CR16: Signal Processing and Networks
Data analysis and processing for networks

Part 1 - Data, signal and image processing on graphs using spectral theory

Pierre Borgnat

DR2 CNRS – Laboratoire de Physique, ENS de Lyon, Université de Lyon

Équipe SISYPHE : Signals, Systems and Physics
et IXXI : Complex System Institute Rhône-Alpes

09/2016

ENS DE LYON

Overview of the lecture

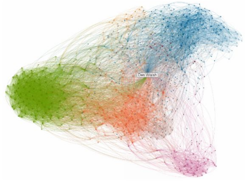
- General objective: revisit classical data analysis techniques (most used in signal and image processing) in the context of discrete structures such as networks and signals defined on graphs.
- The things we will discuss:
  - Introduce you to the emerging field of graph signal & image processing
  - Basic of spectral analysis of graphs, and on the graph Laplacian
  - Harmonic analysis on graphs: wavelets
  - Examples: denoising on graphs; communities;...
- Organization:
  1. Introduction with several examples
  2. Spectral analysis of the Laplacian (on board)
  3. Graph Fourier transform, operators, wavelets
  4. Examples and applications

Introduction: on signals and graphs

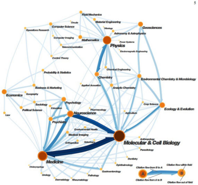
Why data analysis and processing is useful for networks?

- Many examples of data having both labels or values ("signals") and relational properties (graphs)
- Non-trivial estimation issues (e.g., non repeated measures; variables with large distributions (or power-laws); ...)
  $\rightarrow$ **advanced statistical approaches**
- large networks
  $\rightarrow$ **multiscale approaches**
- dynamical networks
  $\rightarrow$ **nonstationary methods**

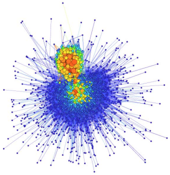
Examples of networks from our digital world



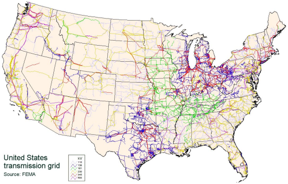
LinkedIn Network



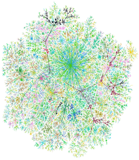
Citation Graph



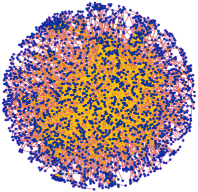
Vehicle Network



USA Power grid

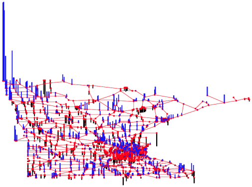


Web Graph

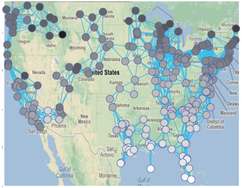


Protein Network

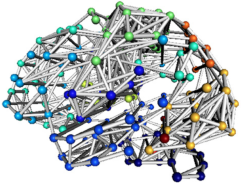
Examples of graph signals



Minnesota Roads



USA Temperature



fcMRI Brain Network

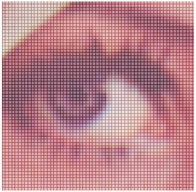


Image Grid

Color Point Cloud

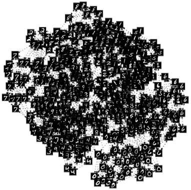
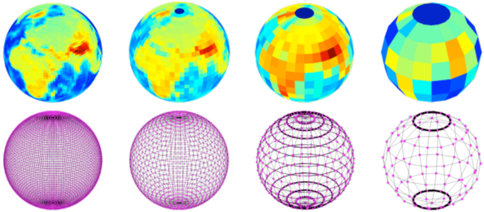


Image Database

Typical problems

[P. Vandergheynst, EPFL, 2013]

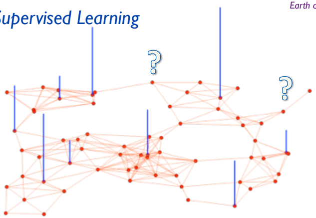
Compression / Visualization

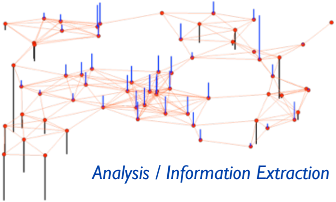


Earth data source: Frederik Simons

Denoising

Semi-Supervised Learning





Analysis / Information Extraction

Fourier transform of signals

"Signal processing 101"

The Fourier transform is of paramount importance:
Given a times series $x_n, \; n = 1, 2, ..., N$, let its Discrete Fourier Transform (DFT) be

$$\forall k \in \mathbb{Z} \quad \hat{x}_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N}$$

Why ?

- Inversion: $x_n = \frac{1}{N} \sum_{k=0}^{N-1} \hat{x}_k e^{-i2\pi kn/N}$
- Best domain to define **Filtering** (operator is diagonal)
- Definition of the **Spectral analysis** (FT of the autocorrelation)
- Alternate representation domains of signals are useful: Fourier domain, DCT, time-frequency representations, wavelets, chirplets,...

Spectral analysis of networks

Spectral theory for network

This is the study of graphs through the **spectral analysis** (eigenvalues, eigenvectors) of matrices **related to the graph**: the adjacency matrix, the Laplacian matrices,....

Notations

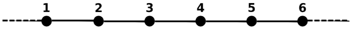
$\mathcal{G} = (V, E, w)$	a weighted graph	
$N = \|V\|$	number of nodes	
A	adjacency matrix	$A_{ij} = w_{ij}$
d	vector of strengths	$d_i = \sum_{j \in V} w_{ij}$
D	matrix of strengths	$D = diag(d)$
f	signal (vector) defined on V	

Relating the Laplacian of graphs to Signal Processing

Laplacian matrix

L or $\mathscr{L}$	laplacian matrix	$L = D - A$
(λ_i)	L's eigenvalues	$0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_{N-1}$
(χ_i)	L's eigenvectors	$L\,\chi_i = \lambda_i\,\chi_i$

A simple example: the straight line



$$\longleftrightarrow \quad L = \begin{pmatrix} \ddots & & & & & & \\ \cdots & -1 & 0 & 0 & 0 & 0 & \\ \cdots & 2 & -1 & 0 & 0 & 0 & \\ & -1 & 2 & -1 & 0 & 0 & \\ & 0 & -1 & 2 & -1 & 0 & \\ & 0 & 0 & -1 & 2 & -1 & \\ & 0 & 0 & 0 & -1 & 2 & \cdots \\ & 0 & 0 & 0 & 0 & -1 & \cdots \\ & & & & & & \ddots \end{pmatrix}$$

For this regular line graph, L is the 1-D classical laplacian operator (i.e. double derivative operator):
its eigenvectors are the Fourier vectors, and its eigenvalues the associated (squared) frequencies

A fundamental analogy

[Shuman et al., *IEEE SP Mag*, 2013]

Objective: Definition of a Fourier Transform adapted to graph signals

f : signal defined on V $\longleftrightarrow$ $\hat{f}$: Fourier transform of f

Fundamental analogy

On *any* graph, the eigenvectors χ_i of the Laplacian matrix L will be considered as the Fourier vectors, and its eigenvalues λ_i the associated (squared) frequencies.

- Works exactly for all regular graphs (+ Beltrami-Laplace)
- Leads to natural generalizations of signal processing

The graph Fourier transform

- $\hat{f}$ is obtained from f's decomposition on the eigenvectors χ_i :

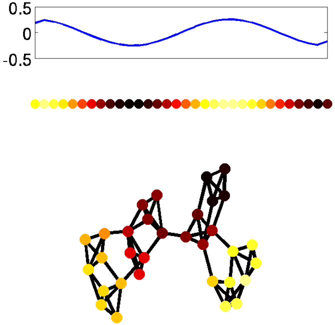
$$\hat{f} = \begin{pmatrix} < \chi_0, f > \\ < \chi_1, f > \\ < \chi_2, f > \\ \dots \\ < \chi_{N-1}, f > \end{pmatrix}$$

Define $\boldsymbol{\chi} = (\chi_0|\chi_1|\dots|\chi_{N-1})$: $\boxed{\hat{f} = \boldsymbol{\chi}^\top f}$

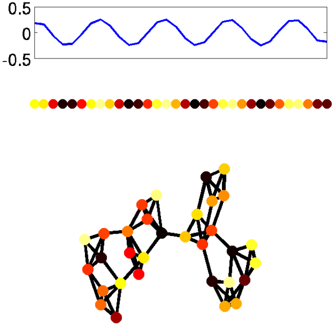
- Reciprocally, the inverse Fourier transform reads: $\boxed{f = \boldsymbol{\chi}\,\hat{f}}$
- The Parseval theorem is valid:
  $\forall (g, h) \quad < g, h > = < \hat{g}, \hat{h} >$

Fourier modes: examples in 1D and in graphs

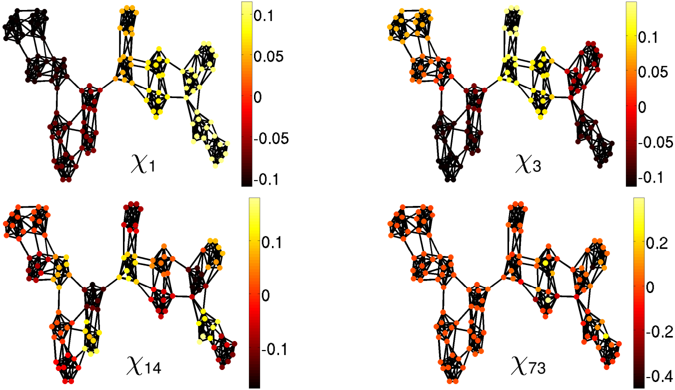
LOW FREQUENCY:



HIGH FREQUENCY:



More Fourier modes



Alternative fundamental spectral correspondance

- With the Normalized Laplacian matrix

$$\mathscr{L} = I - D^{-1/2} A D^{-1/2}$$

  - Related to Ng. et al. normalized spectral clustering
  - Good for degree heterogeneities
  - Related to random walks
  - For community detection
- With the random-walk Laplacian matrix (non symmetrized)

$$L_{rw} = D^{-1} L = I - D^{-1} W$$

  - Better related to random walks
  - Used by Shi-Malik spectral clustering (and graph cuts)
- Using the Adjacency matrix
  - Wigner semi-circular law
  - Discrete Signal Processing in Graphs (good for undirected graphs) [Sandryhaila, Moura, *IEEE TSP*, 2013]

Filtering

Definition of graph filtering

We define a filter function g in the Fourier space.
It is discrete and defined on the eigenvalues $\lambda_i \rightarrow g(\lambda_i)$.

$$\hat{f}^g = \begin{pmatrix} \hat{f}(0)\, g(\lambda_0) \\ \hat{f}(1)\, g(\lambda_1) \\ \hat{f}(2)\, g(\lambda_2) \\ \cdots \\ \hat{f}(N-1)\, g(\lambda_{N-1}) \end{pmatrix} = \hat{\boldsymbol{G}}\,\hat{f} \text{ with } \hat{\boldsymbol{G}} = \begin{pmatrix} g(\lambda_0) & 0 & 0 & \dots & 0 \\ 0 & g(\lambda_1) & 0 & \dots & 0 \\ 0 & 0 & g(\lambda_2) & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & g(\lambda_{N-1}) \end{pmatrix}$$

In the node-space, the filtered signal f^g can be written:

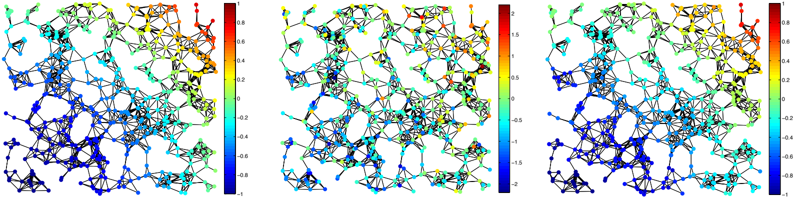
$$f^g = \chi\,\hat{\boldsymbol{G}}\,\chi^\top f$$

Recovery of signals on graphs

[P. Vandergheynst, EPFL, 2013]

- Denoising of a signal with Tikhonov regularization

$$\arg\min_{f} \frac{\tau}{2}||f - y||_2^2 + f^\top L f$$



Original Noisy Denoised

Writing Tikhonov denoising as a Graph filter

- It is easy to solve this regularization problem in the spectral domain

$$\arg\min_f \frac{\tau}{2}||f-y||_2^2 + f^\top Lf \Rightarrow Lf_* + \frac{\tau}{2}(f_* - y) = 0$$

- Move to the spectral domain of the Laplacian

$$\widehat{Lf_*}(i) + \frac{\tau}{2}(\hat{f}_*(i) - \hat{y}(i)) = 0, \quad \forall i \in \{0, 1, ...N-1\}$$

- Solution:

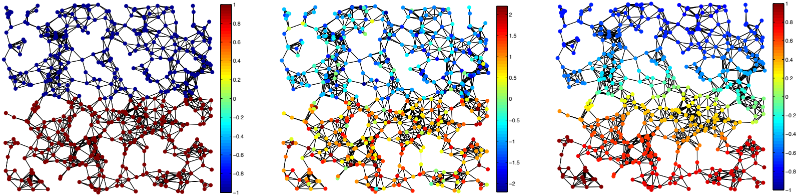
$$\hat{f}_*(i) = \frac{\tau}{\tau + 2\lambda_i}\hat{y}(i)$$

- This is a 1st-order "low pass" filtering (if the λ_i's are considered as frequencies; here, as ω^2)

Recovery of signals on graphs

- Limit of Tikhonov regularization:
  it does not work well for sparse signals

$$\arg\min_f \frac{\tau}{2}||f - y||_2^2 + f^\top L f$$

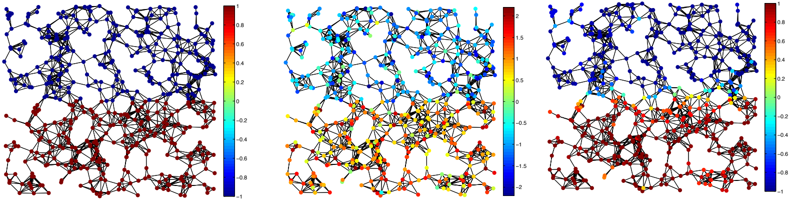


Original Noisy Denoised

Recovery of signals on graphs

- Denoising of a signal with Wavelet regularization

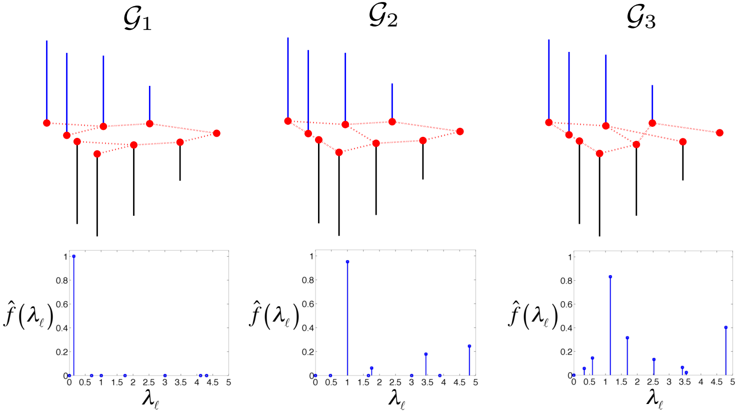
$$\arg\min_{a} ||W^{\top}a - y||_2^2 + \gamma ||a||_1$$



Original Noisy Denoised

- Wavelets will be described later on in the lectures... Stay tuned.

Illustration on the smoothness of graph signals



$\mathbf{f}^{\mathrm{T}}\mathcal{L}_1\mathbf{f} = 0.14$ $\mathbf{f}^{\mathrm{T}}\mathcal{L}_2\mathbf{f} = 1.31$ $\mathbf{f}^{\mathrm{T}}\mathcal{L}_3\mathbf{f} = 1.81$

Generalized translations

[Shuman, Ricaud, Vandergheynst, 2014]

- Classical translation:

$$\left(T_\tau g\right)(t) = g(t-\tau) = \int_{\mathbb{R}} \hat{g}(\xi) e^{-i2\pi\tau\xi} e^{-i2\pi t\xi} d\xi$$

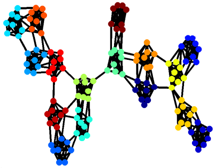
- Graph translations by fundamental analogy:

$$\left(T_n f\right)(a) = \sum_{i=0}^{N-1} \hat{f}(i) \chi_i^*(n) \chi_i(a)$$

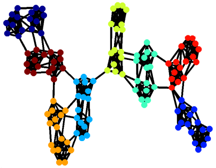
- Example on the Minnesota road networks

Example: graph with multiscale communities

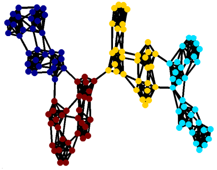
finest scale (16 com.):



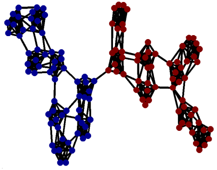
fine scale (8 com.):



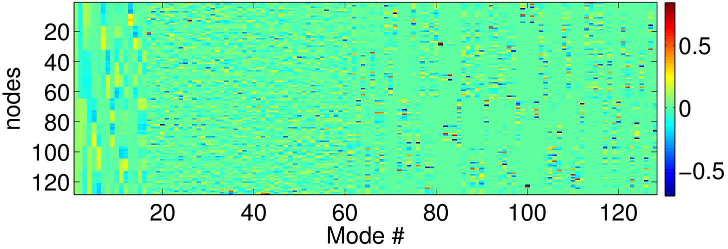
coarser scale (4 com.):

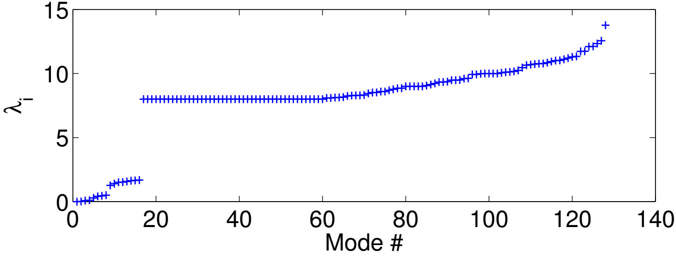


coarsest scale (2 com.):



Spectral analysis: the χ_i and λ_i of this multiscale graph

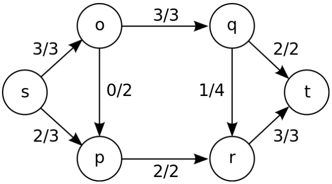




Graph cuts

- Graph cuts in 2 (or bisection): find the partition in two groups of nodes that minimize the cut size (i.e., the number of links cut)
- Exhaustive search can be computationally challenging
- About the problem of cuts:
  An important result is the min-cut max-flow theorem.

  Min-cut pb and Max-flow pb are two primal-dual problems
  The max value of a flow = the min capacity over all cuts
  One possible solution from linear program



Spectral approach for min-cut

- Spectral interpretation: compute the cut as function of the adjacency matrix A
- We have $R = \frac{1}{2} \sum_{i,j \text{ in} \neq \text{groups}} A_{ij}$.

  This is equal to the cut size between the two groups
- Let us note $s_i = \pm 1$ the assignment of node i to group labelled $+1$ or -1
- $R = \frac{1}{2} \sum_{i,j} A_{ij}(1 - s_i s_j) = \frac{1}{4} \sum_{i,j} L_{ij} s_i s_j = \frac{1}{4} \mathbf{s}^\top L \mathbf{s}$
- Hence, the problem reads as:

$$\min_{\mathbf{s}} \; \mathbf{s}^\top L \mathbf{s}$$

Spectral approach for min-cut

- Using the spectral decomposition of the Laplacian:

$$L_{ij} = \sum_{k=1}^{N-1} \lambda_k (\chi_k)_i (\chi_k)_j$$

- The optimal assignment vector (that minimizes R) would be $s_i = (\chi_1)_i \ldots$ if there were no constraints on the s_i's... Note: χ_1 is known as the the Fiedler vector.

- However, $s_i = +1$ or -1...
- Approximated solution: $s_i = \mathrm{sign}((\chi_1)_\mathrm{i})$.
- The estimated groups are still close to χ_1.

Spectral approach for min-RatioCut

- In practice, the cut has to be normalized correctly to find groups of relevant sizes
- One usual metric:
  the Ratio-Cut between sets I and J of nodes

$$R(I,J) = \sum_{i\in I, j\in J} A_{ij}$$

and

$$\text{RatioC}(A,\bar{A}) = \frac{1}{2}\frac{R(A,\bar{A})}{|A|}.$$

- With this normalization, the problem is:

$$\min_{A\subset V} \text{RatioC}(A,\bar{A})$$

Spectral approach for min-RatioCut

- The same problem written in a relaxed form introducing:

$$\mathbf{f}(i) = +\sqrt{\frac{|\bar{A}|}{|A|}} \text{ if } i \in A \text{ and } \mathbf{f}(i) = -\sqrt{\frac{|A|}{|\bar{A}|}} \text{ if } i \in \bar{A}$$

  Then, $||\mathbf{f}|| = \sqrt{|V|}$ and $\mathbf{f}^\top \mathbf{1} = 0$.
- Finally, one has

$$\mathbf{f}^\top L\, \mathbf{f} = |V| \cdot \mathrm{RatioC}(A, \bar{A}).$$

- Hence, problem with relaxed constraints:

$$\begin{array}{l} \min_{\mathbf{f}} \;\; \mathbf{f}^\top L\, \mathbf{f} \\ \text{such that } \mathbf{f}^\top \mathbf{1} = 0, \;\; ||\mathbf{f}||_2 = \sqrt{|\mathrm{V}|} \end{array}$$

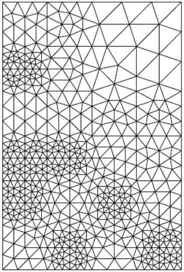
- This method falls down under the category of *Spectral clustering of data* represented by networks

cf. [von Luxburg, *Statistics and Computating*, 2007]

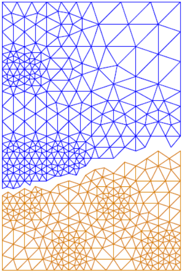
Spectral approach to graph cut

- Example of spectral bisection on an irregular mesh

(a)



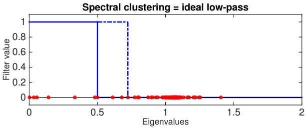
(b)



More generally: spectral clustering

- **Spectral clustering**: unsupervised classification when data are encoded as a graph with similarity matrix $\mathbf{W}$
- In a nutshell: classify according to the eigenvalues of the Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{W}$,
  with eigenvectors χ_k, eigenvalues λ_k and $\mathbf{D}_{aa} = \sum_{b \neq a} \mathbf{W}_{ab}$
- For each data point use the values of the first K χ_k's as feature vectors

$$\mathbf{f}_{K,a} = \mathbf{L}_K \chi^{-1} \delta_a$$

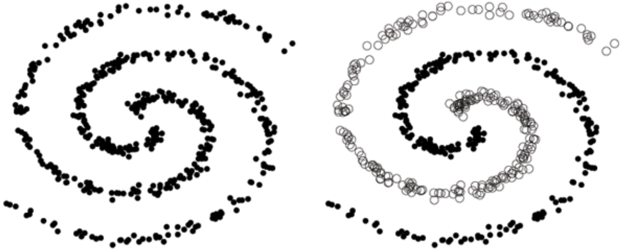


where $\chi = (\chi_1 | \chi_2 | \ldots | \chi_N)$.

- Then use a classification algorithm such as K-means or hierarchical clustering to obtain several groups
- Note: some open issues
  Choice of K? Assessment of the assignments in groups?
- Ex. for choice of K by eigengaps $|\lambda_{k+1} - \lambda_k|$

Spectral clustering

- Example of spectral bisection on data irregularly spread in a space



- It's good, very good in fact for clustering
- However, not really good for natural modules / communities ($\rightarrow$ see lectures on *Complex networks*)

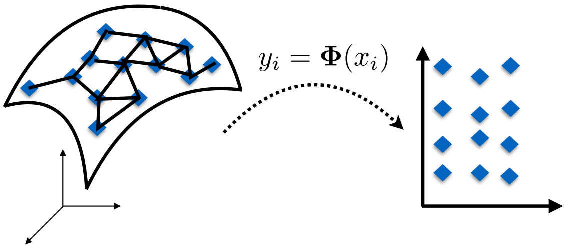
Graph embedding, Laplacian maps

- Spectral clustering := embedding + K-means

$$\forall a \in V : a \rightarrow (\chi_1(a), \chi_2(a), ..., \chi_k(a)) \in \mathbb{R}^k$$

- Objective of embedding: embed vertices in low dimensional space, so as to discover geometry

$$x_i \in \mathbb{R}^d \rightarrow y_i \in \mathbb{R}^k \quad \text{with } k < d$$



Graph embedding, Laplacian maps

- A good embedding preserves locality in the embedding space, so that nearby points are mapped nearby.
  It preserves smoothness.
- For that, minimize the variations of the embedding:

$$\sum_{i,j} A_{ij}(y_i - y_j)^2$$

- Laplacian eigenmaps:

$$\begin{aligned} \text{argmin} \ \ & \mathbf{y}^\top L\,\mathbf{y} \\ \text{such that } & \mathbf{y}^\top A\,\mathbf{y} = 1 \\ \text{and } \ & \mathbf{y}^\top L\,\mathbf{1} = 0 \end{aligned}$$

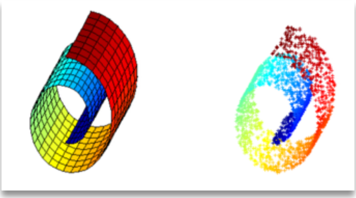
  Alternative formulation:

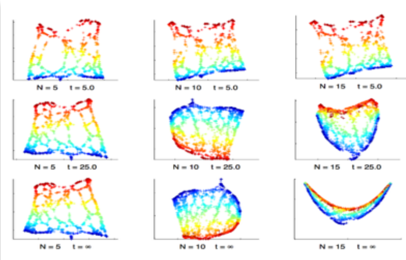
$$L\mathbf{y} = \lambda A\mathbf{y}$$

  (generalized eigenproblem)

Graph embedding, Laplacian maps

- Some examples





[Belkin, Niyogi, 2003]

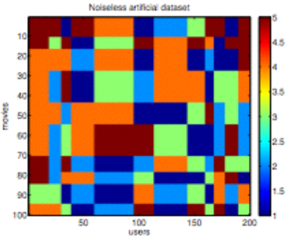
A recommander system

[Vandergheynst et al., EPFL, 2014]

- Assume data in the form M[movie, user] $=$ movie rating

$\mathcal{L}_c$

$\mathcal{L}_r$



- One observes only a subset of M. How to complete it?
- Hypotheses:

  - Users structured as communities,

    and users in community rate similarly

  - Movies are clustered in genres,

    and similar movies are rated similarly by users

A recommander system

[Vandergheynst et al., EPFL, 2014]

- Let us write $A_\Omega(M)$ the observed part of the matrix M
- Matrix completion problem:

$$\min_X \operatorname{rank}(X) \;\; s.t. \;\; A_\Omega(X) = A_\Omega(M).$$

- Problem relaxed with the nuclear norm:
  $||X||_* = Tr((XX^\top)^{1/2}) = \sum_k \sigma_k$
  (where the σ_k are the singular values of $X = U\Sigma V^\top$)
- Tolerance to noise: change $A_\Omega(X) = A_\Omega(M)$ into a penalty term $||A_\Omega \circ (A - M)||$
- Completion by smoothness on the two graphs (users and movies), as quantified by a term

$$\gamma XLX^\top = \gamma \sum_{j,j'} Aij||x_j - x_{j'}||^2.$$

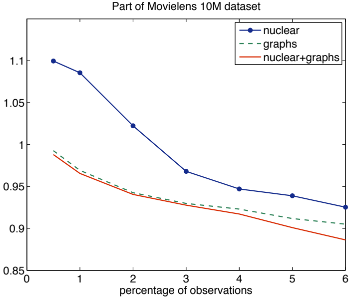
A recommander system

[Vandergheynst et al., EPFL, 2014]

- Hence an optimization problem to solve

$$\min_{X} \gamma_n ||X||_* + ||A_\Omega \circ (A - M)|| + \gamma_r X L_r X^\top + \gamma_c X L_c X^\top$$

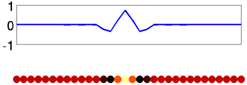
- Solution by advanced optimization tools for convex non-smooth functions.
- Here, the ADMM approach (Alternating Direction Method of Multipliers) works well (see 2nd part of the lectures)



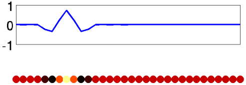
Toward Wavelets on Graphs

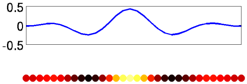
- Fourier is a global analysis. Fourier modes (eigenvectors of the laplacian) are used in classical spectral clustering, but do not enable a jointly local and scale dependent analysis.
- For that classical signal processing (or harmonic analysis) teach us that we need **wavelets**.
- Wavelets : local functions that act as well as a filter around a chosen scale.

  A wavelet:



  – Translated:





  – Scaled

- Classical wavelets $\xrightarrow{\textit{by analogy}}$ Graph wavelets

The classical wavelets

Each wavelet $\psi_{s,a}$ is derived by translating and scaling a mother wavelet ψ:

$$\psi_{s,a}(x) = \frac{1}{s}\psi\left(\frac{x-a}{s}\right)$$

Equivalently, in the Fourier domain:

$$\begin{aligned}
\hat{\psi}_{s,a}(\omega) &= \int_{-\infty}^{\infty} \frac{1}{s}\psi\left(\frac{x-a}{s}\right)\exp^{-i\omega x}\,dx \\
&= \exp^{-i\omega a}\int_{-\infty}^{\infty} \frac{1}{s}\psi\left(\frac{X}{s}\right)\exp^{-i\omega X}\,dX \\
&= \exp^{-i\omega a}\int_{-\infty}^{\infty} \psi\left(X'\right)\exp^{-i\omega X'}\,dX' \\
&= \hat{\delta}_a(\omega)\;\hat{\psi}(s\omega) \qquad \text{where} \quad \delta_a = \delta(x-a)
\end{aligned}$$

One possible definition: $\psi_{s,a}(x) \;=\; \int_{-\infty}^{\infty} \hat{\delta}_a(\omega)\hat{\psi}(s\omega)\exp^{i\omega x}\,d\omega$

The classical wavelets

$$\psi_{s,a}(x) = \int_{-\infty}^{\infty} \hat{\delta}_a(\omega)\hat{\psi}(s\omega) \exp^{i\omega x} d\omega$$

- In this definition, $\hat{\psi}(s\omega)$ acts as a filter bank defined by scaling by a factor s a *filter kernel function* defined in the Fourier space: $\hat{\psi}(\omega)$
- The filter kernel function $\hat{\psi}(\omega)$ is necessarily a *bandpass filter* with:
  - $\hat{\psi}(0) = 0$: the mean of ψ is by definition null
  - $\lim_{\omega \to +\infty} \hat{\psi}(\omega) = 0$: the norm of ψ is by definition finite

(Note: the actual condition is the admissibility property)

Classical wavelets $\xrightarrow{\text{by analogy}}$ Graph wavelets

[Hammond et al. ACHA '11]

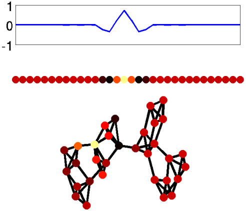
	Classical (continuous) world	Graph world
Real domain	x	node a
Fourier domain	ω	eigenvalues λ_i
Filter kernel	$\hat{\psi}(\omega)$	$g(\lambda_i) \Leftrightarrow \hat{\boldsymbol{G}}$
Filter bank	$\hat{\psi}(s\omega)$	$g(s\lambda_i) \Leftrightarrow \hat{\boldsymbol{G}}_{\boldsymbol{s}}$
Fourier modes	$\exp^{-i\omega x}$	eigenvectors χ_i
Fourier transf. of f	$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(x) \exp^{-i\omega x}\, dx$	$\hat{f} = \boldsymbol{\chi}^\top f$

The wavelet at scale s centered around node a is given by:

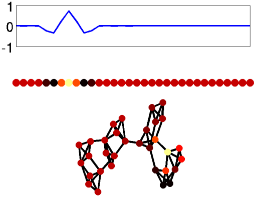
$$\psi_{s,a}(x) = \int_{-\infty}^{\infty} \hat{\delta}_a(\omega)\hat{\psi}(s\omega) \exp^{i\omega x}\, d\omega \longrightarrow \boxed{\psi_{s,a} = \boldsymbol{\chi}\, \hat{\boldsymbol{G}}_{\boldsymbol{s}} \hat{\delta}_a = \boldsymbol{\chi}\, \hat{\boldsymbol{G}}_{\boldsymbol{s}}\, \boldsymbol{\chi}^\top\, \delta_a}$$

Examples of graph wavelets

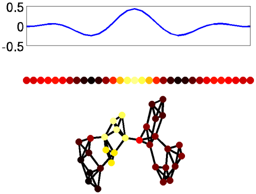
A WAVELET:



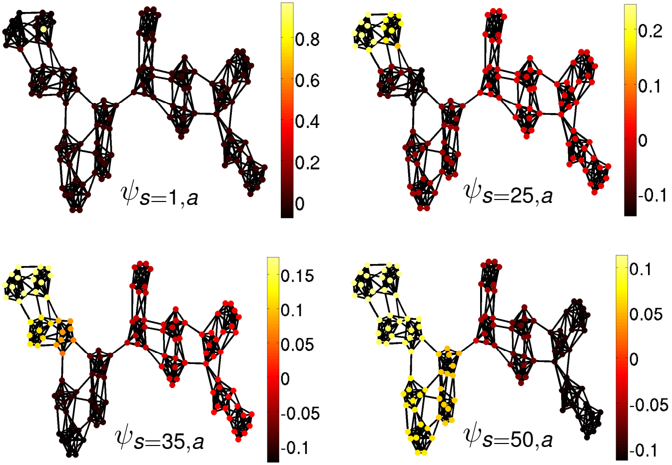
TRANSLATING:



SCALING:

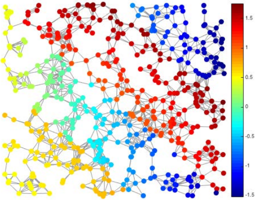


Examples of wavelets: they encode the local topology

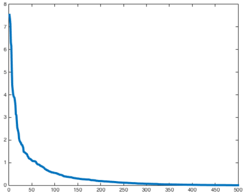


Graph wavelets for data compression

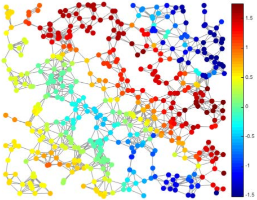
Piecewise-Smooth Signal with Discontinuities



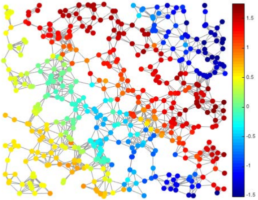
Diffusion Wavelet Coefficients, Sorted by Magnitude



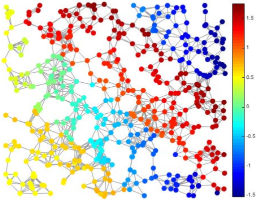
Reconstruction from 10% of Coefficients



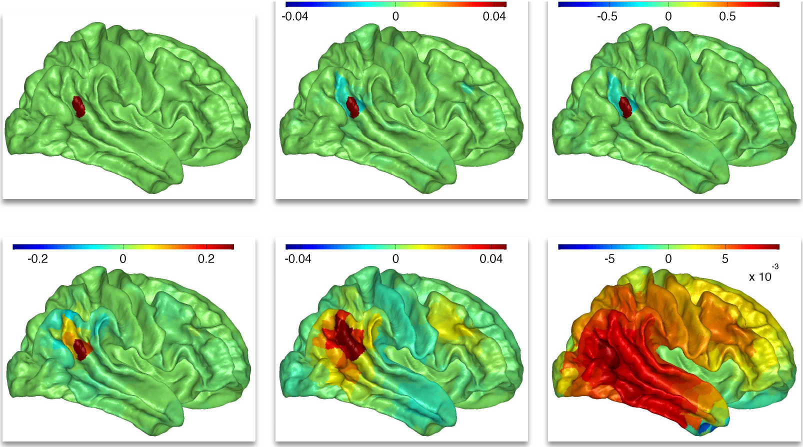
Reconstruction from 20% of Coefficients



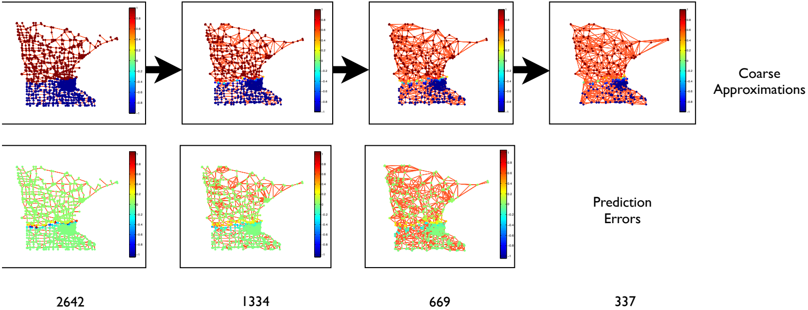
Reconstruction from 50% of Coefficients



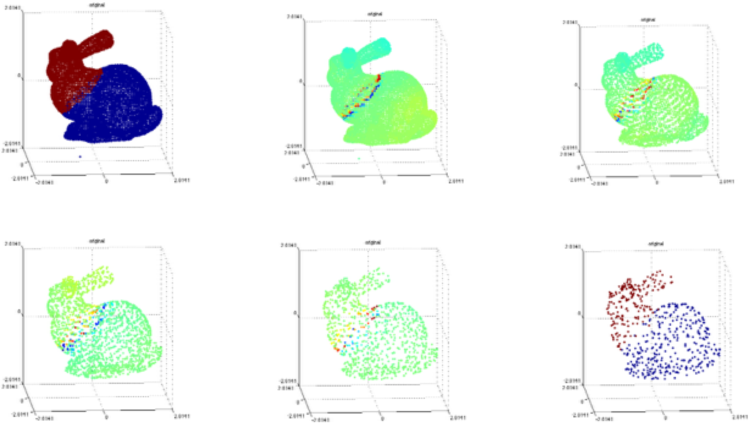
Graph wavelets for brain fMRI data



Graph and Signal graph coarsening



Graph and Signal graph coarsening



The End

- This was an invitation to "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains"
  See [Shuman, Narang, Frossard, Ortega, Vandergheynst, *IEEE SP Mag*, 2013]

http://perso.ens-lyon.fr/pierre.borgnat

Acknowledgements: thanks to Renaud Lambiotte, Pierre Vandergheynst, David Shuman and Nicolas Tremblay for borrowing some (many!) of their figures or slides.