A Probabilistic Perspective on Spectral Dimensionality Reduction

Neil D. Lawrence

Departments of Neuro- and Computer Science, University of Sheffield, U.K.
Challenges of Data Visualization Workshop, NIPS 2010

11th December 2010

Outline

Notation

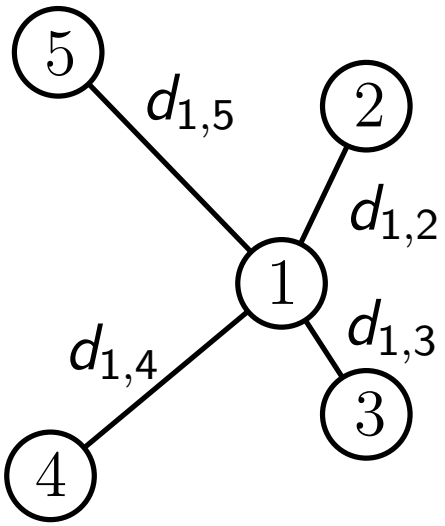
p	data dimensionality	
q	latent dimensionality	
n	number of data points	
$\mathbf{Y}$	*design matrix* containing our data	$n \times p$
$\mathbf{X}$	matrix of latent variables	$n \times q$
$\mathbf{D}$	matrix of interpoint squared distances	$n \times n$
$\mathbf{K}$	similarities/covariance/kernel	$n \times n$
$\mathbf{L}$	Laplacian matrix	$n \times n$

Row vector from matrix $\mathbf{A}$ given by $\mathbf{a}_{i,:}$ column vector $\mathbf{a}_{:,j}$ and element given by $a_{i,j}$.

Outline

Maximum Variance Unfolding

- Maximize $\mathrm{tr}(\mathbf{K})$.: equivalent to maximizing distances between non-neighbors.[1].



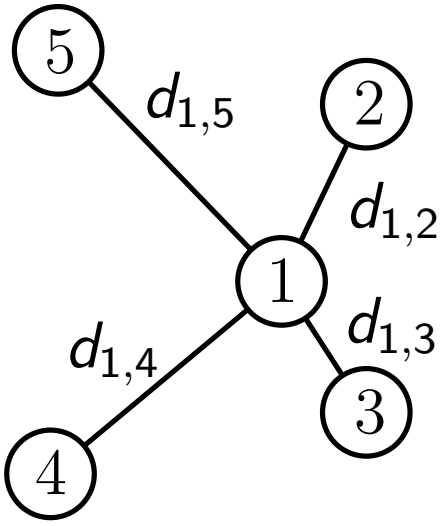
- MVU constrains "feature space" distances to be equal to observed

$$d_{i,j} = k_{i,i} - 2k_{i,j} + k_{j,j}$$

[1] The trace is the *total variance* of the data in feature space

Maximum Entropy Unfolding

- Maximize entropy of distribution subject to constraints on *moments*.

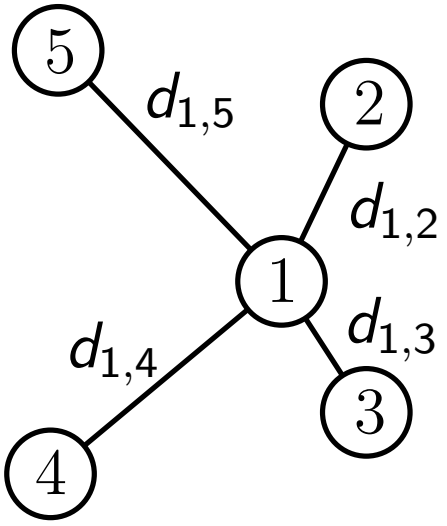


- MEU constraints are on expected distances between neighbors.

$$d_{i,j} = \left\langle \mathbf{y}_{i,:}^\top \mathbf{y}_{i,:} \right\rangle - 2 \left\langle \mathbf{y}_{i,:}^\top \mathbf{y}_{j,:} \right\rangle + \left\langle \mathbf{y}_{j,:}^\top \mathbf{y}_{j,:} \right\rangle$$

Maximum Entropy Unfolding

- Maximize entropy of distribution subject to constraints on *moments*.



- MEU constraints are on expected distances between neighbors.

$$d_{i,j} = k_{i,i} - 2k_{i,j} + k_{j,j}$$

Maximum Entropy

- Maximum entropy distribution.

$$p(\mathbf{Y}) \propto \exp\left(-\frac{1}{2}\text{tr}\left(\gamma \mathbf{Y}\mathbf{Y}^\top\right)\right) \exp\left(-\frac{1}{2}\sum_i \sum_{j \in \mathcal{N}(i)} \lambda_{i,j} d_{i,j}\right)$$

$\mathcal{N}(i)$ is neighborhood, $\{\lambda_{i,j}\}$, Lagrange multipliers.

Maximum Entropy

- Maximum entropy distribution.

$$p(\mathbf{Y}) \propto \exp\left(-\frac{1}{2}\mathrm{tr}\left(\gamma \mathbf{Y}\mathbf{Y}^\top\right) - \frac{1}{4}\mathrm{tr}\left(\mathbf{\Lambda}\mathbf{D}\right)\right)$$

$\mathcal{N}(i)$ is neighborhood, $\{\lambda_{i,j}\}$, Lagrange multipliers. Lagrange multipliers in sparse matrix $\mathbf{\Lambda}$.

Maximum Entropy

- Maximum entropy distribution.

$$p(\mathbf{Y}) = \frac{|\mathbf{L} + \gamma \mathbf{I}|^{\frac{1}{2}}}{(2\pi)^{\frac{np}{2}}} \exp\left(-\frac{1}{2}\mathrm{tr}\left((\mathbf{L} + \gamma \mathbf{I})\mathbf{Y}\mathbf{Y}^\top\right)\right)$$

$\mathcal{N}(i)$ is neighborhood, $\{\lambda_{i,j}\}$, Lagrange multipliers. Introduce Laplacian: $\ell_{i,j} = -\lambda_{i,j}$, $\ell_{i,i} = \sum_{j \in \mathcal{N}(i)} \lambda_{i,j}$, $\mathbf{L1} = \mathbf{0}$.

Outline

Relationship to Laplacian Eigenmaps

- Eigendecomposition of the covariance is

$$\mathbf{K} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$$

- Eigendecomposition of the Laplacian is

$$\mathbf{L} = \mathbf{U}\left(\mathbf{\Lambda}^{-1} - \gamma\mathbf{I}\right)\mathbf{U}^\top$$

- Principal eigenvalues of $\mathbf{K}$ are smallest eigenvalues of $\mathbf{L}$.
- (smallest eigenvalue of $\mathbf{L}$ is zero)
- Laplacian eigenmaps also suggests normalization of Laplacian.

Isomap

- Isomap (Tenenbaum et al., 2000) follows the CMDS framework.
- Sparse graph of distances is created.
- Fill in graph for non-neighbors with a shortest path algorithm.
- Element-wise square the matrix.
- Process this in the usual manner.

Locally Linear Embedding

- Factorize the Laplacian as

$$\mathbf{L} = \mathbf{M}\mathbf{M}^\top$$

- Now constrain $\mathbf{M}^\top \mathbf{1} = \mathbf{0}$ giving $\mathbf{L}\mathbf{1} = \mathbf{0}$.
  - i.e. $m_{i,i} = -\sum_{j \in \mathcal{N}(i)} m_{j,i}$
  - Set $m_{j,i} = 0$ if $j \notin \mathcal{N}(i)$.

Locally Linear Embedding

- Locally linear embeddings (Roweis and Saul, 2000) are then a specific case of MEU where
  1. The diagonal sums, $m_{i,i}$, are further constrained to unity.
  2. Model parameters found by maximizing *pseudolikelihood* of the data.

Point One

- For unit diagonals we have $\mathbf{M} = \mathbf{I} - \mathbf{W}$.
- Here the off diagonal sparsity pattern of $\mathbf{W}$ matches $\mathbf{M}$.
- Thus
$$(\mathbf{I} - \mathbf{W})^\top \mathbf{1} = \mathbf{0}.$$
- LLE proscribes that the smallest eigenvectors of
$$(\mathbf{I} - \mathbf{W})(\mathbf{I} - \mathbf{W})^\top = \mathbf{M}\mathbf{M}^\top = \mathbf{L}$$
(like Laplacian Eigenmaps).
- Equivalent to CMDS on the GRF described by $\mathbf{L}$.

Second Point

- Pseudolikelihood approximation (see e.g. Koller and Friedman, 2009, pg 970): product of the conditional densities:

$$p(\mathbf{Y}) \approx \prod_{i=1}^{n} p(\mathbf{y}_{i,:}|\mathbf{Y}_{\setminus i}),$$

  $\mathbf{Y}_{\setminus i}$ represents data other than the ith point.
- True likelihood is proportional to this but requires renormalization.
- In pseudolikelihood normalization is ignored.

Pseudolikelihood Approximation

- Optimizing the pseudolikelihood is equivalent to optimizing

$$\log p(\mathbf{Y}) \approx \sum_{i=1}^{n} \log p(\mathbf{y}_{i,:}|\mathbf{Y}_{\setminus i})$$

  equivalent to solving n independent regression problems.
- A natural constraint that the regression weights that they sum to one.
- This is how parameters in LLE (Roweis and Saul, 2000) are optimized.
- Constraint arises because $w_{j,i}/m_{i,i}$ and $m_{i,i} = \sum_{j\in\mathcal{N}(i)} w_{j,i}$.
- In LLE a *further* constraint is imposed $m_{i,i} = 1$.

LLE and PCA

- LLE is motivated by considering local linear embeddings of the data.
- Interestingly, as we increase the neighborhood size to $K = n - 1$ we do not recover PCA.
- Strange because PCA is the optimal linear embedding of the data under linear Gaussian constraints.
- But LLE is optimizing a pseudolikelihood: in contrast the MEU algorithm, which LLE approximates, does recover PCA when $K = n - 1$.

Outline

Second Point

- Pseudolikelihood approximation becomes exact if $\mathbf{M}$ is lower triangular, i.e. $\mathbf{M} = \mathbf{U}^\top$.
- log determinant is then

$$\log |\mathbf{L}| = \log \left| \mathbf{U}^\top \mathbf{U} \right| 2 \sum_i \log u_{i,i}$$

- We can force this constraint, but only considering neighbors with an index greater than the node number.
- Then follow a normal LLE procedure.

Outline

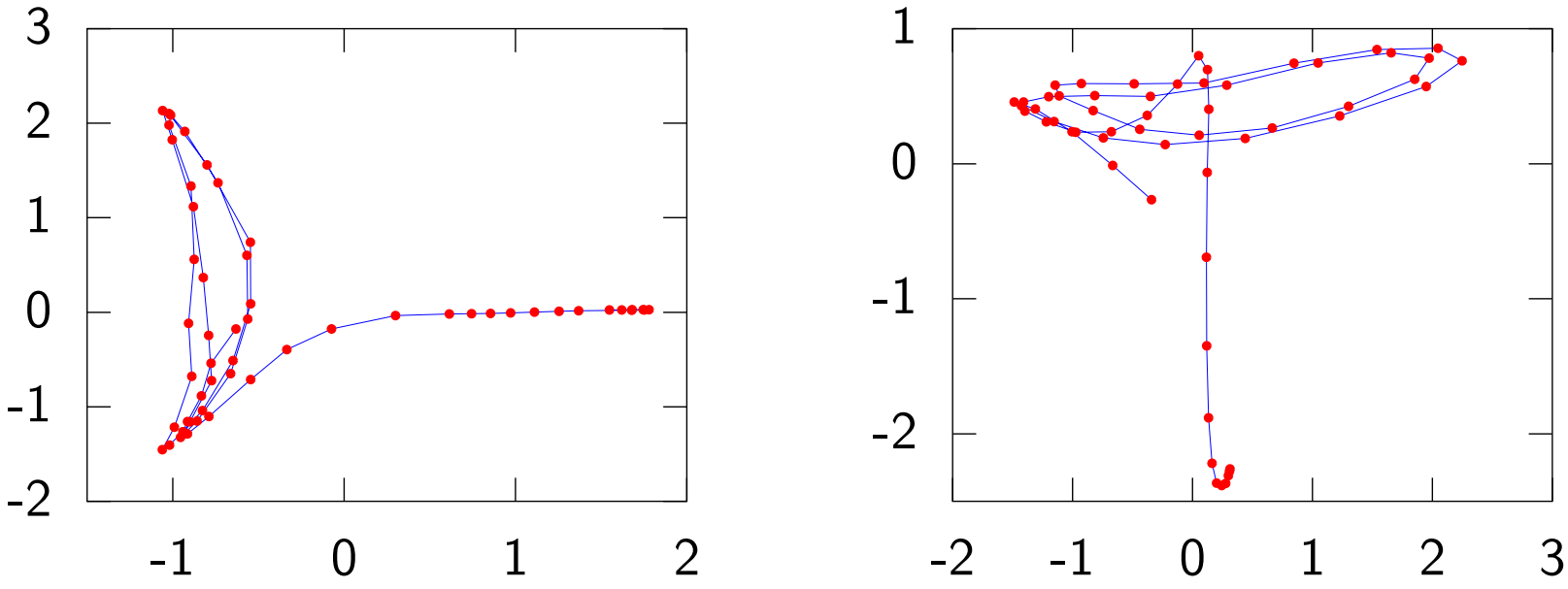
Simple Experiments

- Consider two real data sets.
- We apply each of the spectral methods we have reviewed.
- Apply the MEU framework.
- Follow the suggestion of Harmeling (Harmeling, 2007) and use the GPLVM likelihood (Lawrence, 2005) for embedding quality.
- The higher the likelihood the better the embedding.

Motion Capture Data

- Data consists of a 3-dimensional point cloud of the location of 34 points from a subject performing a run.
- 102 dimensional data set containing 55 frames of motion capture.
- Subject begins the motion from stationary and takes approximately three strides of run.
- Should see this structure in the visualization: a starting position followed by a series of loops.
- Data was made available by Ohio State University.
- The two dominant eigenvectors are visualized in following figures.

Laplacian Eigenmaps and LLE

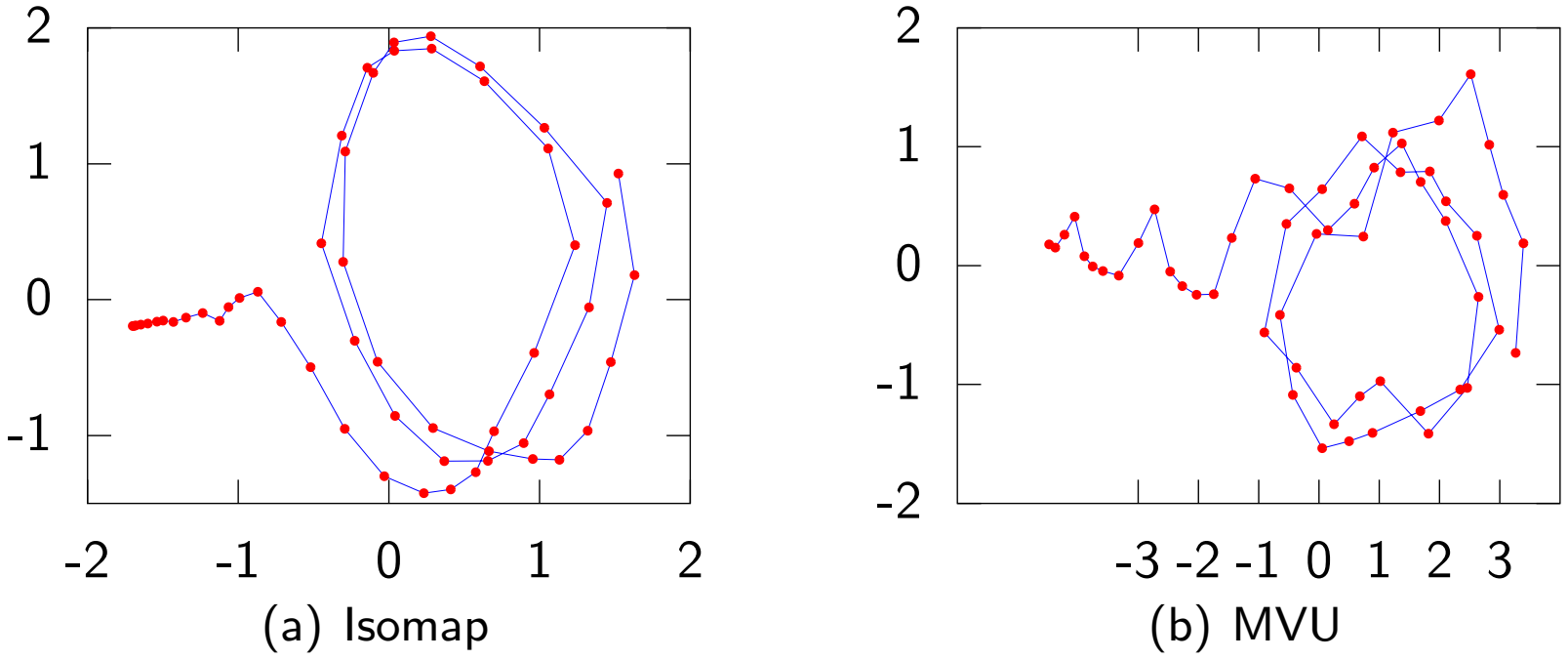


(a) Laplacian Eigenmaps

(b) Locally Linear Embedding

Figure: Models capture either the cyclic structure or the structure associated with the start of the run or both parts.

Isomap and MVU



(a) Isomap (b) MVU

Figure: Models capture either the cyclic structure or the structure associated with the start of the run or both parts.

MEU and ALLE

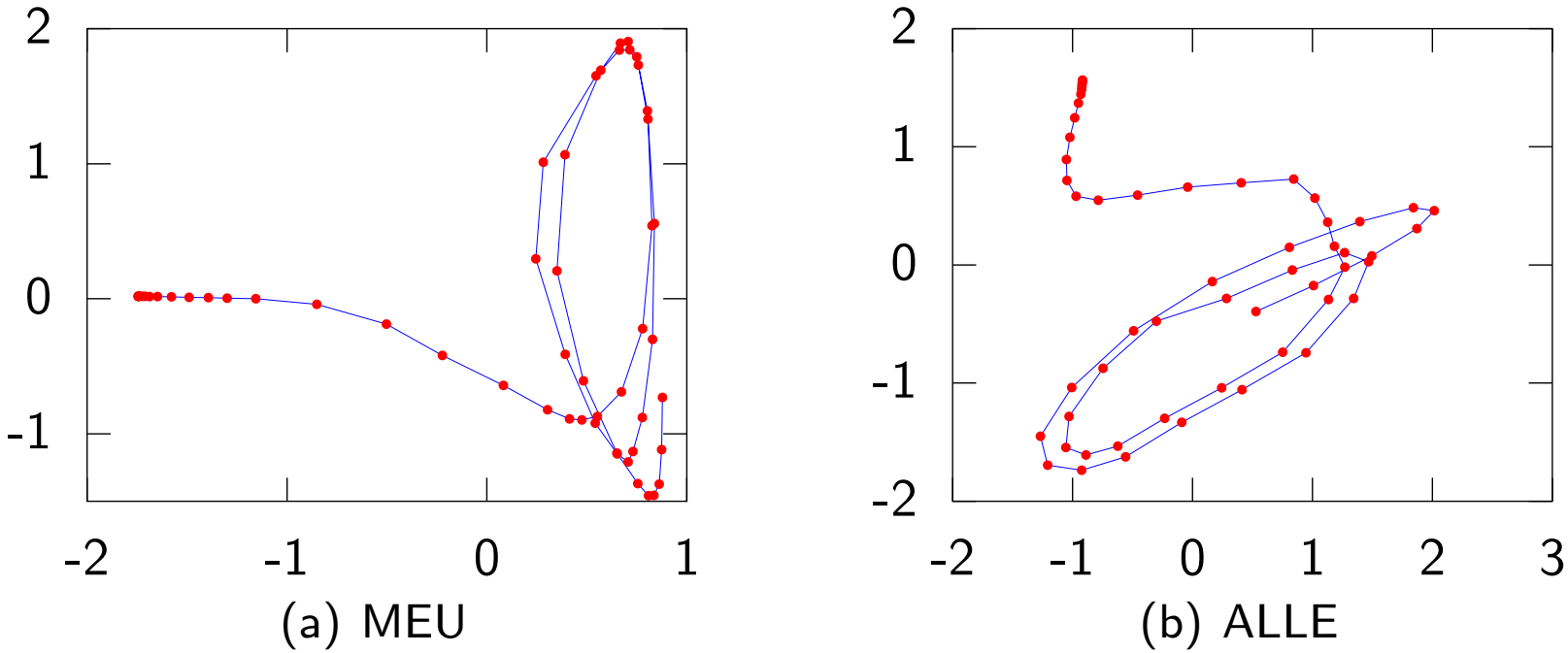


Figure: Models capture either the cyclic structure or the structure associated with the start of the run or both parts.

Motion Capture: Model Scores

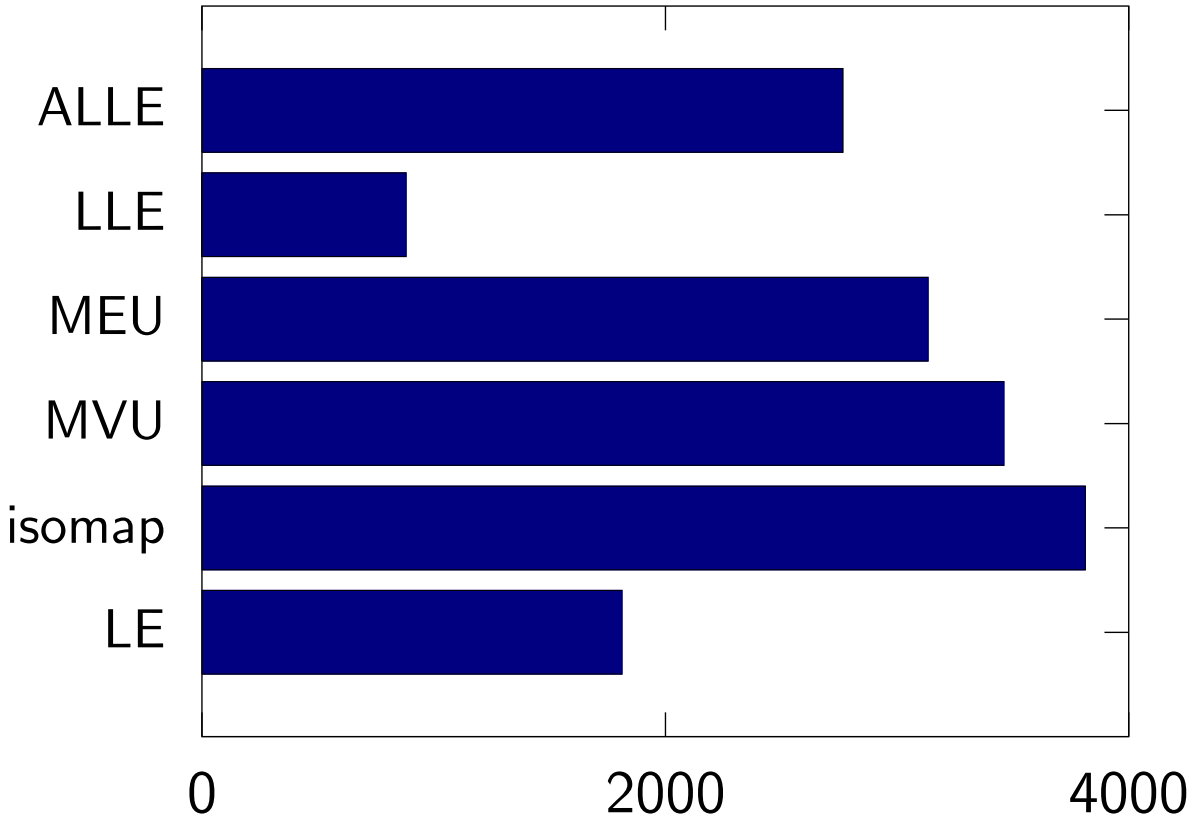


Figure: Model score for the different spectral approaches.

Outline

Discussion

- New perspective on dimensionality reduction algorithms based around maximum entropy.
- Start with MVU and end with GRFs.
- Hope that this perspective on dimensionality reduction will encourage new strands of research at the interface of these areas.

Stages of Spectral Dimensionality Reduction

- Our perspective shows there are three separate stages used in existing spectral dimensionality algorithms.
  1. A neighborhood between data points is selected. Normally k-nearest neighbors or similar algorithms are used.
  2. Interpoint distances between neighbors are fed to the algorithms which provide a similarity matrix. The way the entries in the similarity matrix are computed is the main difference between the different algorithms.
  3. The relationship between points in the similarity matrix is visualized using the eigenvectors of the similarity matrix.

Our Perspective

- Each step is somewhat orthogonal.
- Neighborhood relations need not come from nearest neighbors: can use structure learning.
- Main difference between approaches is how similarity matrix entries are determiend.
- Final step attempts to visualize the similarity using eigenvectors. This is just one possible approach.
- There is an entire field of graph visualization proposing different approaches to visualizing such graphs.

Advantages of Existing Approaches

- Conflating the three steps allows faster complete algorithms.
- E.g. mixing 2nd & 3rd allows speed ups by never computing the similarity matrix.
- We still can understand the algorithm from the unifying perspective while exploiting the computational advantages offered by this neat shortcut.

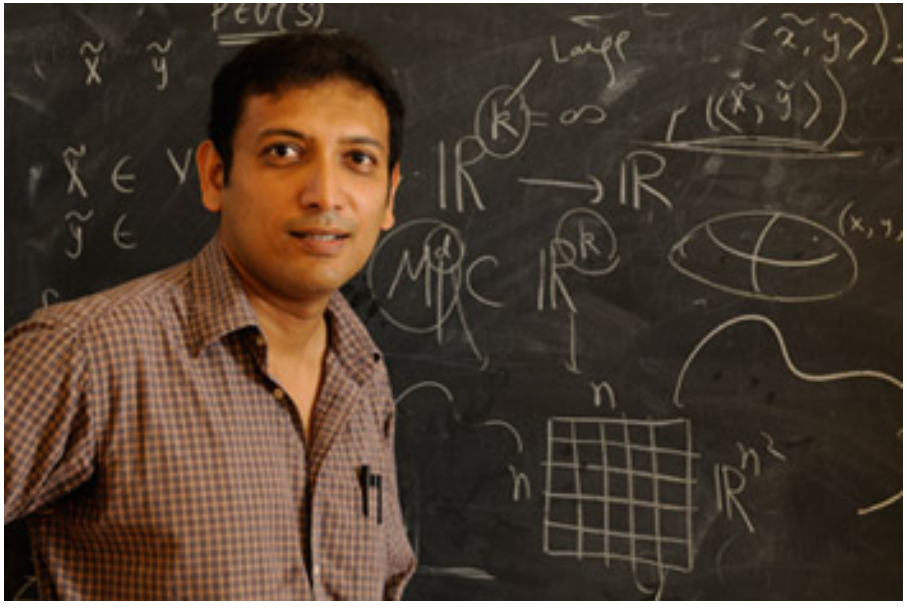
Other Points

- ALLE may provide a good compromise in speed vs accuracy.
- Also looked at structural learning.
  - See `http://arxiv.org/abs/1010.4830` for more details.

Acknowledgements

Conversations with John Kent, Chris Williams, Brenden Lake, Joshua Tenenbaum and John Lafferty have influenced the thinking in this work.

Partha and Sam



References I

S. Harmeling. Exploring model selection techniques for nonlinear dimensionality reduction. Technical Report EDI-INF-RR-0960, University of Edinburgh,

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009. [Google Books] .

N. D. Lawrence. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *Journal of Machine Learning Research*, 6:1783–1816, 11 2005.

S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500): 2323–2326, 2000. [DOI].

J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. [DOI].