

Tutorial - Random Walks on Graphs Large-time Behavior and Applications to Analysis of Large Data Sets

Mauro Maggioni

Mathematics and Computer Science
Duke University

I.P.A.M., 09/10/08

In collaboration with R.R. Coifman, P.W. Jones, Y-M. Jung, R. Schul,
A.D. Szlam

Funding: ONR, NSF

- Setting and Motivation
- Diffusion on Graphs
- Large time diffusion
 - Eigenfunctions, spectral embeddings
 - Good parametrizations of manifolds, heat kernels
- Machine learning example: semisupervised learning
- Conclusion

Structured data in high-dimensional spaces

A deluge of data: documents, web searching, customer databases, hyper-spectral imagery (satellite, biomedical, etc...), social networks, gene arrays, proteomics data, neurobiological signals, sensor networks, financial transactions, traffic statistics (automobilistic, computer networks)...

Common feature/assumption: data is given in a high dimensional space, however it has a much lower dimensional intrinsic geometry.

- (i) physical constraints. For example the effective state-space of at least some proteins seems low-dimensional, at least when viewed at the time scale when important processes (e.g. folding) take place.
- (ii) statistical constraints. For example many dependencies among word frequencies in a document corpus force the distribution of word frequency to low-dimensional, compared to the dimensionality of the whole space.

Structured data in high-dimensional spaces

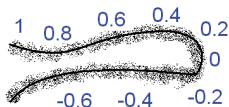
A deluge of data: documents, web searching, customer databases, hyper-spectral imagery (satellite, biomedical, etc...), social networks, gene arrays, proteomics data, neurobiological signals, sensor networks, financial transactions, traffic statistics (automobilistic, computer networks)...

Common feature/assumption: data is given in a high dimensional space, however it has a much lower dimensional intrinsic geometry.

- (i) physical constraints. For example the effective state-space of at least some proteins seems low-dimensional, at least when viewed at the time scale when important processes (e.g. folding) take place.
- (ii) statistical constraints. For example many dependencies among word frequencies in a document corpus force the distribution of word frequency to low-dimensional, compared to the dimensionality of the whole space.

Low-dimensional sets in high-dimensional spaces

In several instances the geometry of the data can help construct useful priors, for tasks such as classification, regression for prediction purposes.

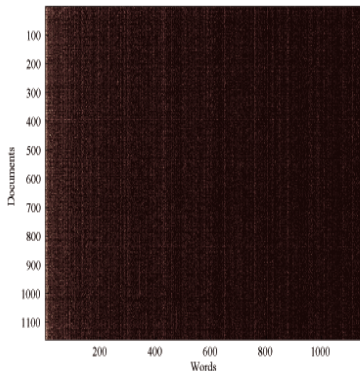


Some issues I am interested in:

- *geometric*: find intrinsic properties, such as local dimensionality, and local parameterizations.
- *approximation theory*: approximate functions on such data, respecting the geometry.

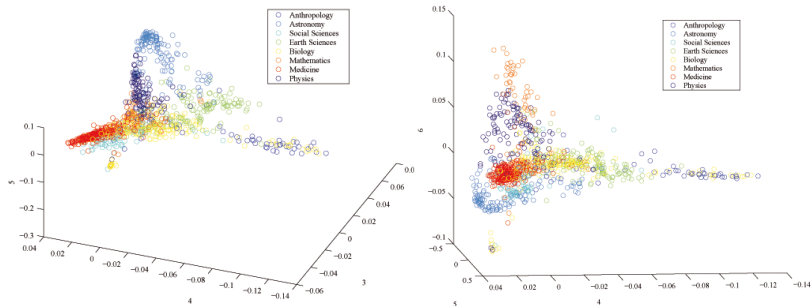
Text documents

About 1100 Science News articles, from 8 different categories. We compute about 1000 coordinates, i -th coordinate of document d represents frequency in document d of the i -th word in a fixed dictionary.



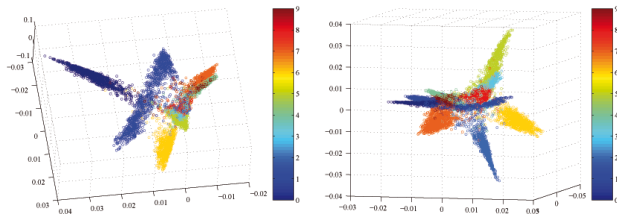
Text documents

About 1100 Science News articles, from 8 different categories. We compute about 1000 coordinates, i -th coordinate of document d represents frequency in document d of the i -th word in a fixed dictionary.



Handwritten Digits

Data base of about 60,000 28×28 gray-scale pictures of handwritten digits, collected by USPS. Point cloud in R^{28^2} .
Goal: automatic recognition.

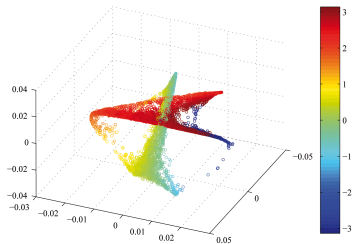
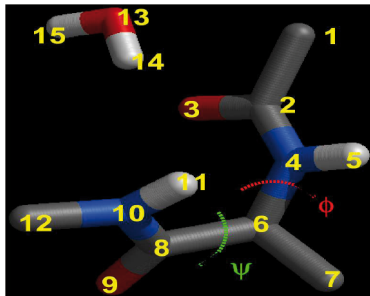


Set of 10,000 picture (28 by 28 pixels) of 10 handwritten digits. Color represents the label (digit) of each point.

A simple example from Molecular Dynamics

[Joint with C. Clementi]

The dynamics of a small protein (12 atoms, H atoms removed) in a bath of water molecules is approximated by a Langevin system of stochastic equations $\dot{x} = -\nabla U(x) + \dot{w}$. The set of states of the protein is a noisy (\dot{w}) set of points in \mathbb{R}^{36} .



Left: representation of an alanine dipeptide molecule. Right: embedding of the set of configurations.

We start by analyzing the intrinsic geometry of the data, and then working on function approximation *on* the data.

- Find parametrizations for the data: manifold learning, dimensionality reduction. Ideally: number of parameters comparable with the intrinsic dimensionality of data + a parametrization should approximately preserve distances + be stable under perturbations/noise
- Construct useful dictionaries of functions on the data: approximation of functions on the manifold, predictions, learning.

We start by analyzing the intrinsic geometry of the data, and then working on function approximation *on* the data.

- Find parametrizations for the data: manifold learning, dimensionality reduction. Ideally: number of parameters comparable with the intrinsic dimensionality of data + a parametrization should approximately preserve distances + be stable under perturbations/noise
- Construct useful dictionaries of functions on the data: approximation of functions on the manifold, predictions, learning.

Random walks and heat kernels on the data

Assume the data $X = \{x_i\}_{i=1}^N \subset \mathbb{R}^D$. Assume we can assign local similarities via a kernel function $W(x_i, x_j) \geq 0$.

Simplest example: $W_\sigma(x_i, x_j) = e^{-\|x_i - x_j\|^2 / \sigma}$.

Model the data as a *weighted graph* (G, E, W) : vertices represent data points, edges connect x_i, x_j with weight $W_{ij} := W(x_i, x_j)$, when positive. Let $D_{ii} = \sum_j W_{ij}$ and

$$\underbrace{P = D^{-1}W}_{\text{random walk}}, \quad \underbrace{T = D^{-\frac{1}{2}}WD^{-\frac{1}{2}}}_{\text{symm. "random walk"}}, \quad \underbrace{H = e^{-tL}}_{\text{Heat kernel}}$$

Here $L = I - T$ is the normalized Laplacian.

Note 1: W depends on the type of data.

Note 2: W should be “local”, i.e. close to 0 for points not sufficiently close.

Random walks and heat kernels on the data

Assume the data $X = \{x_i\}_{i=1}^N \subset \mathbb{R}^D$. Assume we can assign local similarities via a kernel function $W(x_i, x_j) \geq 0$.

Simplest example: $W_\sigma(x_i, x_j) = e^{-\|x_i - x_j\|^2 / \sigma}$.

Model the data as a *weighted graph* (G, E, W) : vertices represent data points, edges connect x_i, x_j with weight $W_{ij} := W(x_i, x_j)$, when positive. Let $D_{ii} = \sum_j W_{ij}$ and

$$\underbrace{P = D^{-1}W}_{\text{random walk}}, \quad \underbrace{T = D^{-\frac{1}{2}}WD^{-\frac{1}{2}}}_{\text{symm. "random walk"}}, \quad \underbrace{H = e^{-tL}}_{\text{Heat kernel}}$$

Here $L = I - T$ is the normalized Laplacian.

Note 1: W depends on the type of data.

Note 2: W should be “local”, i.e. close to 0 for points not sufficiently close.

Random walks and heat kernels on the data

Assume the data $X = \{x_i\}_{i=1}^N \subset \mathbb{R}^D$. Assume we can assign local similarities via a kernel function $W(x_i, x_j) \geq 0$.

Simplest example: $W_\sigma(x_i, x_j) = e^{-\|x_i - x_j\|^2 / \sigma}$.

Model the data as a *weighted graph* (G, E, W) : vertices represent data points, edges connect x_i, x_j with weight $W_{ij} := W(x_i, x_j)$, when positive. Let $D_{ii} = \sum_j W_{ij}$ and

$$\underbrace{P = D^{-1}W}_{\text{random walk}}, \quad \underbrace{T = D^{-\frac{1}{2}}WD^{-\frac{1}{2}}}_{\text{symm. "random walk"}}, \quad \underbrace{H = e^{-tL}}_{\text{Heat kernel}}$$

Here $L = I - T$ is the normalized Laplacian.

Note 1: W depends on the type of data.

Note 2: W should be “local”, i.e. close to 0 for points not sufficiently close.

Basic properties

- $P^t(x, y)$ is the probability of jumping from x to y in t steps
- $P^t(x, \cdot)$ is a “probability bump” on the graph
- P and T are similar, therefore share the same eigenvalues $\{\lambda_i\}$ and the eigenfunctions are related by a simple transformation. Let $T\varphi_i = \lambda_i\varphi_i$, with $1 = \lambda_1 \geq \lambda_2 \geq \dots$
- $\lambda_i \in [-1, 1]$
- “typically” P (or T) is large and sparse, but its high powers are full and low-rank

Functions on graphs

Any function $f : G \rightarrow \mathbb{R}$ is a vector in R^N . Euclidean norm and inner product:

$$\|f\|_2^2 = \sum_{x \in G} |f(x)|^2 d(x) \quad , \quad \langle f, g \rangle = \sum_{x \in G} f(x)g(x)d(x)$$

Other choices are possible

A Laplacian L allows to introduce a notion of smoothness

$$\langle Lf, f \rangle = \sum_x \sum_{y \sim x} W(x, y) \left(\frac{f(x)}{\sqrt{d_x}} - \frac{f(y)}{\sqrt{d_y}} \right)^2 \sim \int_{\text{edges}} |\nabla f|^2 dW$$

Moreover,

$$\lambda_i(L) = \min_{f \perp \langle \varphi_1, \dots, \varphi_{i-1} \rangle} \frac{\langle Lf, f \rangle}{\|f\|_2^2}$$

Functions on graphs

Any function $f : G \rightarrow \mathbb{R}$ is a vector in R^N . Euclidean norm and inner product:

$$\|f\|_2^2 = \sum_{x \in G} |f(x)|^2 d(x) \quad , \quad \langle f, g \rangle = \sum_{x \in G} f(x)g(x)d(x)$$

Other choices are possible

A Laplacian L allows to introduce a notion of smoothness

$$\langle Lf, f \rangle = \sum_x \sum_{y \sim x} W(x, y) \left(\frac{f(x)}{\sqrt{d_x}} - \frac{f(y)}{\sqrt{d_y}} \right)^2 \sim \int_{\text{edges}} |\nabla f|^2 dW$$

Moreover,

$$\lambda_i(L) = \min_{f \perp \langle \varphi_1, \dots, \varphi_{i-1} \rangle} \frac{\langle Lf, f \rangle}{\|f\|_2^2}$$

Dimensionality reduction and embeddings

Assume the data lies on a d -dimensional manifold \mathcal{M} in \mathbb{R}^n (think $n \gg d$): how to find a map $\mathcal{M} \rightarrow \mathbb{R}^D$, with $D \ll n$ (hopefully $D \sim d$)?

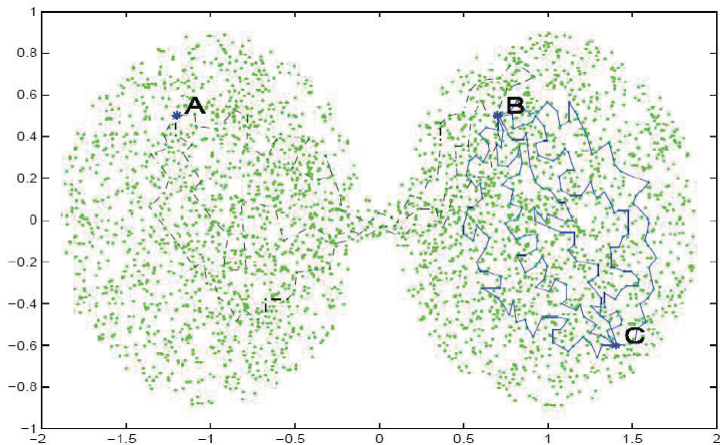
Several techniques rely on a mapping

$$x \mapsto (\varphi_i(x))_{i=1, \dots, D},$$

where φ_i are the eigenvectors of some matrix, e.g. dissimilarity matrix (CMDS), geodesic distance matrix (ISOMAP), or some local averaging operator (LLE, Laplacian eigenmap, Hessian eigenmap, etc...).

Pictures above: eigenfunctions of T : $T\varphi_i = \lambda_i\varphi_i$.

Diffusion distances



[Picture courtesy of S. Lafon]

Diffusion distances for large time

We would like to measure distances between points on a graph by random walks. Diffusion distance at time t :

$$\begin{aligned}d^{(2t)}(x, y) &= \|T^t \delta_x - T^t \delta_y\| = \|T^t(x, \cdot) - T^t(y, \cdot)\| \\&= \sqrt{\sum_{z \in G} |T^t(x, z) - T^t(y, z)|^2} \\&= \sqrt{\sum_i \lambda_i^t (\varphi_i(x) - \varphi_i(y))^2} \\&\sim \|(\lambda_i^t \varphi_i(x))_{i=1}^m - (\lambda_i^t \varphi_i(y))_{i=1}^m\|_{\mathbb{R}^m}\end{aligned}$$

Therefore $\Phi_m^{(2t)} : G \rightarrow \mathbb{R}^m$ with $\Phi_m^{(2t)}(x) = (\lambda_i^t \varphi_i(x))_{i=1}^m$ satisfies

$$\|\Phi_m^{(2t)}(x) - \Phi_m^{(2t)}(y)\|_{\mathbb{R}^m} \sim d^{(2t)}(x, y)$$

at least for t large and m large.

Go LIVE with some “simple” examples!

Spectral embeddings - what else do we know?

Many open questions about these spectral embeddings.
If we give up seeking *global* embeddings, we can prove some strong results.

Another map gains relevance, based on the heat kernel
 $e^{-t\Delta} \sim \mathcal{T}^t$.

Large portions of a manifold \mathcal{M} into \mathbb{R} , where d is *exactly* the intrinsic dimension of \mathcal{M} , can be mapped by $x \mapsto (K_t(x, x_i))_{i=1}^m$, for carefully chosen points $\{x_i\}$, depending on x , in such a way that this map has low-distortion on roughly the largest ball around x which can be at all embedded in \mathbb{R}^d with low distortion.

Connections with the continuous case

When N points are randomly sampled from a Riemannian manifold \mathcal{M} , uniformly w.r.t. volume, then the behavior of the above operators, as $N \rightarrow +\infty$, is quite well understood. In particular, T approximates the heat kernel on \mathcal{M} , and $\mathcal{L} = I - T$, the normalized Laplacian, approximates (up to rescaling), the Laplace-Beltrami operator on \mathcal{M} .

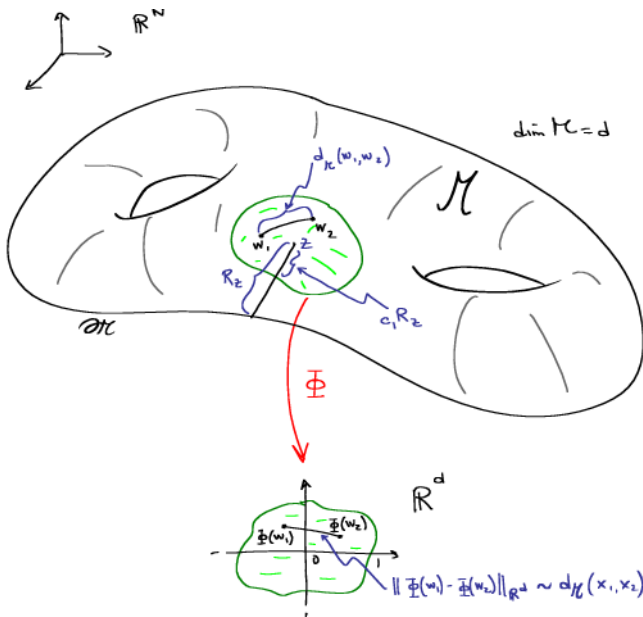
These approximations should be taken with a grain of salt: typically the number of points is not large enough to guarantee that the discrete operators above are close to their continuous counterparts.

Parametrization of point clouds

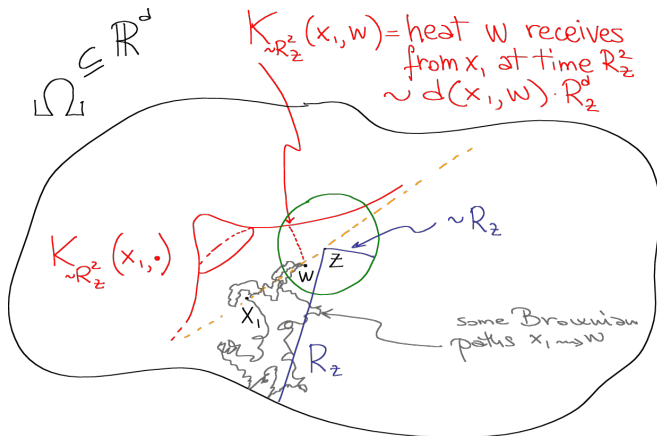
One can try to use the operator T , or its eigenfunctions, which is intrinsic to the data, to construct parametrizations of the data. This is indeed possible; in fact, we [P.W. Jones, R. Schul, MM] showed one can obtain even better parametrizations by using T itself, or heat kernels.

When the data is nonlinear, these embedding are more powerful, and have stronger guarantees, and wider applicability, when \mathcal{M} is nonlinear, of both standard linear embeddings (PCA, random projections,...) and nonlinear embeddings (ISOMAP, LLE, Hessian eigenmap, etc...).

Charts and local parametrizations, I



Charts and local parametrizations, II

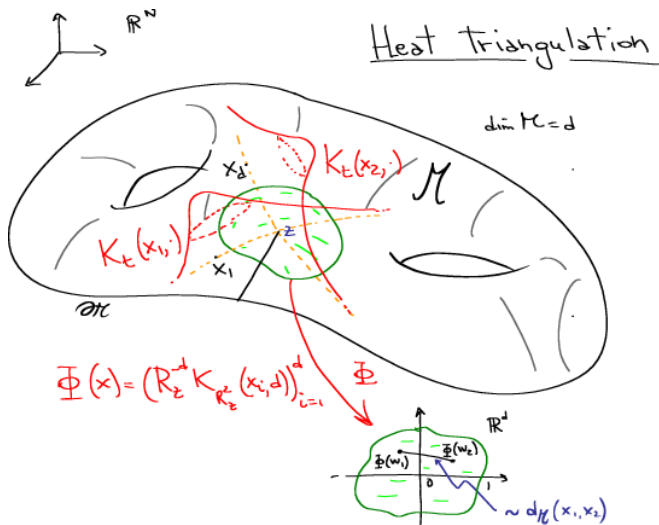


$$w \mapsto (R_z^{-d} K_{\sim R_z^2}(x_i, w))_{i=1, \dots, d}$$

for d reasonably chosen points x_1, \dots, x_d .

The heat kernel computes distances by averaging along all paths, weighted by their probability of happening (Wiener measure for Brownian motion), with paths of length $\sim d(x_i, w)$ having the highest probability.

Charts and local parametrizations, III



Note: this can be interpreted as a “kernel map” that linearizes the data to the “largest extent possible” under a distortion constraint.

Robust parametrizations through heat kernels

Theorem (Heat Triangulation Theorem - with P.W. Jones, R. Schul)

Let (\mathcal{M}, g) be a Riemannian manifold, with g at least C^α , $\alpha > 0$, and $z \in \mathcal{M}$. Let R_z be the radius of the largest ball on \mathcal{M} , centered at z , which is bi-Lipschitz equivalent to a Euclidean ball. Let p_1, \dots, p_d be d linearly independent directions. There are constants $c_1, \dots, c_5 > 0$, depending on $d, c_{\min}, c_{\max}, \|g\|_{\alpha \wedge 1}, \alpha \wedge 1$, and the smallest and largest eigenvalues of the Gramian matrix $(\langle p_i, p_j \rangle)_{i=1, \dots, d}$, such that the following holds. Let y_i be so that $y_i - z$ is in the direction p_i , with $c_4 R_z \leq d_{\mathcal{M}}(y_i, z) \leq c_5 R_z$ for each $i = 1, \dots, d$ and let $t_z = c_6 R_z^2$. The map

$$\begin{aligned} \Phi : B_{c_1 R_z}(z) &\rightarrow \mathbb{R}^d \\ x &\mapsto (R_z^d K_{t_z}(x, y_1), \dots, R_z^d K_{t_z}(x, y_d)) \end{aligned}$$

satisfies, for any $x_1, x_2 \in B_{c_1 R_z}(z)$,

$$\frac{c_2}{R_z} d_{\mathcal{M}}(x_1, x_2) \leq \|\Phi(x_1) - \Phi(x_2)\| \leq \frac{c_3}{R_z} d_{\mathcal{M}}(x_1, x_2).$$

Parametrizations through eigenfunctions

Eigenfunctions of T also can be used to obtain an embedding with similar properties.

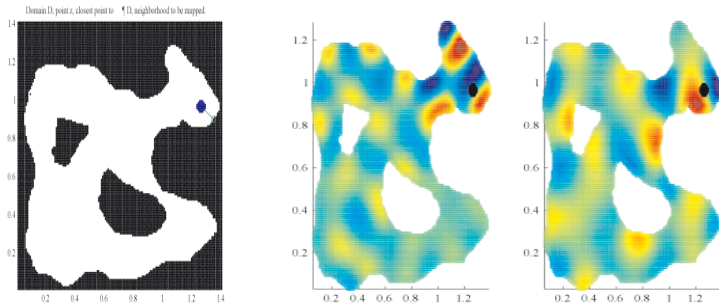


Figure: Top left: a non-simply connected domain in \mathbb{R}^2 , and the point z with its neighborhood to be mapped. Top right: the image of the neighborhood under the map. Bottom: Two eigenfunctions for mapping.

Analysis *on* the set

Equipped with good systems of coordinates on large pieces of the set, one can start doing analysis and approximation intrinsically on the set.

- *Fourier analysis on data*: use eigenfunctions for function approximation. Ok for globally uniformly smooth functions. Conjecture: most functions of interest are not in this class (Belkin, Niyogi, Coifman, Lafon).
- *Diffusion wavelets*: can construct multiscale analysis of wavelet-like functions on the set, adapted to the geometry of diffusion, at different time scales (joint with R.Coifman).
- The *diffusion semigroup* itself on the data can be used as a smoothing kernel. We recently obtained very promising results in image denoising and semisupervised learning (in a few slides, joint with A.D. Szlam and R. Coifman).

Applications

- Hierarchical organization of data and of Markov chains (e.g. documents, regions of state space of dynamical systems, etc...);
- Distributed agent control, Markov decision processes (e.g.: compression of state space and space of relevant value functions);
- Machine Learning (e.g. nonlinear feature selection, semisupervised learning through diffusion, multiscale graphical models);
- Approximation, learning and denoising of functions on graphs (e.g.: machine learning, regression, etc...)
- Sensor networks: compression of measurements collected from the network (e.g. wavelet compression on scattered sensors);
- Multiscale modeling of dynamical systems (e.g.: nonlinear and multiscale PODs);
- Compressing data and functions on the data;
- Data representation, visualization, interaction;
- ...

Semi-supervised Learning on Graphs

[Joint with A.D.Szlam]

Given:

- X : all the data points
- $(\tilde{X}, \{\chi_i(x)\}_{x \in \tilde{X}, i=1, \dots, l})$: a *small* subset of X , with labels:
 $\chi_i(x) = 1$ if x is in class i , 0 otherwise.

Objective:

- guess $\chi_i(x)$ for $x \in X \setminus \tilde{X}$.

Motivation:

- data can be cheaply acquired (X large), but it is expensive to label (\tilde{X} small). If data has useful geometry, then it is a good idea to use X to learn the geometry, and then perform regression by using dictionaries on the data, adapted to its geometry.

Algorithm:

- use the geometry of X to design a smoothing kernel (e.g. heat kernel), and apply such smoothing to the χ_i 's, to obtain $\tilde{\chi}_i$, soft class assignments on all of X . This is already pretty good.
- The key to success is to repeat: incorporate the $\tilde{\chi}_i$'s into the geometry graph, and design a new smoothing kernel \tilde{K} that takes into account the new geometry. Use \tilde{K} to smooth the initial label, to obtain final classification.

Experiments on standard data sets show this technique is very competitive.

Semi-supervised Learning on Graph (cont'd)

	FAKS	FAHC	FAEF	Best of other methods
digit1	2.0	2.1	1.9	2.5 (LapEig)
USPS	4.0	3.9	3.3	4.7 (LapRLS, Disc. Reg.)
BCI	45.5	45.3	47.8	31.4 (LapRLS)
g241c	19.8	21.5	18.0	22.0 (NoSub)
COIL	12.0	11.1	15.1	9.6 (Disc. Reg.)
gc241n	11.0	12.0	9.2	5.0 (ClusterKernel)
text	22.3	22.3	22.8	23.6 (LapSVM)

In the first column we chose, for each data set, the best performing method with model selection, among all those discussed in Chapelle's book. In each of the remaining columns we report the performance of each of our methods with model selection, but with the best settings of parameters for constructing the nearest neighbor graph, among those considered in other tables. The aim of this rather unfair comparison is to highlight the potential of the methods on the different data sets. The training set is 1/15 of the whole set.

Summary for the “Fourier part”

- it is useful to start with only local similarities between data points;
- it is possible to organize this local information by diffusion;
- parametrizations can be found by looking at the eigenvectors of a diffusion operator (Fourier modes);
- these eigenvectors yield a nonlinear embedding into low-dimensional Euclidean space;
- the eigenvectors can be used for global Fourier analysis on the set/manifold.

A (short) list of open problems

- Little is understood about global properties of global eigenfunctions
- Behavior of eigenfunctions under perturbations of the graph
- Properties of eigenfunctions on graphs which are very different from sampled manifolds
- Relationships between eigenfunctions of different Laplacians

Next: going multiscale

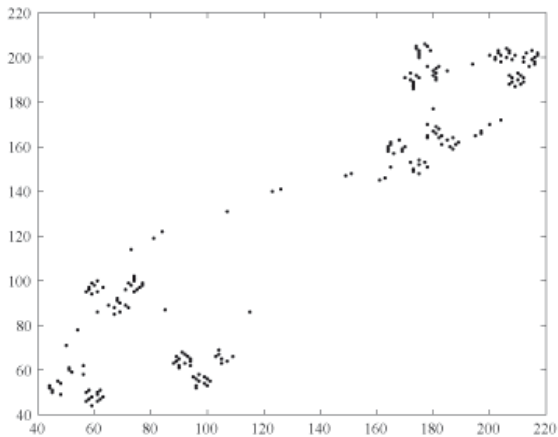
Motivation: Either very local information or very global information: in many problems the intermediate scales are very interesting! Would like **multiscale** information!

Possibility 1: proceed *bottom-up*: repeatedly cluster together in a multi-scale fashion, in a way that is faithful to the operator: diffusion wavelets.

Possibility 2: proceed *top-bottom*: cut greedily according to global information, and repeat procedure on the pieces: recursive partitioning, local cosines...

Possibility 3: do *both*!

A multiscale network



Acknowledgements

- R.R. Coifman, [Diffusion geometry; Diffusion wavelets; Uniformization via eigenfunctions; Multiscale Data Analysis], P.W. Jones (Yale Math), S.W. Zucker (Yale CS) [Diffusion geometry];
- P.W. Jones (Yale Math), R. Schul (UCLA) [Uniformization via eigenfunctions; nonhomogenous Brownian motion];
- S. Mahadevan (U.Mass CS) [Markov decision processes];
- A.D. Szlam (UCLA) [Diffusion wavelet packets, top-bottom multiscale analysis, linear and nonlinear image denoising, classification algorithms based on diffusion];
- G.L. Davis (Yale Pathology), R.R. Coifman, F.J. Warner (Yale Math), F.B. Geshwind, A. Coppi, R. DeVerse (Plain Sight Systems) [Hyperspectral Pathology];
- H. Mhaskar (Cal State, LA) [polynomial frames of diffusion wavelets];
- J.C. Bremer (Yale) [Diffusion wavelet packets, biorthogonal diffusion wavelets];
- M. Mahoney, P. Drineas (Yahoo Research) [Randomized algorithms for hyper-spectral imaging]
- J. Mattingly, S. Mukherjee and Q. Wu (Duke Math, Stat, ISDS) [stochastic systems and learning]; A. Lin, E. Monson (Duke Phys.) [Neuron-glia cell modeling]; D. Brady, R. Willett (Duke EE) [Compressed sensing and imaging]

Funding: ONR, NSF.

Thank you!

www.math.duke.edu/~mauro