

Multiscale analysis on graphs

Mauro Maggioni

Mathematics and Computer Science
Duke University

I.P.A.M. 11/17/08

In collaboration with R.R. Coifman, P.W. Jones, Y-M. Jung, R. Schul,
A.D. Szlam & J.C. Bremer Jr.

Funding: NSF/DHS-FODAVA, DMS, IIS, CCF; ONR.

- Develop tools for studying geometric properties of graphs
- Large scale properties, multiscale properties
- Possible applications:
 - Simplification and compression of graphs
 - Dynamics of graphs at different resolutions
 - Visualization and interaction
- Analysis of functions on graphs
- Possible applications:
 - Classification/regression on graphs
 - Recommendation systems, data organization, search engines, ...

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

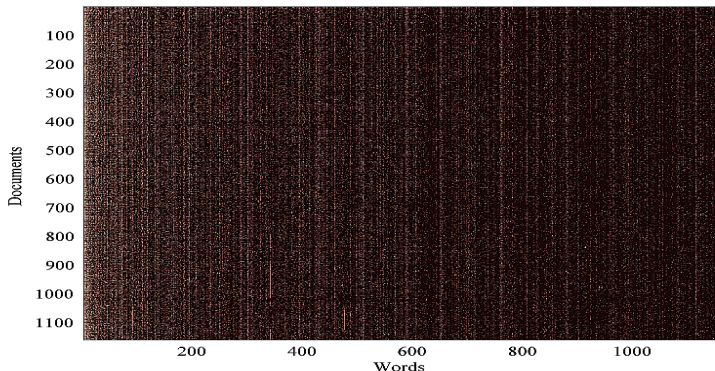
A deluge of data: documents, web searching, customer databases, hyper-spectral imagery (satellite, biomedical, etc...), social networks, gene arrays, proteomics data, neurobiological signals, sensor networks, financial transactions, traffic statistics (automobilistic, computer networks)...

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.

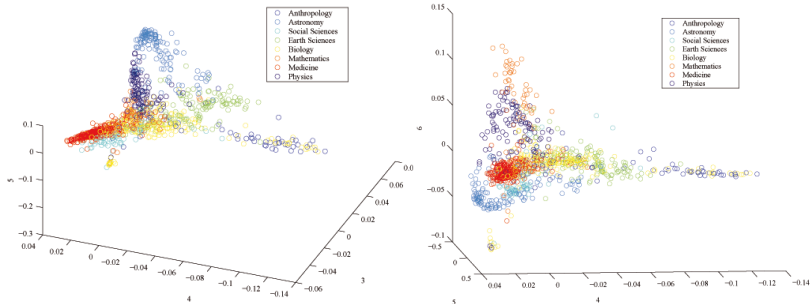
Text documents

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.



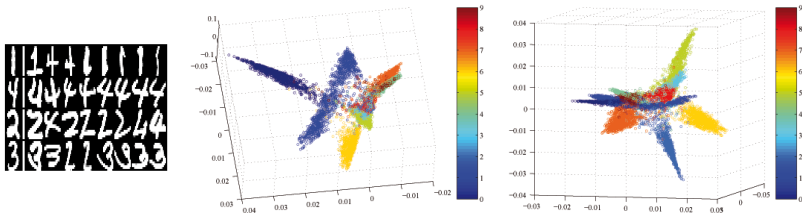
Text documents

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.

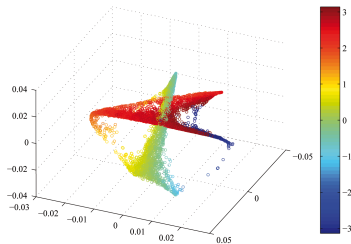
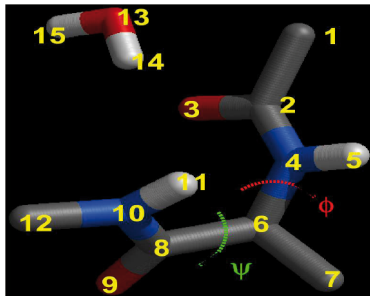


Set of 10,000 pictures (28 by 28 pixels) of 10 handwritten digits. Color represents the label (digit) of each point.

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.

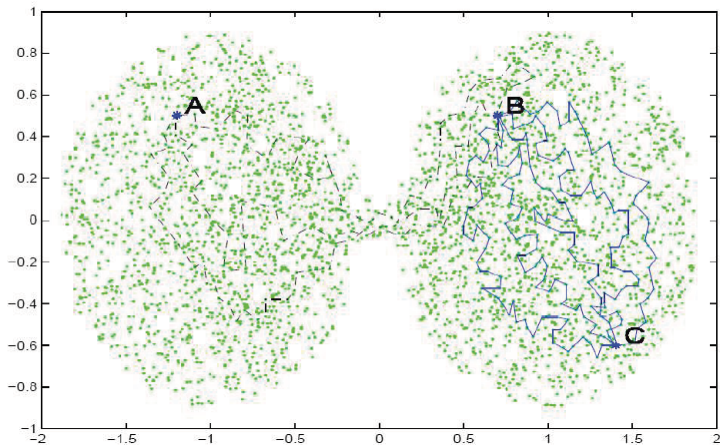
Note 1: W depends on the type of data.

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

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.

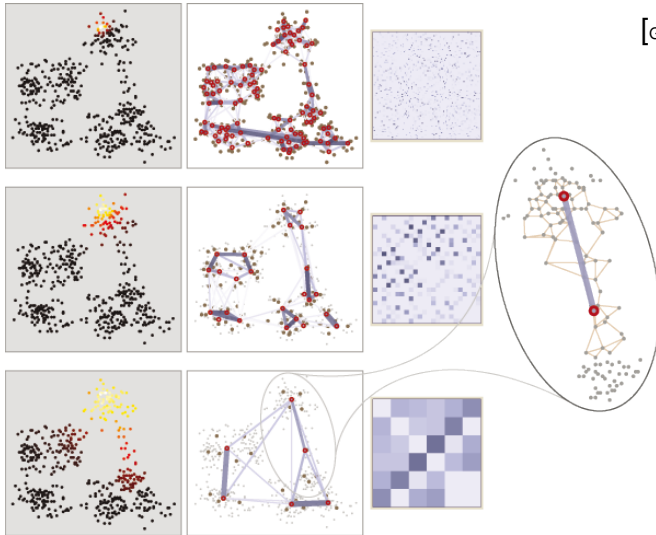
Analysis *on* the set

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).

Multiscale Analysis, a sketch

[Graphics by E. Monson]



Multiscale Analysis - what do we want?

We would like to be able to perform multiscale analysis *of* graphs, and of functions *on* graphs.

Of: produce coarser and coarser graphs, in some sense sketches of the original at different levels of resolution. This could allow a multiscale study of the geometry of graphs.

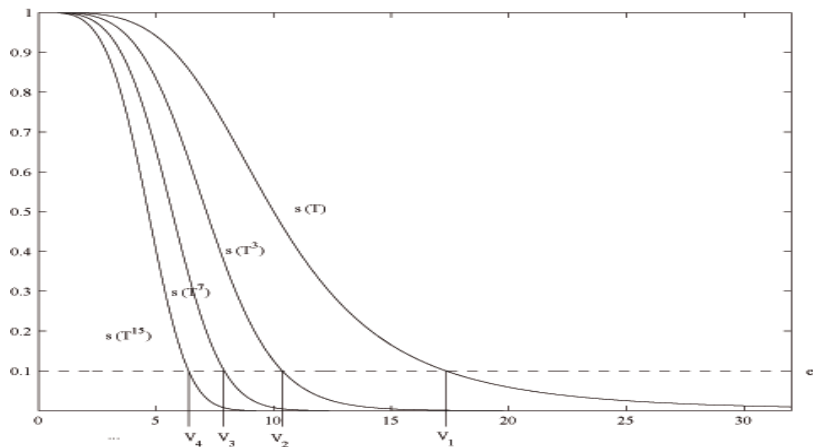
On: produce coarser and coarser functions on graphs, that allow, as wavelets do in low-dimensional Euclidean spaces, to analyse a function at different scales.

We tackle these two questions at once.

Multiscale Analysis, the spectral picture

Let $T = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$ as above be the L^2 -normalized symmetric “random walk”.

The eigenvalues of T and its powers “typically” look like this:



Multiscale Analysis, a bit more precisely

We construct multiscale analyses associated with a diffusion-like process T on a space X , be it a manifold, a graph, or a point cloud. This gives:

- (i) A coarsening of X at different “geometric” scales, in a chain $X \rightarrow X_1 \rightarrow X_2 \rightarrow \dots \rightarrow X_j \dots$;
- (ii) A coarsening (or compression) of the process T at all time scales $t_j = 2^j$, $\{T_j = [T^{2^j}]_{\Phi_j}^{\Phi_j}\}_j$, each acting on the corresponding X_j ;
- (iii) A set of wavelet-like basis functions for analysis of functions (observables) on the manifold/graph/point cloud/set of states of the system.

All the above come *with guarantees*: the coarsened system X_j and coarsened process T_j have random walks “ ϵ -close” to T^{2^j} on X . This comes at the cost of a very careful coarsening: up to $\mathcal{O}(|X|^2)$ operations ($< \mathcal{O}(|X|^3)$!), and only $\mathcal{O}(|X|)$ in certain special classes of problems.

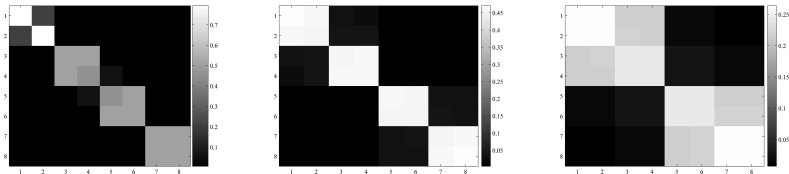
Multiscale Analysis, a trivial example, I

We now consider a simple example of a Markov chain on a graph with 8 states.

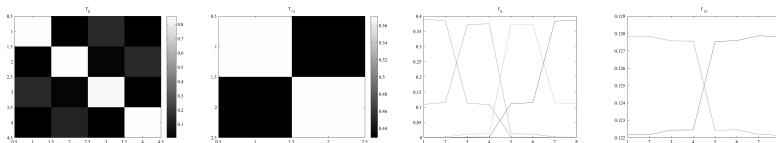
$$T = \begin{pmatrix} 0.80 & 0.20 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.20 & 0.79 & 0.01 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.01 & 0.49 & 0.50 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.50 & 0.499 & 0.001 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.001 & 0.499 & 0.50 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.50 & 0.49 & 0.01 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.01 & 0.49 & 0.50 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.50 & 0.50 \end{pmatrix}$$

From the matrix it is clear that the states are grouped into four pairs $\{\nu_1, \nu_2\}$, $\{\nu_3, \nu_4\}$, $\{\nu_5, \nu_6\}$, and $\{\nu_7, \nu_8\}$, with weak interactions between the the pairs.

Multiscale Analysis, a trivial example, II



Some powers of the Markov chain T , 8×8 , of decreasing effective rank.



Compressed representations $T_6 := T^{2^6}$ (4×4), $T_{13} := T^{2^{13}}$ (2×2), and corresponding soft clusters.

Construction of Diffusion Wavelets

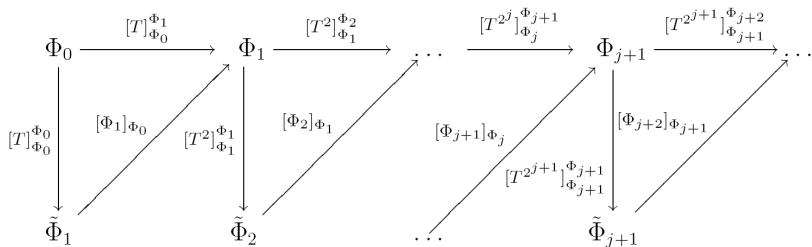


Figure: Diagram for downsampling, orthogonalization and operator compression. (All triangles are ϵ -commutative by construction)

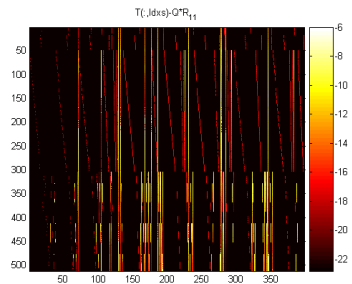
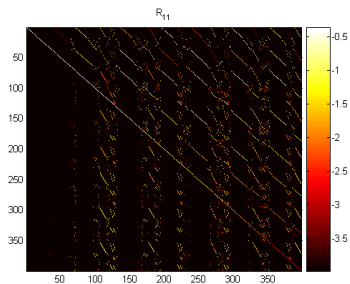
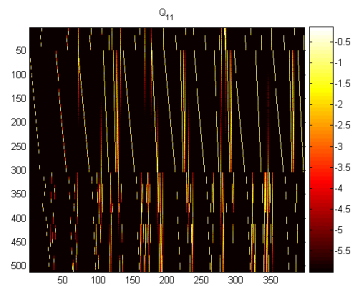
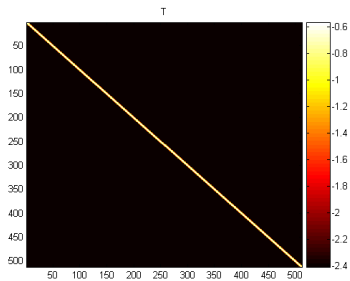
Compression Step

In order to compress the matrix T we use “rank-revealing QR ” decompositions. Fix $\epsilon > 0$.

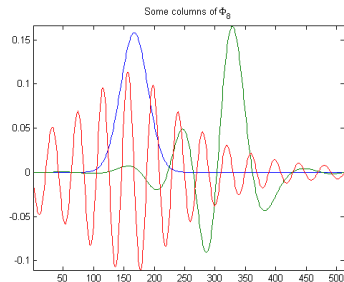
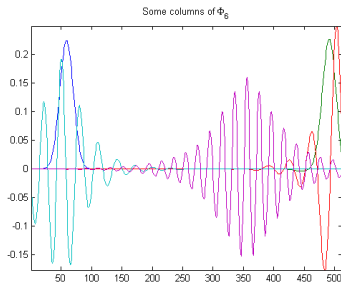
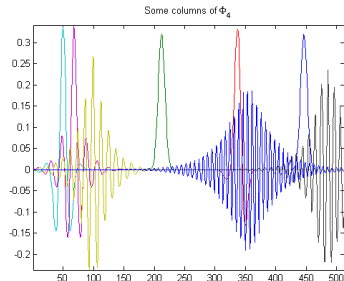
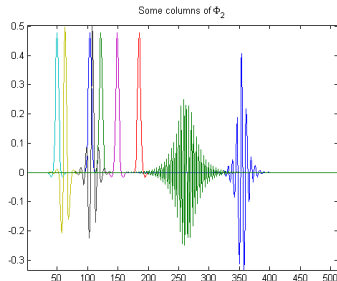
$$T\Pi = QR = \left(Q_{11} \mid Q_{12} \right) \left(\begin{array}{c|c} R_{11} & R_{12} \\ \hline 0 & R_{22} \end{array} \right) \simeq_{\epsilon} Q_{11} \left(R_{11} \mid R_{12} \right)$$

- Q orthogonal, R upper triangular, Π permutation,
 $\|R_{22}\|_2 \simeq \epsilon$
- Q are the scaling functions $[\Phi_1]_{\Phi_0}$, $[R_{11}|R_{12}]$ is $[T]_{\Phi_0}^{\Phi_1}$, the compressed operator from fine to coarse scale.
- The number of columns N_1 of Q_1 (and of R_{11}) determines the dimension of the next coarse scale.
- The first N_1 columns of Π select N_1 representative vertices on the graph.

Example: chain



Example: chain, some scaling functions



$\{\Phi_j\}_{j=0}^J, \{\Psi_j\}_{j=0}^{J-1}, \{[T^{2^j}]_{\Phi_j}^{\Phi_j}\}_{j=1}^J \leftarrow \text{DiffusionWaveletTree}([T]_{\Phi_0}^{\Phi_0}, \Phi_0, J, \text{SpQR}, \epsilon)$

// Input: $[T]_{\Phi_0}^{\Phi_0}$: a diffusion operator, written on the o.n. basis Φ_0

// Φ_0 : an orthonormal basis which ϵ -spans V_0

// J : number of levels to compute

// SpQR : a function compute a sparse QR decomposition, ϵ : precision

// Output: The orthonormal bases of scaling functions, Φ_j , wavelets, Ψ_j , representation of T^{2^j} on Φ_j .

for $j = 0$ **to** $J - 1$ **do**

$[\Phi_{j+1}]_{\Phi_j}, [T]_{\Phi_0}^{\Phi_1} \leftarrow \text{SpQR}([T^{2^j}]_{\Phi_j}^{\Phi_j}, \epsilon)$

$T_{j+1} := [T^{2^{j+1}}]_{\Phi_{j+1}}^{\Phi_{j+1}} \leftarrow [\Phi_{j+1}]_{\Phi_j} [T^{2^j}]_{\Phi_j}^{\Phi_j} [\Phi_{j+1}]_{\Phi_j}^*$

$[\Psi_j]_{\Phi_j} \leftarrow \text{SpQR}(I_{\langle \Phi_j \rangle} - [\Phi_{j+1}]_{\Phi_j} [\Phi_{j+1}]_{\Phi_j}^*, \epsilon)$

end

$Q, R \leftarrow \text{SpQR}(A, \epsilon)$

// Input: A : sparse $n \times n$ matrix ; ϵ : precision

// Output:

// Q, R matrices, possibly sparse, such that $A =_{\epsilon} QR$,

// Q is $n \times m$ and orthogonal,

// R is $m \times n$, and upper triangular up to a permutation,

// the columns of Q ϵ -span the space spanned by the columns of A .

Multiresolution Analysis

Let $V_j = \langle \Phi_j \rangle$, in fact Φ_j (scaling functions) is o.n. basis for V_j .
By construction $L^2(X) = V_0 \supseteq V_1 \supseteq V_2 \supseteq \dots$, and $V_j \rightarrow \langle \varphi_1 \rangle$.
Let W_j be the orthogonal complement of V_{j+1} into V_j . One can construct an o.n. basis Ψ_j (wavelets) for W_j .
 $L^2(X) = W_0 \oplus \dots W_j \oplus V_j$, therefore we have

$$f = \sum_j \sum_{k \in \mathcal{K}_j} \underbrace{\langle f, \psi_{j,k} \rangle}_{\text{wavelet coeff.'s}} \psi_{j,k}.$$

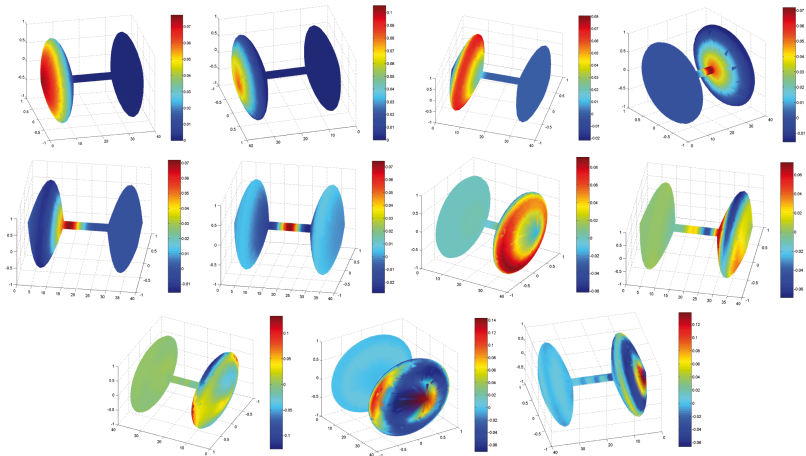
Signal processing tasks by adjusting wavelet coefficients.

Properties of Diffusion Wavelets

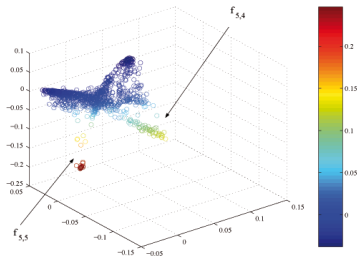
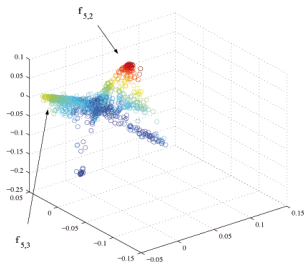
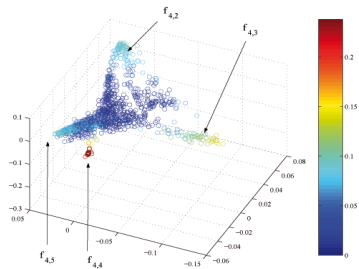
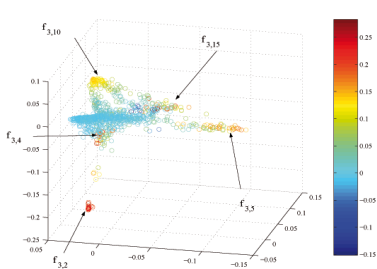
- Multiscale analysis and wavelet transform
- Compact support and estimates on support sizes (currently being improved);
- Vanishing moments (w.r.t. low-frequency eigenfunctions);
- Bounds on the sizes of the approximation spaces (depend on the spectrum of T , which in turn depends on geometry);
- Approximation and stability guarantees of the construction (theory in development).

One can also construct diffusion wavelet packets, and therefore quickly-searchable libraries of waveforms.

Diffusion Wavelets on Dumbbell manifold



Example: Multiscale text document organization

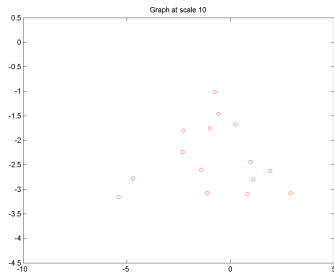
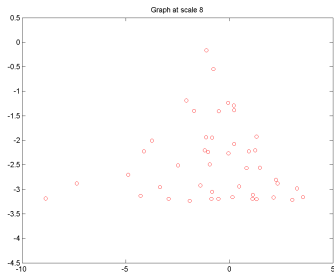
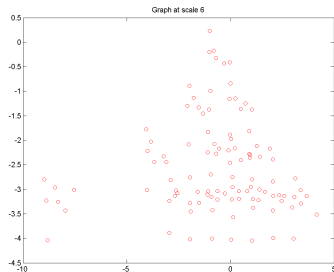
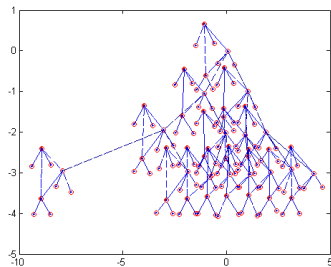


Doc/Word multiscales

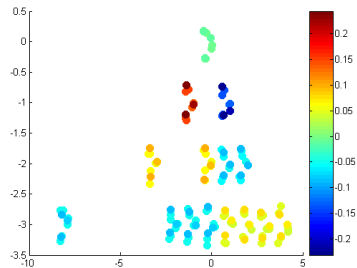
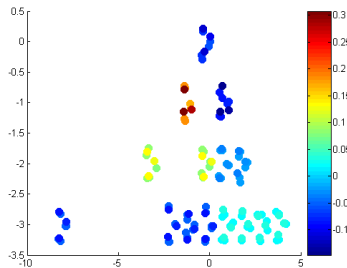
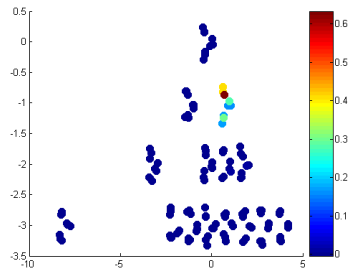
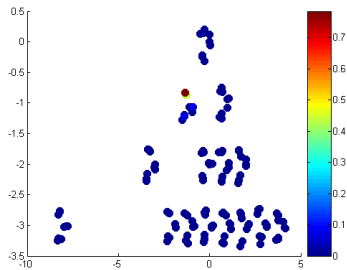
Scaling Fcn	Document Titles	Words
$\varphi_{2,3}$	Acid rain and agricultural pollution Nitrogen's Increasing Impact in agriculture	nitrogen,plant, ecologist,carbon, global
$\varphi_{3,3}$	Racing the Waves Seismologists catch quakes Tsunami! At Lake Tahoe? How a middling quake made a giant tsunami Waves of Death Seabed slide blamed for deadly tsunami Earthquakes: The deadly side of geometry	earthquake,wave, fault,quake, tsunami
$\varphi_{3,5}$	Hunting Prehistoric Hurricanes Extreme weather: Massive hurricanes Clearing the Air About Turbulence New map defines nation's twister risk Southern twisters Oklahoma Tornado Sets Wind Record	tornado,storm, wind,tornadoe, speed

Some example of scaling functions on the documents, with some of the documents in their support, and some of the words most frequent in the documents.

Ongoing work: example of hierarchical graph



Ongoing work: example of hierarchical graph



Local Discriminant Bases

One can in fact build a large dictionary of orthonormal bases (wavelet packets) by further splitting the wavelet subspaces into orthogonal subspaces.

Because of hierarchical organization, one can search such dictionary fast for “best bases” for tasks such as compression, denoising, classification.

LDB (Coifman, Saito) is the best basis for classification.

Local Discriminant Bases

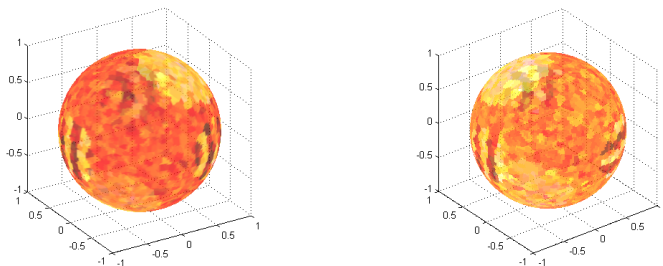


Figure: Left to right, a realization of a function from class 1 and 2 respectively. Note that the third smooth texture patch is on the back side of the sphere, and can be viewed in semitransparency. The other two smooth patches are decoys in random non-overlapping positions.

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.

Semi-supervised Learning on Graphs, cont'd

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. See Chapelle et al. for references

Some open questions and applications

Fourier part:

- Little known about global properties of eigenfunctions
- Behavior of eigenfunctions under perturbations of the graph
- Eigenfunctions on graphs different from sampled manifolds
- Relationships between eigenfunctions of different Laplacians

Multiscale part:

- Geometric multiscale properties of graphs
- Visualization of these multiscale decompositions
- Better constructions? Better sparsity of basis functions and compressed graphs? More efficient algorithms?

Applications

- Multiscale signal processing on graphs
- Multiscale learning and clustering on graphs
- We will see at least a couple of applications to the analysis of networks and network traffic in the next talks!

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]
- R. Brady, E. Monson (Duke CS) [Visualization]

Funding: NSF/DHS-FODAVA, DMS, IIS, CCF; ONR.

Thank you!

www.math.duke.edu/~mauro