Laplacian Eigenmaps for Dimensionality Reduction and Data Representation

Mikhail Belkin
misha@math.uchicago.edu
Department of Mathematics, University of Chicago, Chicago, IL 60637, U.S.A.

Partha Niyogi
niyogi@cs.uchicago.edu
Department of Computer Science and Statistics, University of Chicago,
Chicago, IL 60637 U.S.A.

Presented by Daniel Boley

Paper Goals

- Dimensionality Reduction
- Assume data lies on a manifold in high dimensional space
- Maintain locality in reduced representation
- Theory related spectral graph partitioning
- Also related to Local Linear Embedding.

Motivating Examples

- images of same object/scene from different angles
- common motifs repeated in data.
- PCA is too rigid, does not admit non-affine manifolds.

Algorithm

1. Induce a neighborhood graph over the data:
   - vertices $\longleftrightarrow$ sample points
   - edges $\longleftrightarrow$ join points which are neighbors.

2. Assign suitable weights for the edges.

3a. Compute eigenvectors for graph Laplacian.

3b. Project points onto m leading eigenvectors.

Algorithm - Step 1

1. Step 1 (constructing the adjacency graph). We put an edge between nodes i and j if $\mathbf{x}_i$ and $\mathbf{x}_j$ are "close." There are two variations:

   (a) ϵ-neighborhoods (parameter $\epsilon \in \mathbb{R}$). Nodes i and j are connected by an edge if $\|\mathbf{x}_i - \mathbf{x}_j\|^2 < \epsilon$ where the norm is the usual Euclidean norm in $\mathbb{R}^l$. *Advantages*: Geometrically motivated, the relationship is naturally symmetric. *Disadvantages*: Often leads to graphs with several connected components, difficult to choose ϵ.

   (b) n nearest neighbors (parameter $n \in \mathbb{N}$). Nodes i and j are connected by an edge if i is among n nearest neighbors of j or j is among n nearest neighbors of i. Note that this relation is symmetric. *Advantages*: Easier to choose; does not tend to lead to disconnected graphs. *Disadvantages*: Less geometrically intuitive.

Algorithm - Step 2

2. Step 2 (choosing the weights).[1] Here as well, we have two variations for weighting the edges:

   (a) Heat kernel (parameter $t \in \mathbb{R}$). If nodes i and j are connected, put
   $$W_{ij} = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{t}};$$
   otherwise, put $W_{ij} = 0$. The justification for this choice of weights will be provided later.

   (b) Simple-minded (no parameters ($t = \infty$)). $W_{ij} = 1$ if vertices i and j are connected by an edge and $W_{ij} = 0$ if vertices i and j are not connected by an edge. This simplification avoids the need to choose t.

Algorithm - Step 3 prep

3. Step 3 (eigenmaps). Assume the graph G, constructed above, is connected. Otherwise, proceed with step 3 for each connected component. Compute eigenvalues and eigenvectors for the generalized eigenvector problem,

$$L\mathbf{f} = \lambda D\mathbf{f}, \tag{2.1}$$

where D is diagonal weight matrix, and its entries are column (or row, since W is symmetric) sums of W, $D_{ii} = \sum_j W_{ji}$. $L = D - W$ is the Laplacian matrix. Laplacian is a symmetric, positive semidefinite matrix that can be thought of as an operator on functions defined on vertices of G.

Let $\mathbf{f}_0, \ldots, \mathbf{f}_{k-1}$ be the solutions of equation 2.1, ordered according to their eigenvalues:

Algorithm - Step 3

$$
\begin{aligned}
L\mathbf{f}_0 &= \lambda_0 D\mathbf{f}_0 \\
L\mathbf{f}_1 &= \lambda_1 D\mathbf{f}_1 \\
&\cdots \\
L\mathbf{f}_{k-1} &= \lambda_{k-1} D\mathbf{f}_{k-1} \\
0 = \lambda_0 \leq \lambda_1 &\leq \cdots \leq \lambda_{k-1}.
\end{aligned}
$$

We leave out the eigenvector $\mathbf{f}_0$ corresponding to eigenvalue 0 and use the next m eigenvectors for embedding in m-dimensional Euclidean space:

$$\mathbf{x}_i \to (\mathbf{f}_1(i), \ldots, \mathbf{f}_m(i)).$$

Example - Swiss Roll

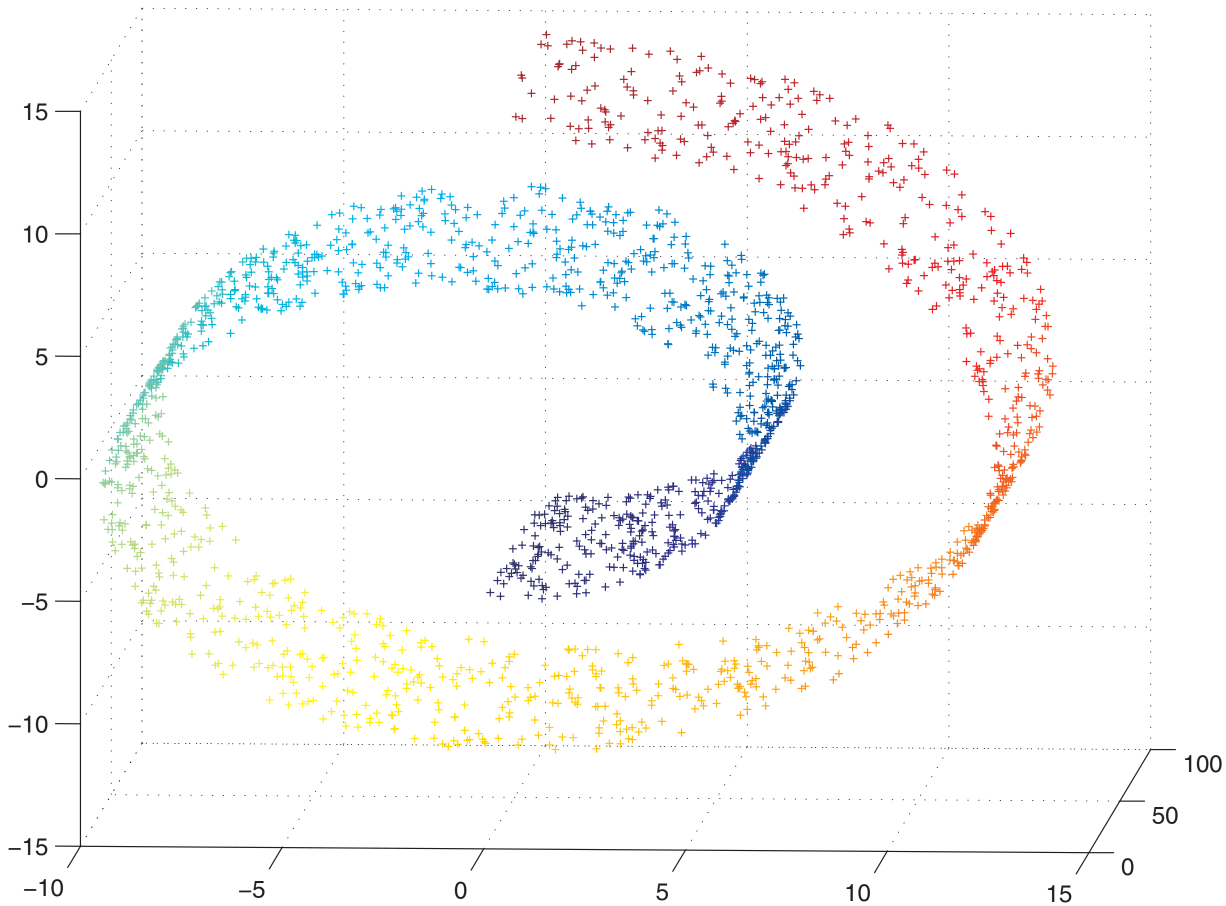

Figure 1: 2000 Random data points on the swiss roll.

Swiss Roll Projections

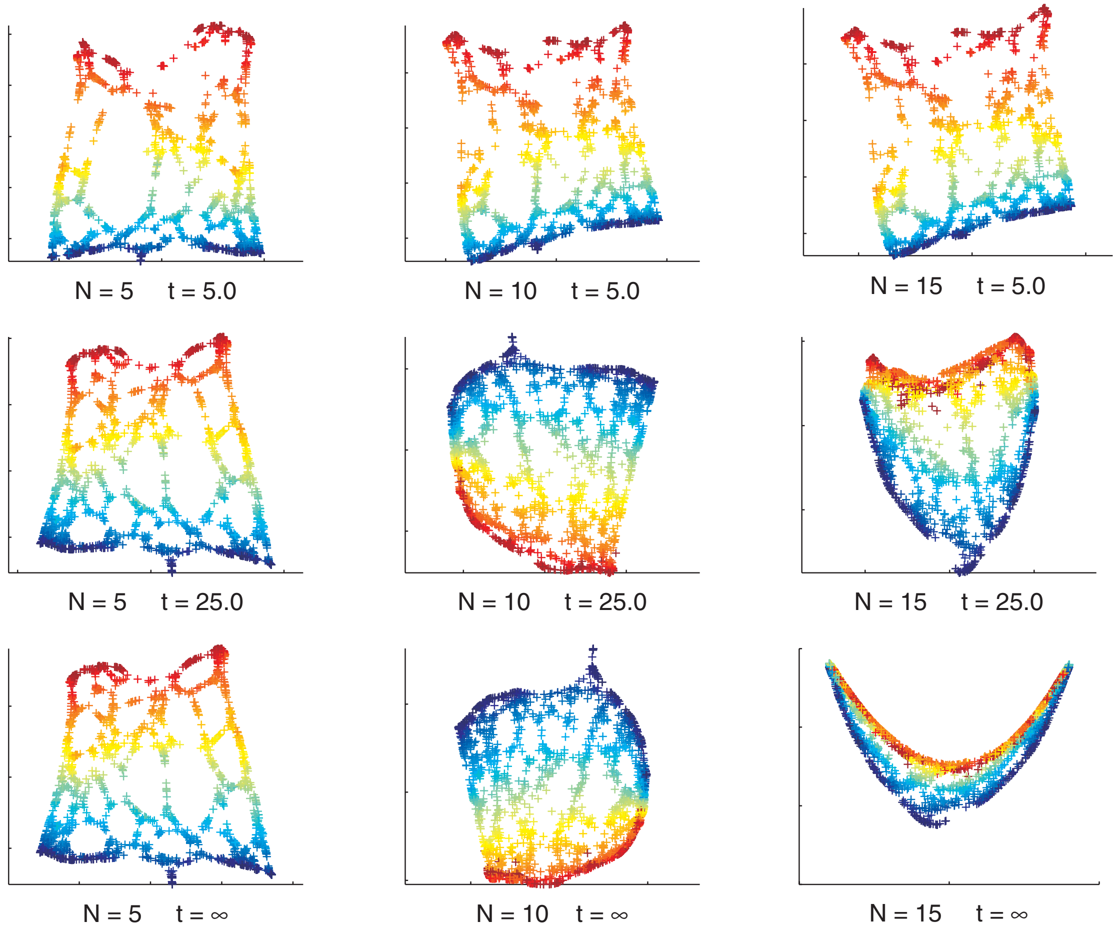

Figure 2: Two-dimensional representations of the swiss roll data, for different values of the number of nearest neighbors N and the heat kernel parameter t. $t = \infty$ corresponds to the discrete weights.

Example: Bars

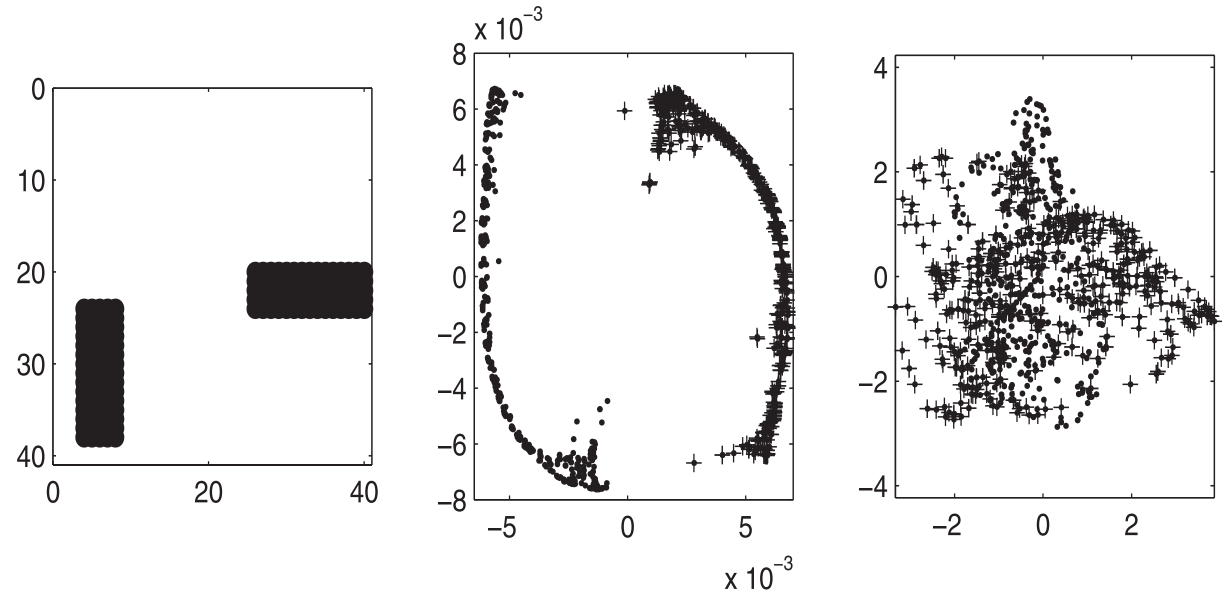

Figure 3: (Left) A horizontal and a vertical bar. (Middle) A two-dimensional representation of the set of all images using the Laplacian eigenmaps. (Right) The result of PCA using the first two principal directions to represent the data. Blue dots correspond to images of vertical bars, and plus signs correspond to images of horizontal bars.

300 Words (bigrams)

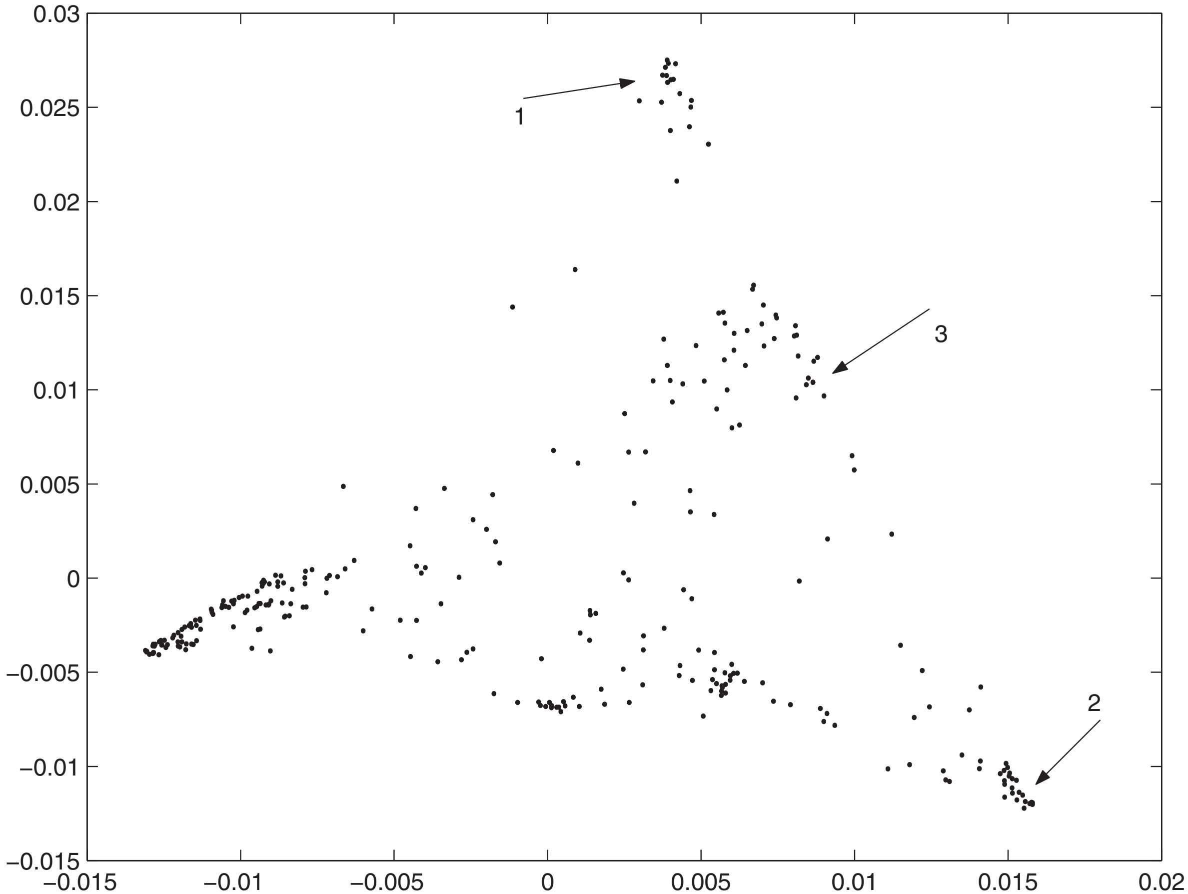

Figure 4: The 300 most frequent words of the Brown corpus represented in the spectral domain.

Some specific words

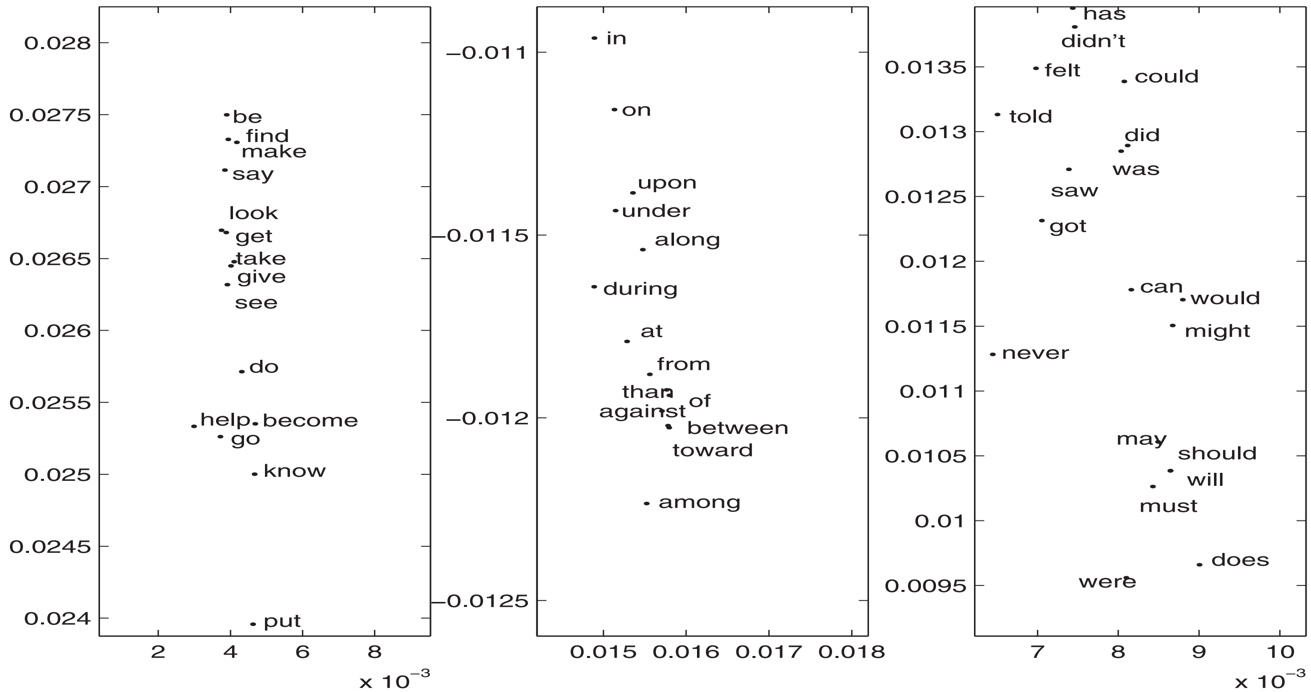

Figure 5: Fragments labeled by arrows: (left) infinitives of verbs, (middle) prepositions, and (right) mostly modal and auxiliary verbs. We see that syntactic structure is well preserved.

Example - Speech Sounds

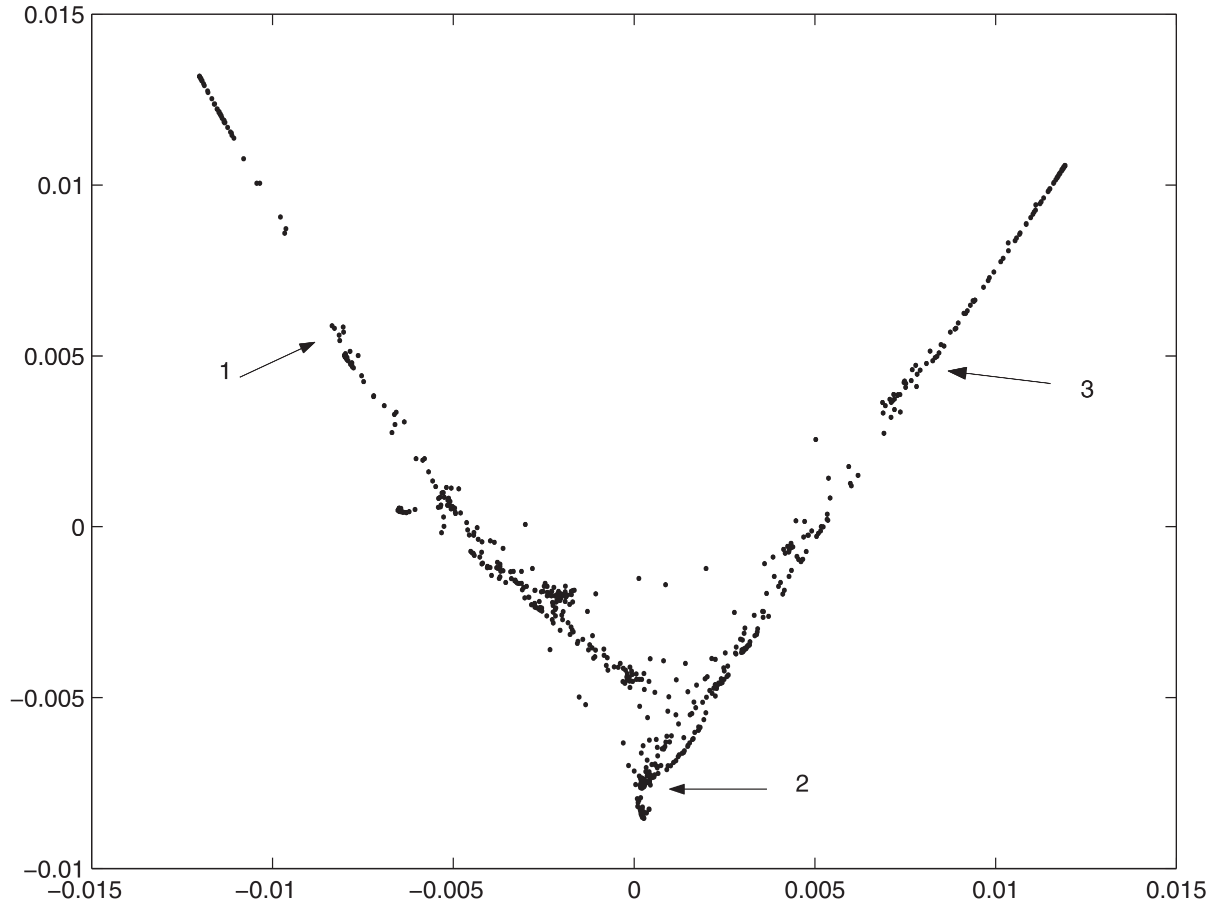

Figure 6: The 685 speech data points plotted in the two-dimensional Laplacian spectral representation.

Some Specific Sounds

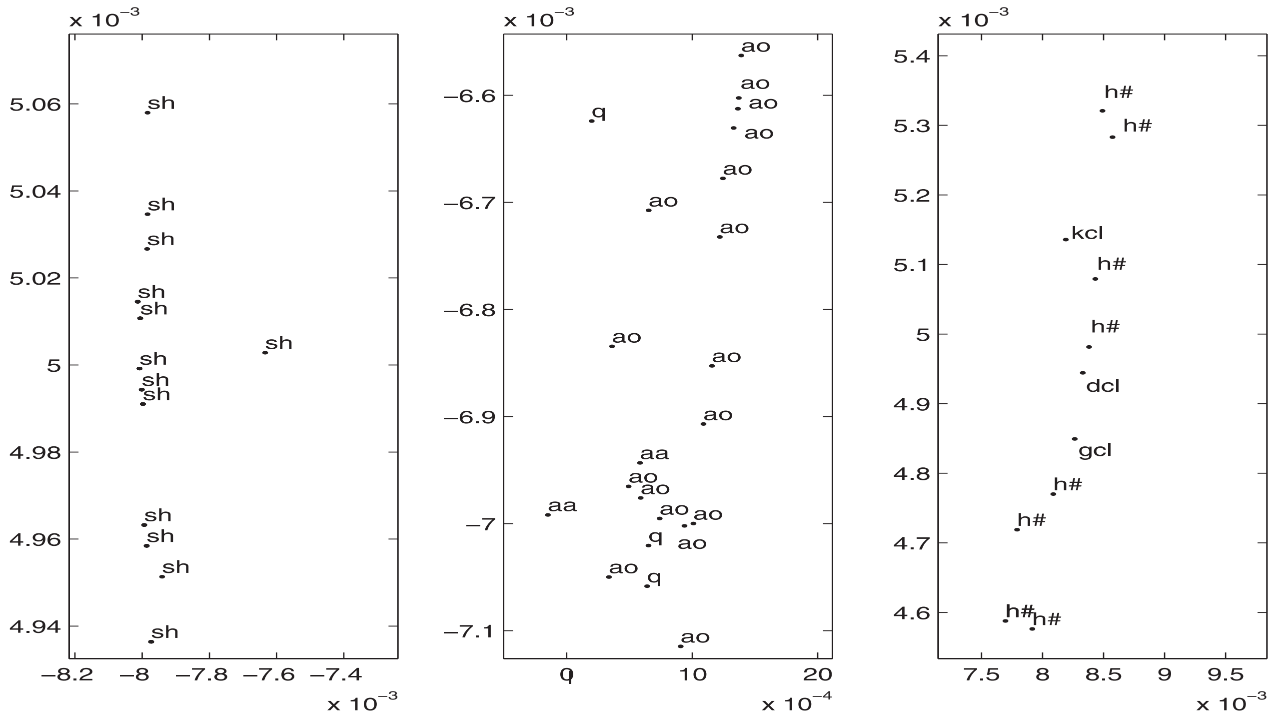

Figure 7: A blowup of the three selected regions corresponding to the arrows in Figure 6. Notice the phonetic homogeneity of the chosen regions. The data points corresponding to the same region have similar phonetic identity, though they may (and do) arise from occurrences of the same phoneme at different points in the utterance. The symbol *sh* stands for the fricative in the word *she*; *aa* and *ao* stand for vowels in the words *dark* and *all*, respectively; *kcl*, *dcl*, and *gcl* stand for closures preceding the stop consonants *k*, *d*, *g*, respectively. *h#* stands for silence.