capable of discovering useful manifold structure in speech data.

Index Terms: speech analysis, manifold learning, dimensionality reduction, classification.

1. Introduction

In speech processing, the speech signal is often modeled by relatively high dimensional features such as discrete Fourier transform (DFT) or linear prediction (LP) coefficients. However due to physiological constraints the speech production apparatus has relatively few degrees of freedom. Thus, humans are only capable of generating a limited range of sounds which occupy a confined region of the acoustic space. In this case, we can imagine the speech data as lying on or near a manifold embedded in the high dimensional acoustic space. It has been proposed that speech intrinsically lies on some such low dimensional manifold [1, 2].

It is desirable to reduce the dimensionality of the speech signal prior to processing. Traditionally, signal processing techniques have been applied to speech in order to reduce the dimensionality by extracting information that is judged to capture information about the energy and spectral characteristics of the signal. The extracted information is often transformed according to some perceptually motivated scheme to better model the speech auditory path; for example, Mel-frequency cepstral coefficients (MFCC) and perceptual linear prediction (PLP) parameters. These acoustically and perceptually motivated representations are based on our knowledge and assumptions of speech production and perception, and as such do not attempt to automatically discover the underlying low dimensional structure of speech.

A number of automatic dimensionality reduction algorithms, driven by the statistics of the data, have been proposed that aim to extract a meaningful low dimensional representation of high dimensional data. Applications of these dimensionality reduction algorithms include data compression, visualisation, noise reduction, and feature extraction. Dimensionality reduction methods can be categorised as linear or nonlinear methods. Linear methods are limited to discovering the structure of data lying on or near a linear subspace of the high dimensional input space. The most widely used linear dimensionality reduction methods include the classic principal component analysis (PCA) [3] and multidimensional scaling (MDS). These methods have been applied to a wide range of speech processing problems including, feature transformation for improved speech recognition performance, speaker adaptation, data compaction, and speech analysis.

Jansen and Niyogi [2] have recently shown that certain classes of speech sounds lie on a low dimensional manifold nonlinearly embedded in the high dimensional acoustic space. A low dimensional submanifold such as this may have a highly nonlinear structure that linear methods would fail to discover. Recently, a number of manifold learning (also referred to as nonlinear dimensionality reduction) algorithms have been proposed [4, 5, 6] which overcome the limitations of linear methods. These methods have been successfully applied to a number of benchmark manifold problems and have also proved useful in several image processing applications.

Manifold learning algorithms may also be useful in speech analysis; for example, to project speech into a low dimensional space for visualisation or extract features for use in speech recognition. However there has been relatively little research conducted in this area to date. A number of exploratory studies have shown that manifold learning algorithms can be used to successfully visualise speech data in a low dimensional space [7, 6, 8] and for phone classification [9].

In this paper, we apply several manifold learning algorithms—locally linear embedding (LLE) [4, 10], isometric feature mapping (Isomap) [5], and Laplacian eigenmaps [6]—to speech data. The ability of these algorithms to discover low dimensional structure within speech data is evaluated and compared. Their performance is also contrasted with that of the classical, linear, PCA method [3]. This paper is structured as follows. In Section 2, the manifold learning algorithms LLE, Isomap and Laplacian eigenmaps are described. The corpus, experiments and results are detailed in Section 3. Section 4 discusses a number of limitations of the manifold learning algorithms. Finally, in Section 5, the conclusions are presented.

2. Manifold learning algorithms

2.1. Locally linear embedding

LLE [4, 10] is an unsupervised learning algorithm that computes low dimensional embeddings of high dimensional data. The principle of LLE is to compute a low dimensional embedding with the property that nearby points in the high dimensional space re-

main nearby and similarly co-located with respect to one another in the low dimensional space. In other words, the embedding is optimised to preserve local neighbourhoods.

The LLE algorithm can be summarised in three steps:

1. For each data point X_i, compute its k nearest neighbours (based on Euclidean distance or some other appropriate definition of 'nearness').
2. Compute weights W_{ij} that best reconstruct each data point X_i from its neighbours, minimising the reconstruction error E:
$$E(W) = \sum_i \left| X_i - \sum_j W_{ij} X_j \right|^2 \quad (1)$$
3. Compute the low dimensional embeddings Y_i, best reconstructed by the weights W_{ij}, minimising the cost function Ω:
$$\Omega(W) = \sum_i \left| Y_i - \sum_j W_{ij} Y_j \right|^2 \quad (2)$$

In step 2, the reconstruction error is minimised subject to two constraints: first, that each input is reconstructed only from its nearest neighbours, or $W_{ij} = 0$ if X_i is not a neighbour of X_j; second, that the reconstruction weights for each data point sum to one, or $\sum_j W_{ij} = 1 \; \forall i$. The optimum weights for each input can be computed efficiently by solving a constrained least squares problem.

The cost function in step 3 is also based on locally linear reconstruction errors, but here the weights W_{ij} are kept fixed while optimising the outputs Y_i. The embedding cost function in Equation (2) is a quadratic function in Y_i. The minimisation is performed subject to constraints that the outputs are centered and have unit covariance. The cost function has a unique global minimum solution for the outputs Y_i. This is the result returned by LLE as the low dimensional embedding of the high dimensional data points X_i. The embedding cost function can be minimised by solving a sparse $N \times N$ eigenvalue problem, as detailed in [10].

2.2. Isomap

The Isomap algorithm [5] offers a differently motivated approach to manifold learning. Isomap is a nonlinear generalisation of MDS that seeks a mapping from high dimensional space $\mathbf{X}$ to low dimensional feature space $\mathbf{Y}$ that preserves geodesic distances between pairs of data points—that is, distances on the manifold from which the data is sampled.

While Isomap and LLE have similar aims, Isomap is based on a different principle than LLE. In particular, Isomap attempts to preserve the global geometric properties of the manifold while LLE attempts to preserve the local geometric properties of the manifold.

As with LLE, the Isomap algorithm consists of three steps:

1. Construct a neighbourhood graph - Determine which points are neighbours on the manifold based on distances $d(i, j)$ between pairs of points i, j in the input space (as in step 1 of LLE). These neighbourhood relations are then represented as a weighted graph over the data points with edges of weight $d(i, j)$ between neighbouring points.
2. Compute the shortest path between all pairs of points among only those paths that connect nearest neighbours using a technique such as Dijkstra's algorithm.
3. Apply classical MDS to embed the data in a d-dimensional Euclidean space so as to preserve these geodesic distances.

2.3. Laplacian eigenmaps

The principle of the Laplacian eigenmaps algorithm is similar to that of LLE, to compute a low dimensional representation of high dimensional data that faithfully preserves proximity relations. It was originally motivated by the way that heat transmits from one point to another point. The algorithm is structured as follows:

1. Construct a neighbourhood graph as in Isomap.
2. Assign weights W_{ij} to the edges of the graph. These weights are typically constant, e.g. $W_{ij} = 1/k$, or exponentially decaying, e.g. $W_{ij} = e^{(-\|X_i - X_j\|^2/\sigma)}$, where σ is a scaling parameter.
3. Let θ denote the diagonal weight matrix with elements $\theta_{ii} = \sum_j W_{ij}$. The embeddings $\mathbf{Y}$ are computed by minimizing the cost function:
$$\epsilon = \sum_{ij} \frac{W_{ij} \|Y_i - Y_j\|^2}{\sqrt{\theta_{ii}\theta_{jj}}} \quad (3)$$

The outputs are constrained as in LLE. This cost function incurs a heavy penalty if neighbouring high dimensional points are mapped far apart.

3. Experiments

3.1. Data

The speech data used in this study was taken from the Boston University radio corpus [11]. This corpus provides radio news data, recorded by four male and three female radio announcers. Speech was recorded at a sampling frequency of 16 kHz. For these experiments, data was taken from two female (F1A and F2B) and two male (M1B and M2B) speakers; this provided a large amount of clean speech data from both genders.

Based on the phonetic transcriptions provided, all tokens of a subset of phones were extracted from the corpus. The phones extracted can be grouped into several broad phone classes and labeled using TIMIT phone symbols: vowels ('aa', 'ae', 'uw', 'iy', 'eh'), fricatives ('s', 'sh'), stops ('p', 't', 'k'), nasals ('m', 'n') and, semivowels and glides ('l', 'y'). One 40 ms frame was extracted from the middle of each vowel (tokens of duration less than 40 ms were discarded). The raw speech frames were amplitude normalised, preemphasized with the filter $H(z) = 1 - 0.98z^{-1}$ and Hamming windowed. Following this preprocessing, DFT feature vectors were computed for each frame. These features were then converted to log magnitude spectra.

3.2. Vowel separability analysis

150 tokens of each of the five vowels listed above were randomly selected for each speaker (i.e. 750 tokens per speaker). The resulting 3000 log magnitude spectra feature vectors were provided as input data to the LLE, Isomap, Laplacian eigenmaps and PCA algorithms. The number of nearest neighbours, k, used in the manifold learning algorithms was set equal to 12. Embedding spaces of dimensionality 1–15 were generated. The two dimensional embeddings of the vowels 'aa', 'uw' and 'iy' produced by each method are shown in Fig. 1, the vowels 'ae' and 'eh' are omitted for visual clarity. All three manifold learning algorithms were found to be useful for visualisation of the vowel data with individual vowels clustered in each embedding space. PCA also produces recognisable vowel clusters, in contrast to previously reported findings [7].

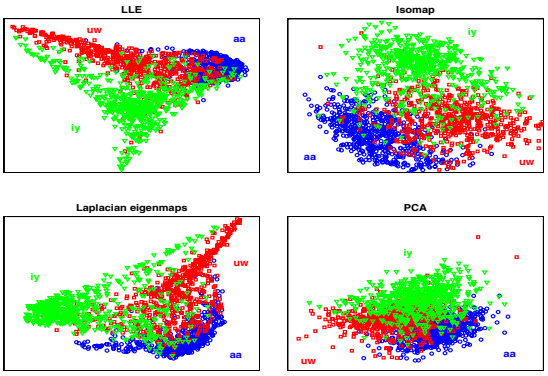

Figure 1: *600 tokens of each the vowels 'aa', 'uw' and 'iy' within the two dimensional vowel space produced by LLE, Isomap, Laplacian eigenmaps and PCA.*

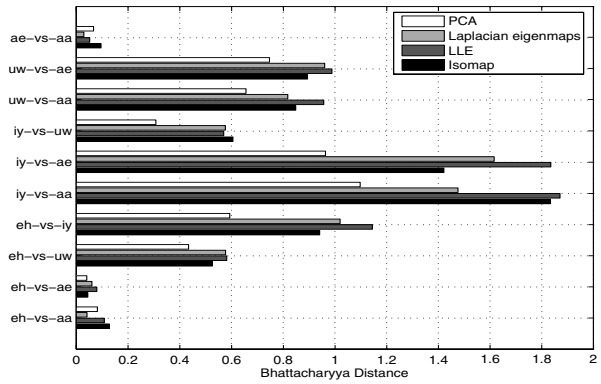

Figure 2: *Vowel pair separability in the the two dimensional embedding space produced by PCA, Laplacian eigenmaps, LLE and Isomap.*

However they have a greater degree of overlap compared to the embeddings produced using manifold learning techniques.

In order to formally evaluate the performance of each algorithm, a measure of vowel separability within the resulting low dimensional spaces was computed. The Bhattacharyya distance [12], measures the separability of two distributions and was used as a metric in this study. The Bhattacharyya distance between two distributions with mean vectors, $\mathbf{m}_1$ and $\mathbf{m}_2$, and covariance matrices, C_1 and C_2, can be computed as follows:

$$D_{bhat} = \frac{1}{8}(\mathbf{m}_2 - \mathbf{m}_1)^T \left[\frac{C_1 + C_2}{2}\right]^{-1} (\mathbf{m}_2 - \mathbf{m}_1) + \frac{1}{2} ln \frac{\left|\frac{C_1+C_2}{2}\right|}{\sqrt{|C_1|\,|C_2|}} \quad (4)$$

The Bhattacharyya distance was computed for all possible vowel pair combinations. This was performed in each of the 1–15 dimensional spaces produced by each algorithm. The ranking of both algorithm performance and vowel pairs according to Bhattacharyya distance was consistent across each of the 1–15 dimensional spaces. The results in two dimensional space are given in Fig. 2. In general the manifold learning algorithms outperform PCA, with LLE yielding the best vowel separability for 70% of the vowel pairs. Also, the separability of each vowel pair is shown to correspond to the relative position of each vowel in formant space. Vowels occupying a small region of formant space, i.e. 'aa', 'ae' and 'eh', have low Bhattacharyya distances between them.

3.3. Phone classification

To evaluate the usefulness of each manifold learning method, the low dimensional embeddings produced above were used as feature vectors in a vowel classification experiment. The training set consisted of 400 labeled feature vectors randomly chosen from each vowel, with the remaining 200 feature vectors per vowel used as test data. The embedding dimension d ranged from 1–15. A K-nearest-neighbour (K-NN) classifier, with $K = 2$, was implemented and used to assign each test feature vector to a vowel. This procedure was also used to classify tokens into phone classes. In this experiment, 1000 tokens were randomly selected from each of the five phone classes described in Section 3.1 and low dimensional embeddings computed as above. A K-NN classifier was trained on 2500 labeled feature vectors, 500 randomly chosen from each phone class, and tested on the remaining 2500 feature vectors. Each experiment was repeated using a Gaussian mixture model based classifier with three mixture components. The results were found to be consistent with those of K-NN.

The results of the K-NN classification experiments are shown in Fig. 3. For low dimensions the manifold learning algorithms outperform PCA. These results suggest the manifold learning techniques are successful at revealing low dimensional structure in speech data. Note, however that a crossover in error rates occurs as the number of dimensions increases. It appears that after this point LLE and Laplacian eigenmaps cannot extract further information from their locally linear neighbourhoods. In contrast, the PCA and Isomap features outperform LLE and Laplacian eigenmaps features for higher dimensions ($d > 4$).

As a baseline, both the vowel and phone class classification experiments were also performed using MFCC feature vectors, of order 12, as input. The resulting test error rate using a K-NN classifier is 25.7% for vowels and 31% for phone classes. These perceptually motivated features have been widely shown to be useful in speech recognition and, as expected, outperform the other features. Further investigation is planned into the possible benefit of using perceptual weighting, such as that used in MFCC, as a preprocessing step prior to manifold learning.

3.4. Pitch manifold

In addition to the phone separability experiments, LLE, Isomap, Laplacian eigenmaps and PCA were used to analyse data from individual vowels. For each vowel, 500 tokens were randomly selected. The equivalent DFT features were then reduced to two dimensional embedding space using LLE, Isomap, Laplacian eigenmaps and PCA. The manifold learning algorithms each used $k = 12$ nearest neighbours. A visual inspection of the two dimensional embeddings produced by the manifold learning algorithms found a distinct pattern within the data. The distribution of tokens within the embedding space was found to correspond to the pitch of the token. An example of this is shown in Fig. 4. It can be seen that pitch is consistently distributed in the embedding space produced by LLE. Moving clockwise from the top-right of Fig. 4, pitch can be seen to increase. Isomap and Laplacian eigenmaps were found to produce similar 'pitch manifolds', however the LLE

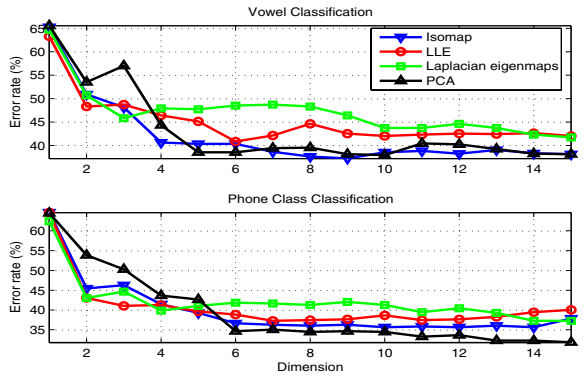

Figure 3: *Comparison of Isomap, LLE, Laplacian eigenmaps and PCA features for K-NN classification of vowels (top) and phone classes (bottom). The plots show error rates on the test set.*

embeddings revealed a more consistent pitch structure. The two dimensional representation resulting from PCA was found to yield relatively limited and inconsistent pitch structure. This suggests that vowels sounds of varying pitch may lie on a submanifold non-linearly embedded in acoustic space.

4. Limitations

The manifold learning algorithms discussed above have a number of properties that may limit their usefulness in speech processing applications. Firstly, these algorithms operate in batch mode. They do not provide a means of mapping new points between the high and low dimensional spaces, without re-running the algorithm with the new points added into the original data set. This would be a significant barrier in speech recognition applications. A number of approaches have been proposed [10, 13] to overcome this limitation but they have yet to be tested on speech data. Secondly, manifold learning algorithms do not scale well to large data sets ($N > 10000$). This is due to computational bottlenecks in the Isomap algorithm and difficulties resolving eigenvalues in the LLE and Laplacian eigenmaps methods.

5. Conclusions

Manifold learning algorithms have been found to be useful in speech analysis. These algorithms are capable of producing meaningful low dimensional representations of speech data. Such representations are useful in speech analysis as they reveal information that may relate to formant positions and place of articulation. A space in which different phones are well separated would also, clearly, be beneficial in ASR. While these algorithms need further development before they can perform as well as perceptually motivated features, such as MFCCs, they were found to be useful as a front-end for statistical recognition of speech sounds, outperforming PCA in very low dimensions. These algorithms have also been shown to produce embeddings which reveal other potentially interesting information, such as pitch.

6. Acknowledgments

This work is supported by the Irish Research Council for Science, Engineering and Technology; grant number RS/2003/11.

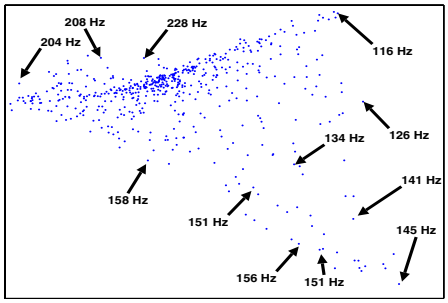

Figure 4: *Two dimensional LLE embedding of 500 tokens of the vowel 'aa'.*

7. References

[1] R. Togneri, M. Alder, and J. Attikiouzel, "Dimension and structure of the speech space," *IEE Proceedings-I*, vol. 139, no. 2, pp. 123–127, 1992.

[2] A. Jansen and P. Niyogi, "A geometric perspective on speech sounds," Tech. Rep., University of Chicago, June 2005.

[3] I.T. Jolliffe, *Principal Component Analysis*, Springer Series in Statistics. Springer-Verlag, New York, 1986.

[4] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, December 2000.

[5] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, pp. 2319–2323, 2000.

[6] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Advances in Neural Information Processing Systems*. 2002, vol. 14, pp. 585–591, MIT Press.

[7] V. Jain and L.K. Saul, "Exploratory analysis and visualization of speech and music by locally linear embedding," in *Proc. ICASSP*, 2004, vol. 3, pp. 984–987.

[8] Rajesh M. Hegde and Hema A. Murthy, "Cluster and intrinsic dimensionality analysis of the modified group delay feature for speaker classification," *Lecture Notes in Computer Science*, vol. 3316, pp. 1172–1178, January 2004.

[9] M. Belkin and P. Niyogi, "Semi-supervised learning on riemannian manifolds," *Machine Learning*, vol. 56, no. 1 - 3, pp. 209–239, July 2004.

[10] L. K. Saul and S. T. Roweis, "Think globally, fit locally: unsupervised learning of low dimensional manifolds," *Journal of Machine Learning Research*, vol. 4, pp. 119–155, 2003.

[11] M. Ostendorf, P. J. Price, and S. Shattuck-Hufnagel, "The Boston University Radio News Corpus," Tech. Rep. ECS-95-001, Boston University, March 1995.

[12] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, Inc., Boston, second edition, 1990.

[13] Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. Le Roux, and M. Ouimet, "Out-of-sample extensions for LLE, Isomap, MDS, eigenmaps, and spectral clustering," in *Advances in Neural Information Processing Systems*. 2004, vol. 16, MIT Press.