Signal Processing on Graphs

Santiago Segarra, Weiyu Huang, and Alejandro Ribeiro
Dept. of Electrical and Systems Engineering
University of Pennsylvania
aribeiro@seas.upenn.edu
http://www.seas.upenn.edu/users/~aribeiro/

April 15, 2019

Graph Signals

Graph Signals

Graph Laplacian

Graph Fourier Transform (GFT)

Ordering of frequencies

Inverse graph Fourier transform (iGFT)

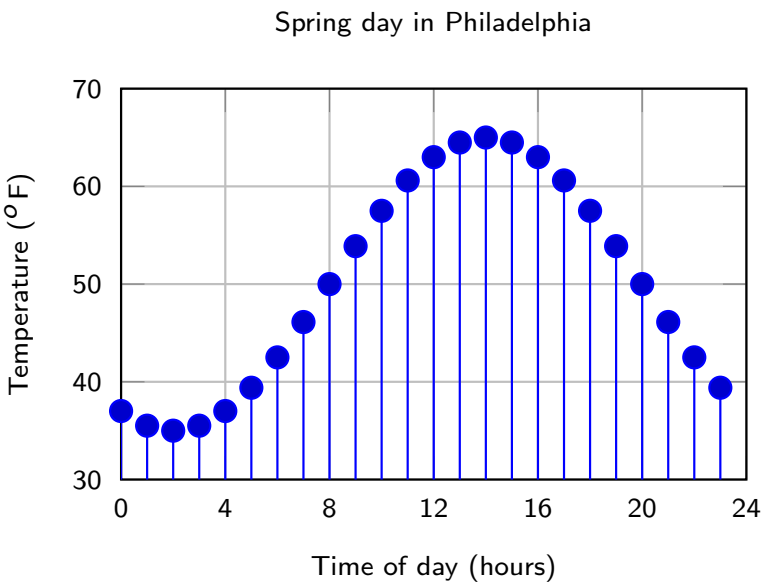
Graph Filters

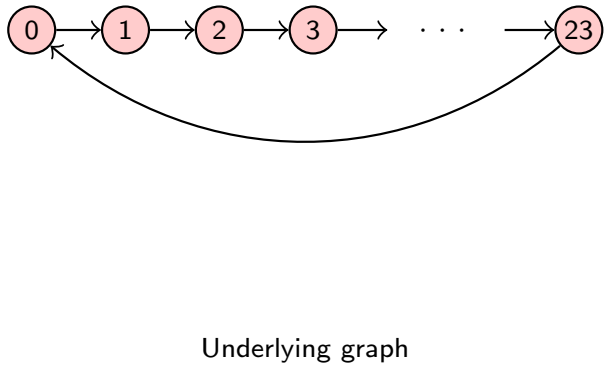
Application: Gene Network

Information sciences at ESE

The support of one dimensional signals

- We have studied one-dimensional signals, image processing, PCA
- It is time to understand them in a more unified way
- Consider the support of one-dimensional signals
- There is an underlying graph structure
  - $\Rightarrow$ Each node represents discrete time instants (e.g. hours in a day)
  - $\Rightarrow$ Edges are unweighted and directed

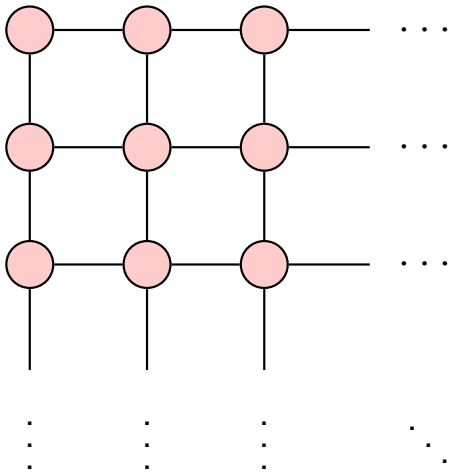




The support of images

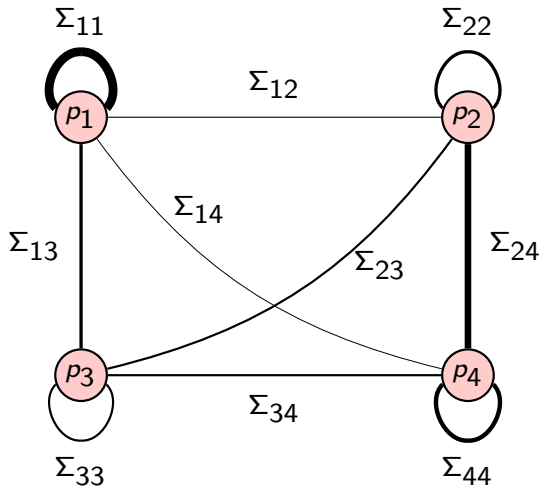
- Similarly, images also have an underlying graph structure
- Each node represents a single pixel
- Edges denote neighborhoods of pixels

  $\Rightarrow$ Unweighted and undirected



PCA uses another underlying graph

- The previous underlying graph assumes a structure between pixels (neighbors in lattice) a priori of seeing the images
- PCA considers images as defined on a different graph
- Each node represents a single pixel
- Edges denote covariance between pairs of pixels in the realizations
  - $\Rightarrow$ A posteriori after seeing the images
  - $\Rightarrow$ Undirected and weighted, including self loops



Graphs

- Formally, a graph (or a network) is a triplet $(\mathcal{V}, \mathcal{E}, W)$
- $\mathcal{V} = \{1, 2, \ldots, N\}$ is a finite set of N nodes or vertices
- $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is a set of edges defined as order pairs (n, m)

  $\Rightarrow$ Write $\mathcal{N}(n) = \{m \in \mathcal{V} : (m, n) \in \mathcal{E}\}$ as the in-neighbors of n
- $W : \mathcal{E} \to \mathbb{R}$ is a map from the set of edges to scalar values, w_{nm}

  $\Rightarrow$ Represents the level of relationship from n to m

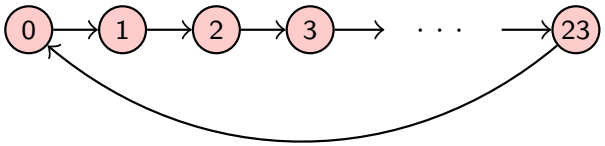
  $\Rightarrow$ Unweighted graphs $\Rightarrow$ $w_{nm} \in \{0, 1\}$, for all $(n, m) \in \mathcal{E}$

  $\Rightarrow$ Undirected graphs $\Rightarrow$ $(n, m) \in \mathcal{E}$ if and only if $(m, n) \in \mathcal{E}$ and $w_{nm} = w_{mn}$, for all $(n, m) \in \mathcal{E}$

  $\Rightarrow$ In-neighbors are neighbors

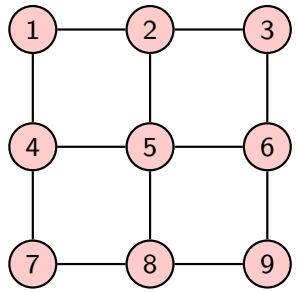
  $\Rightarrow$ More often weights are strictly positive, $W : \mathcal{E} \to \mathbb{R}_{++}$

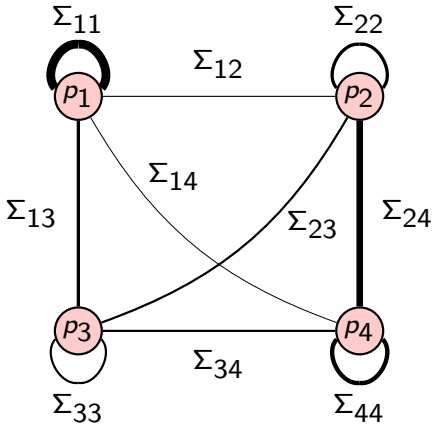
Graphs – examples



- Unweighted and directed graphs
  - $\Rightarrow \mathcal{V} = \{0, 1, \ldots, 23\}$
  - $\Rightarrow \mathcal{E} = \{(0,1), (1,2), \ldots, (22,23), (23,0)\}$
  - $\Rightarrow W : (n,m) \mapsto 1$, for all $(n,m) \in \mathcal{E}$

- Unweighted and undirected graphs
  - $\Rightarrow \mathcal{V} = \{1, 2, 3, \ldots, 9\}$
  - $\Rightarrow \mathcal{E} = \{(1,2), (2,3), \ldots, (8,9), (1,4), \ldots, (6,9)\}$
  - $\Rightarrow W : (n,m) \mapsto 1$, for all $(n,m) \in \mathcal{E}$





- Weighted and undirected graphs
  - $\Rightarrow \mathcal{V} = \{p_1, p_2, p_3, p_4\}$
  - $\Rightarrow \mathcal{E} = \{(p_1,p_1), (p_1,p_2), \ldots, (p_4,p_4)\} = \mathcal{V} \times \mathcal{V}$
  - $\Rightarrow W : (n,m) \mapsto \Sigma_{nm} = \Sigma_{mn}$, for all (n,m)

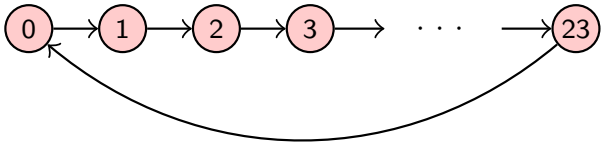
Adjacency matrices

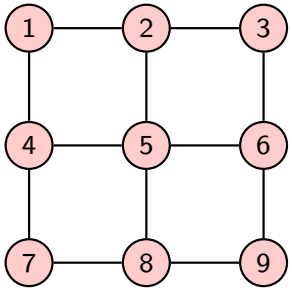
- Given a graph $G = (\mathcal{V}, \mathcal{E}, W)$ of N vertices,
- Its adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ is defined as

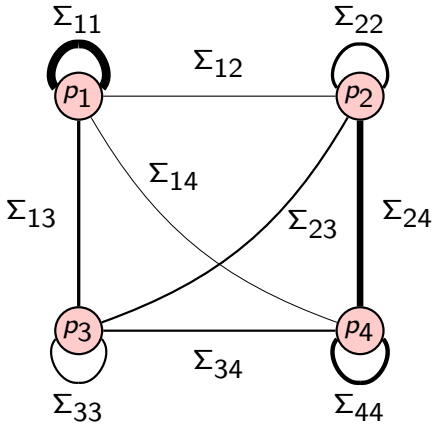
$$A_{nm} = \begin{cases} w_{nm}, & \text{if} (n, m) \in \mathcal{E} \\ 0, & \text{otherwise} \end{cases}$$

- A matrix representation incorporating all information about G
  - $\Rightarrow$ For unweighted graphs, positive entries represent connected pairs
  - $\Rightarrow$ For weighted graphs, also denote proximities between pairs
- Inherently defines an ordering of vertices
  - $\Rightarrow$ same ordering as in graph signals that we will see soon

Adjacency matrices – examples







$$\begin{bmatrix} & 1 & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 \\ 1 & & & & \end{bmatrix}$$

- Different ordering will yield different $\mathbf{A}$

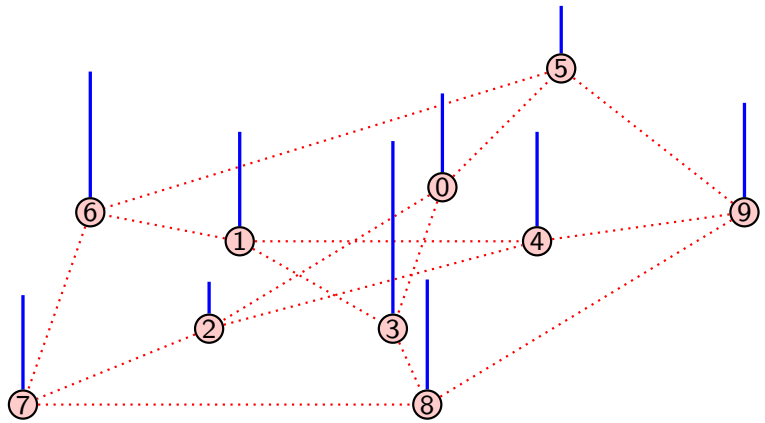
$$\begin{bmatrix} & 1 & & 1 & \\ 1 & & 1 & & \ddots \\ & 1 & & & \\ 1 & & & & \ddots \\ & \ddots & & \ddots & \end{bmatrix}$$

$$\begin{bmatrix} \Sigma_{11} & \Sigma_{12} & \cdots \\ \Sigma_{12} & \Sigma_{22} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} = \mathbf{\Sigma}$$

Graph signals

- Graph signals are mappings $x : \mathcal{V} \to \mathbb{R}$
- Defined on the vertices of the graph
- May be represented as a vector $\mathbf{x} \in \mathbb{R}^N$
- x_n represents the signal value at the nth vertex in $\mathcal{V}$
- Inherently utilizes an ordering of vertices

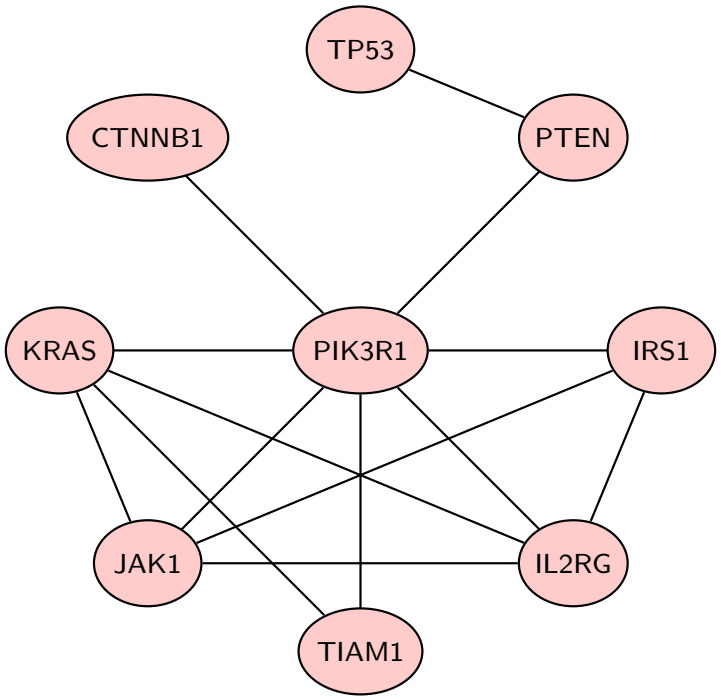
  $\Rightarrow$ same ordering as in adjacency matrices



$$\mathbf{x} = \begin{bmatrix} s_0 \\ s_1 \\ s_3 \\ \vdots \\ s_9 \end{bmatrix} = \begin{bmatrix} 0.6 \\ 0.7 \\ 0.3 \\ \vdots \\ 0.7 \end{bmatrix}$$

Graphs – Gene networks

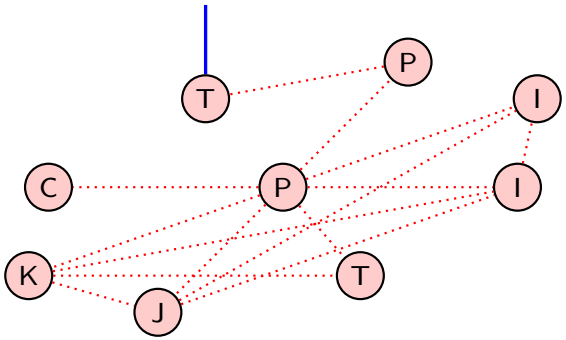
- Graphs representing gene-gene interactions
  - $\Rightarrow$ Each node denotes a single gene (loosely speaking)
  - $\Rightarrow$ Connected if their coded proteins participate in same metobolism



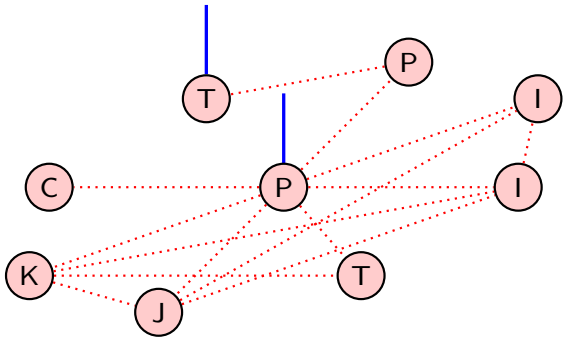
A sample network

Graph signals – Genetic profiles

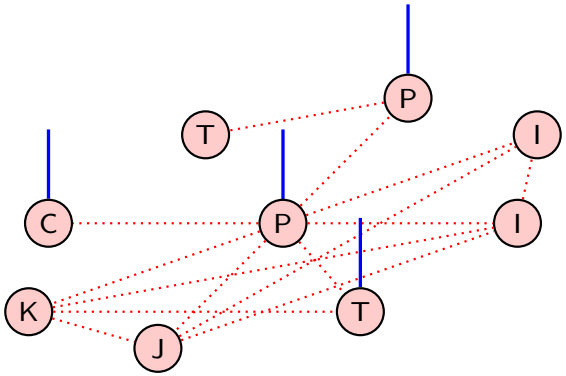
- Genetic profiles for each patient can be considered as a graph signal
  $\Rightarrow$ Signal on each node is 1 if mutated and 0 otherwise



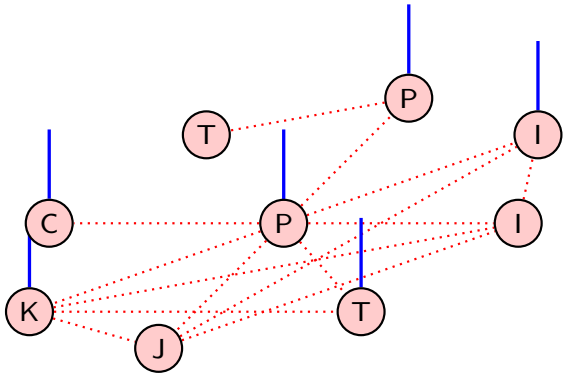
Sample patient 1 with subtype 1



Sample patient 2 with subtype 1



Sample patient 1 with subtype 2



Sample patient 2 with subtype 2

Plans

- We are going to derive following concepts for graph signal processing
  - $\Rightarrow$ Total variations
  - $\Rightarrow$ Frequency
  - $\Rightarrow$ the notion of high or low frequency will be less obvious
  - $\Rightarrow$ DFT and iDFT for graph signals
  - $\Rightarrow$ Graph filtering
- And apply graph signal processing to gene mutation dataset

Graph Laplacian

Graph Signals

Graph Laplacian

Graph Fourier Transform (GFT)

Ordering of frequencies

Inverse graph Fourier transform (iGFT)

Graph Filters

Application: Gene Network

Information sciences at ESE

Degree of a node

- The degree of a node is the sum of the weights of its incident edges
- Given a weighted and undirected graph $G = (\mathcal{V}, \mathcal{E}, W)$
- The degree of node i, $\deg(i)$ is defined as $\deg(i) = \sum_{j \in \mathcal{N}(i)} w_{ij}$
  $\Rightarrow$ where $\mathcal{N}(i)$ is the neighborhood of node i
- Equivalently, in terms of the adjacency matrix $\mathbf{A}$
  $\Rightarrow \deg(i) = \sum_j A_{ij} = \sum_j A_{ji}$
- The degree matrix $\mathbf{D} \in \mathbb{R}^{N \times N}$ is a diagonal matrix s.t. $D_{ii} = deg(i)$
- In directed graphs, each node has an out-degree and an in-degree
  $\Rightarrow$ Weights in outgoing and incoming edges need not coincide

Laplacian of a graph

- Given a graph G with adjacency matrix $\mathbf{A}$ and degree matrix $\mathbf{D}$
- We define the Laplacian matrix $\mathbf{L} \in \mathbb{R}^{N \times N}$ as

$$\mathbf{L} = \mathbf{D} - \mathbf{A}$$

- Equivalently, $\mathbf{L}$ can be defined element-wise as

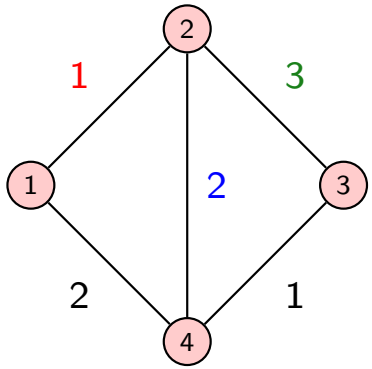
$$L_{ij} = \begin{cases} \deg(i) = \sum_{j \in \mathcal{N}(i)} w_{ij} & \text{if } i = j \\ -w_{ij} & \text{if } j \in \mathcal{N}_i \\ 0 & \text{otherwise} \end{cases}$$

- We assume undirected $G \Rightarrow \deg(i)$ is well-defined
- The normalized Laplacian can be obtained as $\mathcal{L} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$

  $\Rightarrow$ We will mainly focus on the unnormalized version

An example of a graph Laplacian

- Consider the weighted and undirected graph and its Laplacian



$$\mathbf{L} = \begin{bmatrix} 3 & -1 & 0 & -2 \\ -1 & 6 & -3 & -2 \\ 0 & -3 & 4 & -1 \\ -2 & -2 & -1 & 5 \end{bmatrix}$$

- Diagonal elements are strictly positive since no node is isolated

  $\Rightarrow$ Every node has a non-zero degree

- Off-diagonal elements are non-positive

Multiplication by the Laplacian

- Consider a graph G with Laplacian $\mathbf{L}$ and a graph signal $\mathbf{x}$ on G
- Signal $\mathbf{y} = \mathbf{L}\mathbf{x}$ results from multiplying $\mathbf{x}$ with the Laplacian
- Component y_i of $\mathbf{y}$ is (matrix product definition, separate $j = i$)

$$y_i = \sum_j L_{ij} x_j = L_{ii} x_i + \sum_{j \neq i} L_{ij} x_j$$

- We know $L_{ij} = 0$ when $j \notin \mathcal{N}_i$ and $L_{ij} = -w_{ij}$ when $j \in \mathcal{N}_i$. Then

$$y_i = L_{ii} x_i - \sum_{j \in \mathcal{N}_i} w_{ij} x_j$$

- We also know that $L_{ii} = \deg(i) = \sum_{j \in \mathcal{N}_i} w_{ij}$. Therefore

$$y_i = \sum_{j \in \mathcal{N}_i} w_{ij} x_i - \sum_{j \in \mathcal{N}_i} w_{ij} x_j = \sum_{j \in \mathcal{N}_i} w_{ij} (x_i - x_j)$$

- Replaces x_i by weighted average of difference with neighbors

Graph signal diffusion

- Multiplying by the Laplcian yields $\Rightarrow y_i = \sum_{j \in \mathcal{N}_i} w_{ij}(x_i - x_j)$
- y_i measures the difference between $\mathbf{x}$ at a node and its neighborhood
- We say the signal diffuses through the graph. Like heat diffuses
- Temperature at i is averaged with neighbor's temperatures
- Further Laplacian multiplications continue diffusion process
  - $\Rightarrow \mathbf{L}^2\mathbf{x}$ brings in energy from 2-hop neighborhood
  - $\Rightarrow \mathbf{L}^3\mathbf{x}$ brings in energy from 3-hop neighborhood
  - $\Rightarrow \ldots \mathbf{L}^k\mathbf{x}$ brings in energy from k-hop neighborhood

Variability of the Laplacian quadratic form

- The Laplacian quadratic form of graph signal $\mathbf{x}$ is $\Rightarrow \mathbf{x}^T\mathbf{L}\mathbf{x}$
- Quadratic form is a number: row vector $\times$ matrix $\times$ column vector

Theorem

The Laplacian quadratic form of signal $\mathbf{x} = [x_0, x_1, \ldots, x_N]$ *is explicitly given by*

$$\mathbf{x}^T\mathbf{L}\mathbf{x} = \frac{1}{2}\sum_i \sum_{j \in \mathcal{N}_i} w_{ij}(x_i - x_j)^2 = \frac{1}{2}\sum_{(i,j)\in\mathcal{E}} w_{ij}(x_i - x_j)^2$$

- $\mathbf{x}^T\mathbf{L}\mathbf{x}$ quantifies the local variation of signal $\mathbf{x}$
  - $\Rightarrow$ signals can be ordered depending on how much they vary
  - $\Rightarrow$ will be important to order frequencies

Proof of quadratic form variability

Proof.

- We already know the result of the product $\mathbf{y} = \mathbf{L}\mathbf{x}$. Components are:

$$y_i = \sum_{j \in \mathcal{N}_i} w_{ij}(x_i - x_j)$$

- Use explicit form of the product $\mathbf{x}^T\mathbf{y} = \sum_i x_i y_i$ to write

$$\mathbf{x}^T\mathbf{y} = \sum_i x_i y_i = \sum_i \sum_{j \in \mathcal{N}_i} x_i w_{ij}(x_i - x_j)$$

- The term $x_i w_{ij}(x_i - x_j)$ has a symmetric $x_j w_{ji}(x_j - x_i)$. Group them

$$\mathbf{x}^T\mathbf{y} = \frac{1}{2}\sum_i x_i w_{ij}(x_i - x_j) + x_j w_{ji}(x_j - x_i)$$

- But the terms in the sum are now simply

$$x_i w_{ij}(x_i - x_j) + x_j w_{ji}(x_j - x_i) = w_{ij}(x_i - x_j)^2$$

□

Eigenvalues and eigenvectors of the Laplacian

- Denote by λ_i and $\mathbf{v}_i$ the eigenvalues and eigenvectors of $\mathbf{L}$
- Since $\mathbf{x}^T \mathbf{L}\mathbf{x} > 0$ for $\mathbf{x} \neq 0$, $\mathbf{L}$ is positive semi-definite
  $\Rightarrow$ All eigenvalues are nonnegative, i.e. $\lambda_i \geq 0$ for all i
- A constant vector $\mathbf{1}$ is an eigenvector of $\mathbf{L}$ with eigenvalue 0

$$[\mathbf{L}\mathbf{1}]_i = \sum_{j \in \mathcal{N}(i)} w_{ij}(1-1) = 0$$

- Thus, $\lambda_1 = 0$ and $\mathbf{v}_1 = 1/N\ \mathbf{1}$
  $\Rightarrow$ In connected graphs $\lambda_i > 0$ for $i = 2, \ldots, n$
  $\Rightarrow$ Multiplicity of $\lambda = 0$ equals the nr. of connected components

Graph Fourier Transform (GFT)

Graph Signals

Graph Laplacian

Graph Fourier Transform (GFT)

Ordering of frequencies

Inverse graph Fourier transform (iGFT)

Graph Filters

Application: Gene Network

Information sciences at ESE

Graph-shift operator

- Given an arbitrary graph $G = (\mathcal{V}, \mathcal{E}, W)$
- A graph-shift operator $\mathbf{S} \in \mathbb{R}^{N \times N}$ of graph G ia a matrix satisfying
  $\Rightarrow S_{ij} = 0$ for $i \neq j$ and $(i, j) \notin \mathcal{E}$
- $\mathbf{S}$ can take nonzero values in the edges of G or in its diagonal
- We have already seen some possible graph-shift operators
  $\Rightarrow$ Adjacency $\mathbf{A}$, Degree $\mathbf{D}$ and Laplacian $\mathbf{L}$ matrices
- We restrict our attention to normal shifts $\mathbf{S} = \mathbf{V}\Lambda\mathbf{V}^H$
  $\Rightarrow$ Columns of $\mathbf{V} = [\mathbf{v}_1 \mathbf{v}_2 \dots \mathbf{v}_N]$ correspond to the eigenvectors of $\mathbf{S}$
  $\Rightarrow \Lambda$ is a diagonal matrix containing the eigenvalues of $\mathbf{S}$

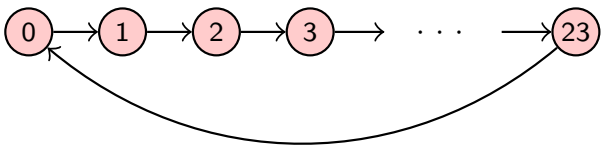
Graph Fourier Transform (GFT)

- Given a graph G and a graph signal $\mathbf{x} \in \mathbb{R}^N$ defined on G
  $\Rightarrow$ Consider a normal graph-shift $\mathbf{S} = \mathbf{V}\Lambda\mathbf{V}^H$
- The Graph Fourier Transform (GFT) of $\mathbf{x}$ is defined as

$$\tilde{\mathbf{x}}(k) = \langle \mathbf{x}, \mathbf{v}_k \rangle = \sum_{n=1}^{N} \mathbf{x}(n)\mathbf{v}_k^*(n)$$

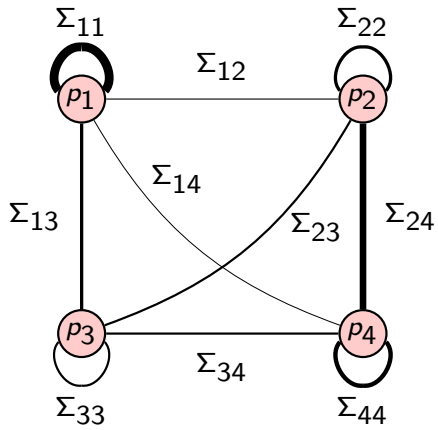
- In matrix form, $\tilde{\mathbf{x}} = \mathbf{V}^H\mathbf{x}$
- Given that the columns of $\mathbf{V}$ are the eigenvectors $\mathbf{v}_i$ of $\mathbf{S}$
  $\Rightarrow$ $\tilde{\mathbf{x}}(k) = \mathbf{v}_k^H\mathbf{x}$ is the inner product between $\mathbf{v}_k$ and $\mathbf{x}$
  $\Rightarrow$ $\tilde{\mathbf{x}}(k)$ is how similar $\mathbf{x}$ is to $\mathbf{v}_k$
  $\Rightarrow$ In particular, GFT $\equiv$ DFT when $\mathbf{V}^H = \mathbf{F}$, i.e. $\mathbf{v}_k = \mathbf{e}_{kN}$

DFT and PCA as particular cases of GFT



- For the directed cycle graph, GFT $\equiv$ DFT
  - $\Rightarrow$ if $\mathbf{S} = \mathbf{A}$ or
  - $\Rightarrow$ if $\mathbf{S} = \mathbf{L}$ for symmetrized graph
  - $\Rightarrow$ then $\mathbf{V}^H = \mathbf{F}$

- For the covariance graph, GFT $\equiv$ PCA
  - $\Rightarrow$ if $\mathbf{S} = \mathbf{A}$, then $\mathbf{V}^H = \mathbf{P}^H$



Ordering of frequencies

Graph Signals

Graph Laplacian

Graph Fourier Transform (GFT)

Ordering of frequencies

Inverse graph Fourier transform (iGFT)

Graph Filters

Application: Gene Network

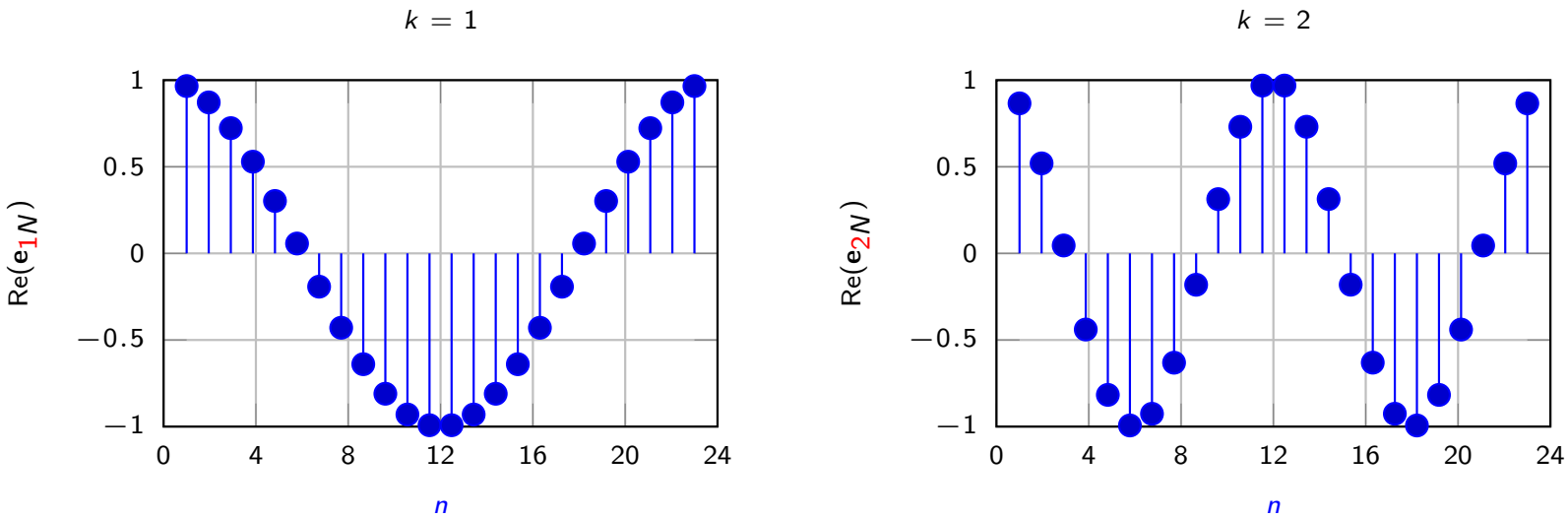
Information sciences at ESE

Ordering of frequencies

- Recall in conventional DFT, the kth DFT component can be written

$$X(k) = \langle \mathbf{x}, \mathbf{e}_{kN} \rangle = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}$$

- We say $X(k)$ the component for higher frequency given higher k
  - $\Rightarrow$ There exists a natural ordering of frequencies
  - $\Rightarrow$ Higher k $\Rightarrow$ higher oscillations



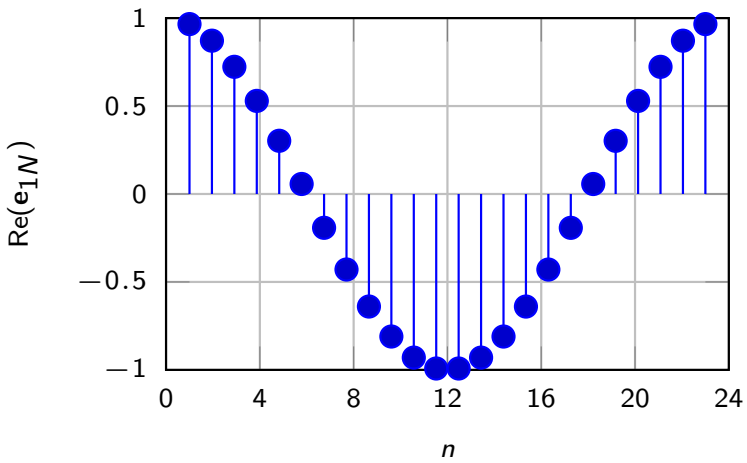
Quantifying oscillations – Zero crossings

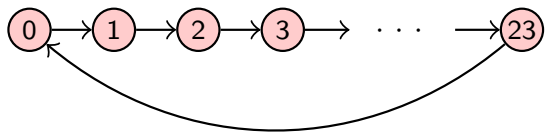
- We want to quantify the qualitative intuition of 'high oscillations'
- Classical zero crossings – # of places signals change signs

$$ZC(\mathbf{x}) = \sum_n \mathbf{1}\{x_n x_{n-1} < 0\}$$

- Graph zero crossings – # of edges signals on two ends differ in signs

$$ZC_G(\mathbf{x}) = \sum_{n=1}^{N} \sum_{m \in \mathcal{N}(n)} \mathbf{1}\{x_n x_m < 0\}$$





Underlying graph

Quantifying oscillations – Total variations

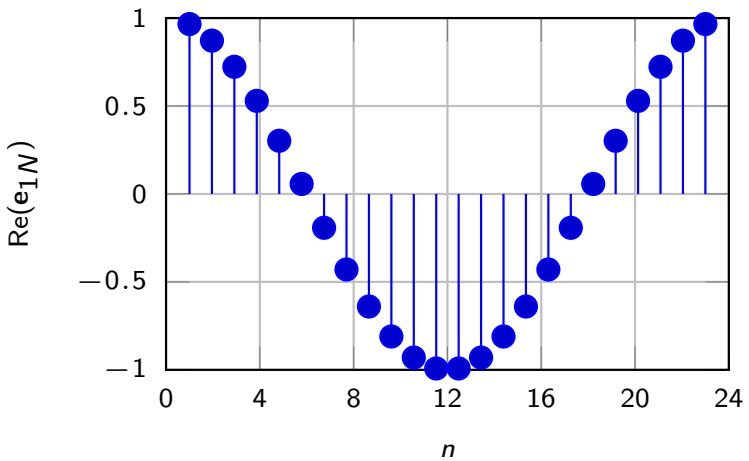
- Classical total variations – sum of squared differences in consecutive signal samples

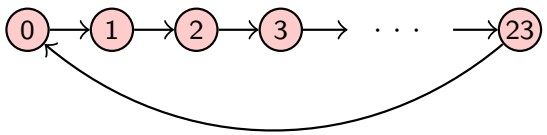
$$TV(\mathbf{x}) = \sum_n (x_n - x_{n-1})^2$$

- Graph total variations – sum of squared differences between signals on two ends of edges multiplied by the corresponding edge weights

  $\Rightarrow$ Also known as Laplacian quadratic form

$$TV_G(\mathbf{x}) = \sum_{n=1}^{N} \sum_{m \in \mathcal{N}(n)} (x_n - x_m)^2 w_{mn} = \mathbf{x}^T L \mathbf{x}$$





Underlying graph

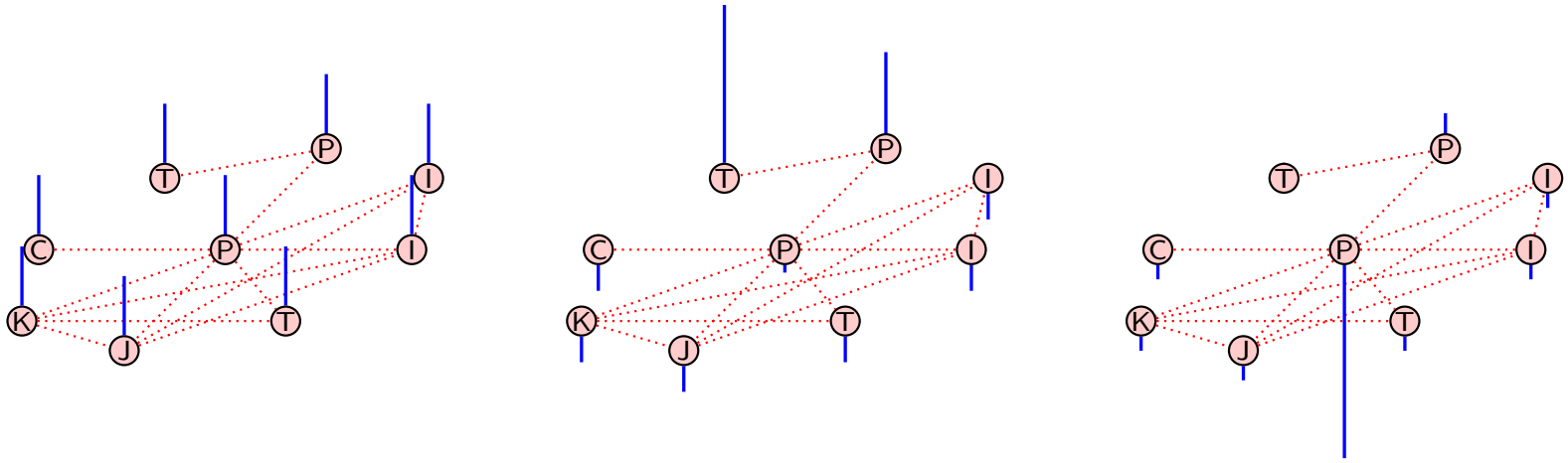
Graph frequencies

- The Laplacian eigenvalues can be interpreted as frequencies
- Larger eigenvalues $\Rightarrow$ Higher frequencies

- The eigenvectors associated with large eigenvalues oscillate rapidly
  $\Rightarrow$ Dissimilar values on vertices connected by edges with high weight
- The eigenvectors associated with small eigenvalues vary slowly
  $\Rightarrow$ Similar values on vertices connected by edges with high weight
- Eigenvector associated with eigenvalue 0 is constant
  $\Rightarrow$ for connected graph

Graph frequencies – Gene networks

- Three graph Laplacian eigenvectors for the gene networks



$\mathbf{v}_0$ $\mathbf{v}_1$ $\mathbf{v}_9$

- $ZC_G(\mathbf{v}_0) = 0$
- $TV_G(\mathbf{v}_0) = 0$
- $ZC_G(\mathbf{v}_1) = 2$
- $TV_G(\mathbf{v}_1) = 0.4$
- $ZC_G(\mathbf{v}_1) = 20$
- $TV_G(\mathbf{v}_1) = 8.0$

Inverse graph Fourier transform (iGFT)

Graph Signals

Graph Laplacian

Graph Fourier Transform (GFT)

Ordering of frequencies

Inverse graph Fourier transform (iGFT)

Graph Filters

Application: Gene Network

Information sciences at ESE

Inverse graph Fourier transform

- Recall the graph Fourier transform $\mathbf{x}$
  - $\Rightarrow$ of any signal $\mathbf{x} \in \mathbb{R}^N$ on the vertices of graph G
  - $\Rightarrow$ is the expansion of $\mathbf{x}$ of the eigenvectors of the Laplacian

$$\tilde{\mathbf{x}}(k) = \langle \mathbf{x}, \mathbf{v}_k \rangle = \sum_{n=1}^{N} x(n) v_k^*(n)$$

- In matrix form, $\tilde{\mathbf{x}} = \mathbf{V}^H \mathbf{x}$
- The inverse graph Fourier transform is

$$\mathbf{x}(n) = \sum_{k=0}^{N-1} \tilde{\mathbf{x}}(k) v_k(n)$$

- In matrix form, $\mathbf{x} = \mathbf{V}\tilde{\mathbf{x}}$

Inverse theorem, like a pro

- Recap in proving theorems we have monkey steps and one smart step

  $\Rightarrow$ That was orthonormality $\Rightarrow$ $\mathbf{V}^H$ is Hermitian $\Rightarrow$ $\mathbf{V}\mathbf{V}^H = \mathbf{I}$

Theorem

The inverse graph Fourier transform (iGFT) is, indeed, the inverse of the GFT.

Proof.

- Write $\mathbf{x} = \mathbf{V}\tilde{\mathbf{x}}$ and $\tilde{\mathbf{x}} = \mathbf{V}^H\mathbf{x}$ and exploit fact that $\mathbf{V}$ is Hermitian

$$\mathbf{x} = \mathbf{V}\tilde{\mathbf{x}} = \mathbf{V}\mathbf{V}^H\mathbf{x} = \mathbf{I}\mathbf{x} = \mathbf{x}$$

□

- This is the last inverse theorem we will see...

Energy conservation (Parseval) theorem, like a pro

Theorem

The GFT preserves energy $\Rightarrow \|\mathbf{x}\|^2 = \mathbf{x}^H\mathbf{x} = \tilde{\mathbf{x}}^H\tilde{\mathbf{x}} = \|\tilde{\mathbf{x}}\|^2$

Proof.

- Use GFT to write $\tilde{\mathbf{x}} = \mathbf{V}^H\mathbf{x}$ and the fact that $\mathbf{V}$ is Hermitian

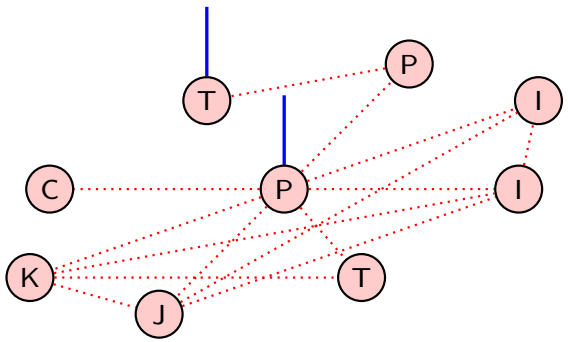
$$\|\tilde{\mathbf{x}}\|^2 = \tilde{\mathbf{x}}^H\tilde{\mathbf{x}} = \left(\mathbf{V}^H\mathbf{x}\right)^H \mathbf{V}^H\mathbf{x} = \mathbf{x}^H\mathbf{V}\mathbf{V}^H\mathbf{x} = \mathbf{x}^H\mathbf{x} = \|\mathbf{x}\|^2 \qquad \square$$

- This is the last energy conservation theorem we will see...

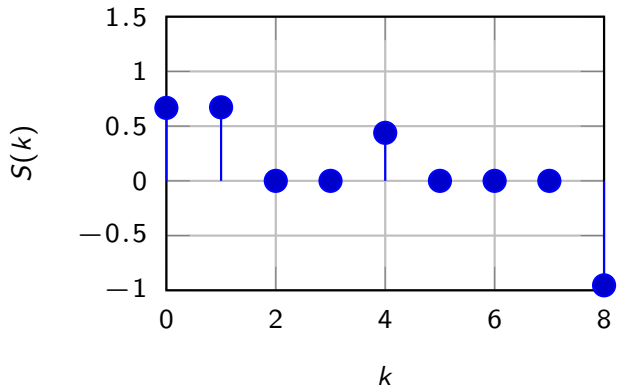
Graph signal representations in two domains

- Graph signals can be equivalently represented in two domains
  - $\Rightarrow$ The vertex domain and the graph spectral domain

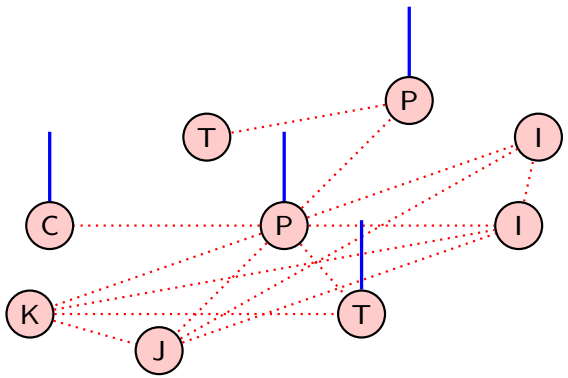
Sample patient 2 with subtype 1



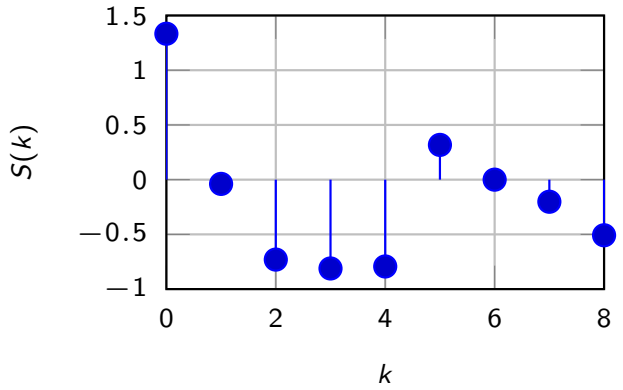
Spectral representation for patient 2 with subtype 1



Sample patient 1 with subtype 2



Spectral representation for patient 1 with subtype 2



Graph Filters

Graph Signals

Graph Laplacian

Graph Fourier Transform (GFT)

Ordering of frequencies

Inverse graph Fourier transform (iGFT)

Graph Filters

Application: Gene Network

Information sciences at ESE

Linearity and shift-invariance

- A graph filter $f : \mathbb{R}^N \to \mathbb{R}^N$ is a map between graph signals
  $\Rightarrow$ Given a graph signal $\mathbf{x} \in \mathbb{R}^N$, its filtered version is $\mathbf{y} = f(\mathbf{x})$
- We will focus on filters f that are linear and shift-invariant
- A linear filter f is one that satisfies

$$\mathbf{y}_1 = f(\mathbf{x}_1), \quad \mathbf{y}_2 = f(\mathbf{x}_2) \quad \Longrightarrow \quad \alpha_1 \mathbf{y}_1 + \alpha_2 \mathbf{y}_2 = f(\alpha_1 \mathbf{x}_1 + \alpha_2 \mathbf{x}_2)$$

- A shift-invariant filter f satisfies

$$f(\mathbf{S}\mathbf{x}) = \mathbf{S} f(\mathbf{x})$$

  where $\mathbf{S}$ is the graph-shift operator of the graph where $\mathbf{x}$ is defined
- Shift-invariance is the graph analog of time invariance in classical SP

Graph filters as matrix polynomials

- Given a graph G and a graph-shift operator $\mathbf{S} \in \mathbb{R}^{N \times N}$ on G
- We define the graph filter $\mathbf{H}$ as

$$\mathbf{H} := h_0 \mathbf{S}^0 + h_1 \mathbf{S}^1 + h_2 \mathbf{S}^2 + \ldots = \sum_{\ell=0}^{L} h_\ell \mathbf{S}^\ell$$

- $\mathbf{H}$ is a polynomial on the graph-shift operator $\mathbf{S}$ with coefficients h_i
  $\Rightarrow$ L is the degree of the filter
- Filter $\mathbf{H}$ acts on a graph signal $\mathbf{x} \in \mathbb{R}^N$ to generate $\mathbf{y} = \mathbf{H}\mathbf{x}$
  $\Rightarrow$ If we define $\mathbf{x}^{(\ell)} := \mathbf{S}^\ell \mathbf{x} = \mathbf{S}\mathbf{x}^{(\ell-1)}$

$$\mathbf{y} = \sum_{\ell=0}^{L} h_\ell \mathbf{x}^{(\ell)}$$

- Why is $\mathbf{H}$ defined as a polynomial on $\mathbf{S}$?

Matrix polynomials are linear and shift-invariant

Proposition

The graph filter $\mathbf{H} = \sum_{\ell=0}^{L} h_\ell \mathbf{S}^\ell$ is linear and shift-invariant.

Proof.

- Since $\mathbf{H}$ is a matrix, linearity is trivial

$$\mathbf{y}_1 = \mathbf{H}\mathbf{x}_1, \quad \mathbf{y}_2 = \mathbf{H}\mathbf{x}_2 \quad \Longrightarrow \quad \alpha_1\mathbf{y}_1 + \alpha_2\mathbf{y}_2 = \mathbf{H}(\alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2)$$

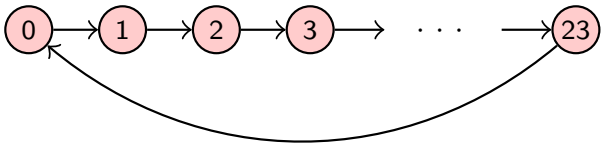
- For shift-invariance, note that $\mathbf{S}$ commutes with $\mathbf{S}^i$ for all i

$$\mathbf{H}(\mathbf{S}\mathbf{x}) = \left(\sum_{\ell=0}^{L} h_\ell \mathbf{S}^\ell\right)\mathbf{S}\mathbf{x} = \mathbf{S}\left(\sum_{\ell=0}^{L} h_\ell \mathbf{S}^\ell\right)\mathbf{x} = \mathbf{S}(\mathbf{H}\mathbf{x})$$

In fact, no other formulation of $\mathbf{H}$ is linear and shift-invariant

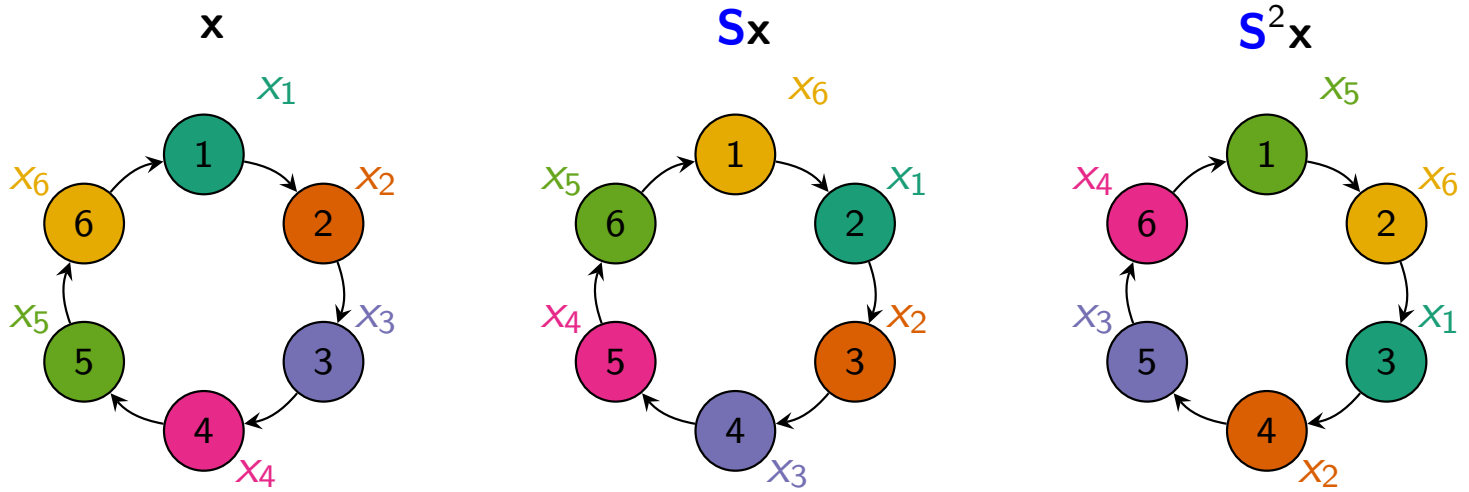
$\Rightarrow$ We will not show this

Connection with filters of time-varying signals



- Consider the particular case where $\mathbf{S} = \mathbf{A}_{dc}$
  $\Rightarrow$ Adjacency matrix of a directed cycle

- Focus on a signal $\mathbf{x}$ defined on a cyclic graph with 6 nodes



- Consider the output signal $\mathbf{y} = \mathbf{H}\mathbf{x}$

$$\mathbf{y} = h_0\mathbf{x} + h_1\mathbf{S}^1\mathbf{x} + h_2\mathbf{S}^2\mathbf{x} + h_3\mathbf{S}^3\mathbf{x} + h_4\mathbf{S}^4\mathbf{x} + h_5\mathbf{S}^5\mathbf{x}$$

Connection with filters of time-varying signals

- Let's focus on the first component of signal $\mathbf{y}$

$$y_1 = h_0[\mathbf{S}^0\mathbf{x}]_1 + h_1[\mathbf{S}^1\mathbf{x}]_1 + h_2[\mathbf{S}^2\mathbf{x}]_1 + h_3[\mathbf{S}^3\mathbf{x}]_1 + h_4[\mathbf{S}^4\mathbf{x}]_1 + h_5[\mathbf{S}^5\mathbf{x}]_1$$
$$= h_0 x_1 + h_1 x_6 + h_2 x_5 + h_3 x_4 + h_4 x_3 + h_5 x_2$$

- In general, for element y_n of $\mathbf{y}$, exploiting the fact that $\mathbf{x}$ is cyclic

$$y_n = \sum_{l=0}^{N-1} h_l x_{n-l}$$

- Defining $\mathbf{h} := [h_0, h_1, \ldots, h_5]^T$ we may write

$$\mathbf{y} = \mathbf{h} * \mathbf{x}$$

- Thus, for the particular case where $\mathbf{S} = \mathbf{A}_{dc}$

  $\Rightarrow$ $\mathbf{h}$ recovers the impulse response of the filter

Frequency response of a graph filter

- Recalling that $\mathbf{S} = \mathbf{V}\Lambda\mathbf{V}^H$, we may write

$$\mathbf{H} = \sum_{\ell=0}^{L} h_\ell \mathbf{S}^\ell = \mathbf{V}\left(\sum_{\ell=0}^{L} h_\ell \Lambda^\ell\right)\mathbf{V}^H$$

- The application $\mathbf{Hx}$ of filter $\mathbf{H}$ to $\mathbf{x}$ can be split into three parts
  - $\Rightarrow \mathbf{V}^H$ takes signal $\mathbf{x}$ to the graph frequency domain $\tilde{\mathbf{x}}$
  - $\Rightarrow \widehat{\mathbf{H}} := \sum_{\ell=0}^{L} h_\ell \Lambda^\ell$ modifies the frequency coefficients to obtain $\tilde{\mathbf{y}}$
  - $\Rightarrow \mathbf{V}$ brings the signal $\tilde{\mathbf{y}}$ back to the graph domain $\mathbf{y}$
- Since $\widehat{\mathbf{H}}$ is diagonal, define $\widehat{\mathbf{H}} =: diag(\widehat{\mathbf{h}})$
  - $\Rightarrow \widehat{\mathbf{h}}$ is the frequency response of the filter $\mathbf{H}$
  - $\Rightarrow$ Output at frequency i depends only on input at frequency i

$$\tilde{y}_i = \widehat{h}_i \tilde{x}_i$$

Frequency response and filter coefficients

- In order to design a graph with a particular frequency response $\widehat{\mathbf{h}}$
  $\Rightarrow$ Need to know the relation between $\widehat{\mathbf{h}}$ and the filter coefficients $\mathbf{h}$
- Define the matrix $\boldsymbol{\Psi} := \begin{pmatrix} 1 & \lambda_1 & \dots & \lambda_1^{L-1} \\ \vdots & \vdots & & \vdots \\ 1 & \lambda_N & \dots & \lambda_N^{L-1} \end{pmatrix}$

Proposition

The frequency response $\widehat{\mathbf{h}}$ of a graph filter with coefficients $\mathbf{h}$ is given by

$$\widehat{\mathbf{h}} = \boldsymbol{\Psi}\mathbf{h}$$

Proof.

- Since $\widehat{\mathbf{h}} := \text{diag}(\sum_{\ell=0}^{L} h_\ell \boldsymbol{\Lambda}^\ell)$ we have that $\widehat{h}_i = \sum_{\ell=0}^{L} h_\ell \lambda_i^\ell$
- Defining $\boldsymbol{\lambda}_i = [\lambda_i^0, \lambda_i^1, \dots, \lambda_i^{L-1}]^T$ we have that $\widehat{h}_i = \boldsymbol{\lambda}_i^T \mathbf{h}$
- Stacking the values for all $\widehat{h}_i$, the result follows

Graph filter design

- Given the desired frequency response $\widehat{\mathbf{h}}$ of the graph filter

  $\Rightarrow$ We can find the graph coefficients $\mathbf{h}$ as

$$\mathbf{h} = \boldsymbol{\Psi}^{-1}\widehat{\mathbf{h}}$$

- Since $\boldsymbol{\Psi}$ is Vandermonde

  $\Rightarrow$ $\boldsymbol{\Psi}$ is invertible as long as $\lambda_i \neq \lambda_j$ for $i \neq j$

- For the particular case when $\mathbf{S} = \mathbf{A}_{dc}$, we have that $\lambda_i = e^{-\mathrm{j}\frac{2\pi}{N}(i-1)}$

$$\boldsymbol{\Psi} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & e^{-\mathrm{j}\frac{2\pi(1)(1)}{N}} & \dots & e^{-\mathrm{j}\frac{2\pi(1)(N-1)}{N}} \\ \vdots & \vdots & & \vdots \\ 1 & e^{-\mathrm{j}\frac{2\pi(N-1)(1)}{N}} & \dots & e^{-\mathrm{j}\frac{2\pi(N-1)(N01)}{N}} \end{pmatrix} = \mathbf{F}$$

  $\Rightarrow$ The frequency response is the DFT of the impulse response

$$\widehat{\mathbf{h}} = \mathbf{F}\mathbf{h}$$

Application: Gene Network

Graph Signals

Graph Laplacian

Graph Fourier Transform (GFT)

Ordering of frequencies

Inverse graph Fourier transform (iGFT)

Graph Filters

Application: Gene Network

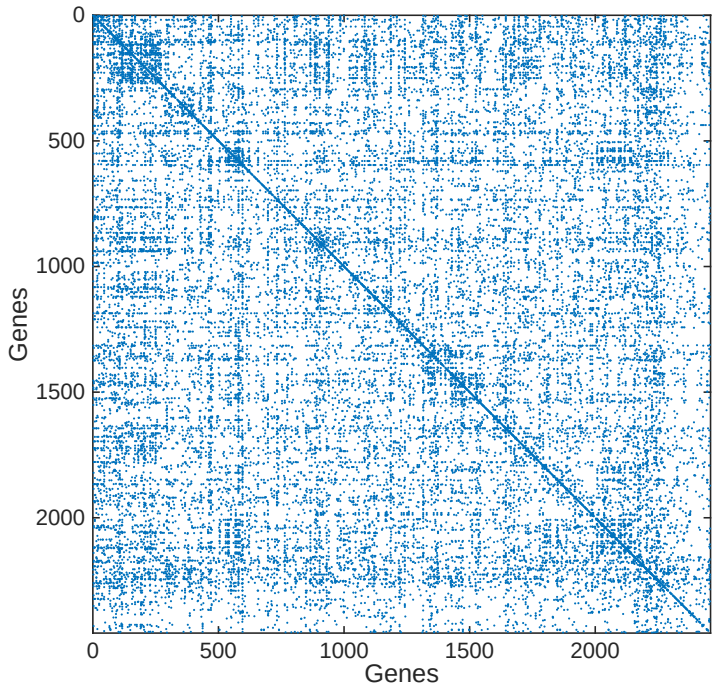
Information sciences at ESE

Motivation

- Patients diagnosed with same disease exhibit different behaviors
- Each patient has a genetic profile describing gene mutations
- Would be beneficial to infer phenotypes from genotypes
  $\Rightarrow$ Targeted treatments, more suitable suggestions, etc.
- Traditional approaches consider different genes to be independent
  $\Rightarrow$ Not so ideal, as different genes may affect same metabolism
- Alternatively, consider genetic network
  $\Rightarrow$ Genetic profiles becomes graph signals on genetic network
  $\Rightarrow$ We will see how this consideration improves subtype classification

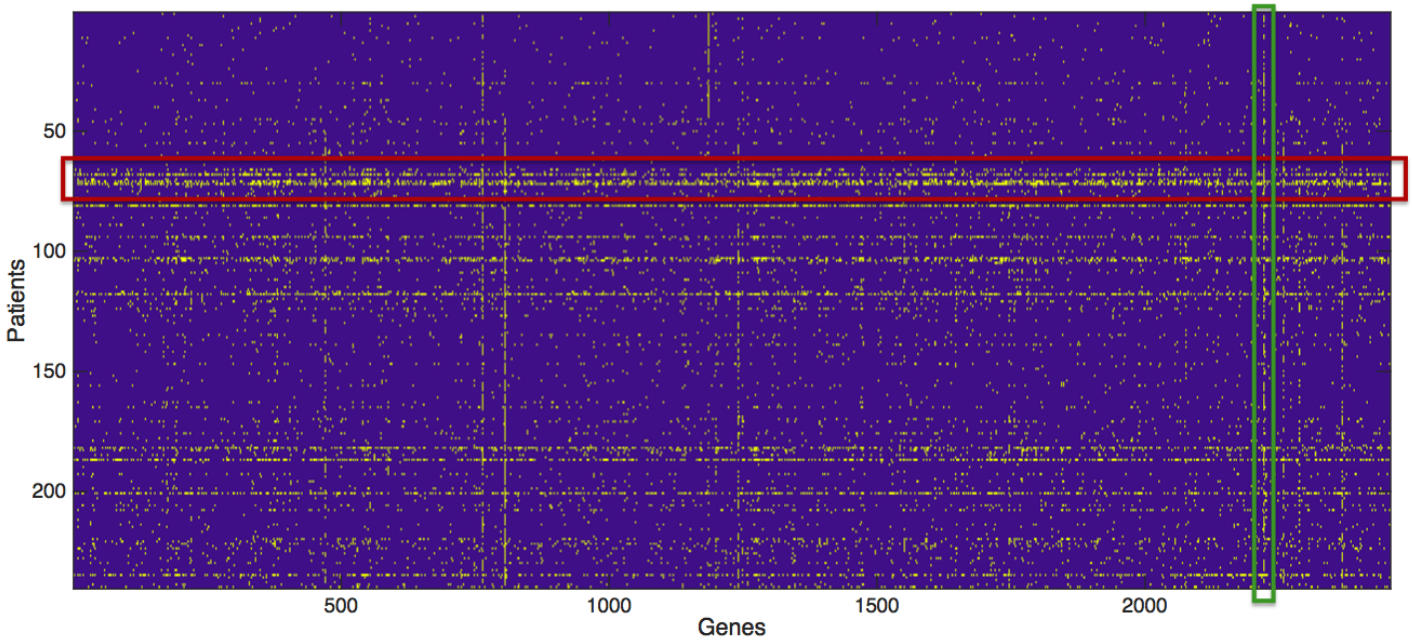
Genetic network

- Undirected and unweighted graph with 2458 nodes
  $\Rightarrow$ Describes gene-to-gene interactions
- Each node represents a gene in human DNA related to breast cancer
- An edge between two genes represents interaction
  $\Rightarrow$ Proteins encoded participate in the same metabolism process
- Adjacency matrix of the gene network



Genetic profiles

- Genetic profile of 240 women with breast cancer
  - $\Rightarrow$ 44 with serous subtype and 196 with endometrioid subtype
  - $\Rightarrow$ Patient i has an associated profile $\mathbf{x}_i \in \{0,1\}^{2458}$
- Mutations are very varied across patients
  - $\Rightarrow$ Some patients present a lot of mutations
  - $\Rightarrow$ Some genes are consistently mutated across patients



- Can we use the genetic profile to classify patients across subtypes?

k-nearest neighbor classification

- Quantify the distance between genetic profiles
  $\Rightarrow d(i,j) = \|\mathbf{x}_i - \mathbf{x}_j\|_2$
- Given a patient i to classify, all other patients' subtypes are known
- Find the k most similar profiles, i.e. j such that $d(i,j)$ is minimized
  $\Rightarrow$ Assign to i the most common subtype among these k neighbors
- Compare estimated with real subtype $\mathbf{y}$ for all patients
- We obtain the following error rates

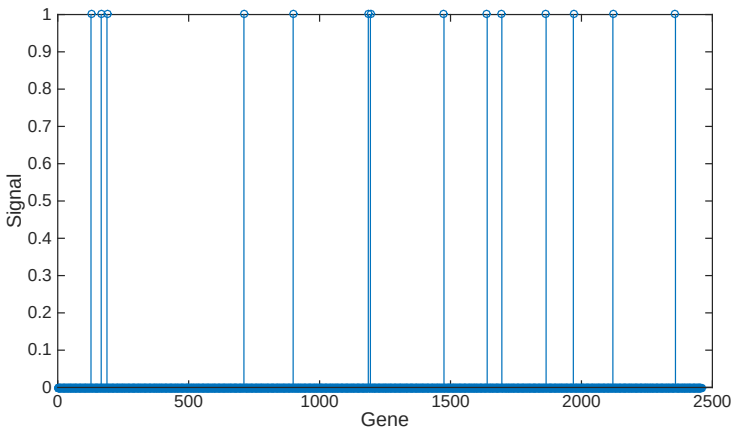
$$k = 3 \Rightarrow 13.3\%, \qquad k = 5 \Rightarrow 12.9\%, \qquad k = 7 \Rightarrow 14.6\%$$

- Can we do any better using graph signal processing?

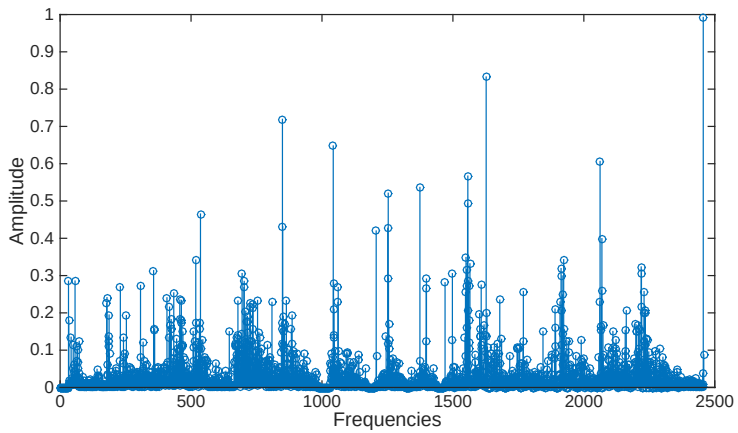
Genetic profile as a graph signal

- Each genetic profile $\mathbf{x}_i$ can be seen as a graph signal
  $\Rightarrow$ On the genetic network
- We can look at the frequency components $\tilde{\mathbf{x}}_i$ using the GFT
  $\Rightarrow$ Use as shift operator $\mathbf{S}$ the Laplacian of the genetic network

Example of signal $\mathbf{x}_i$



Frequency representation $\tilde{\mathbf{x}}_i$

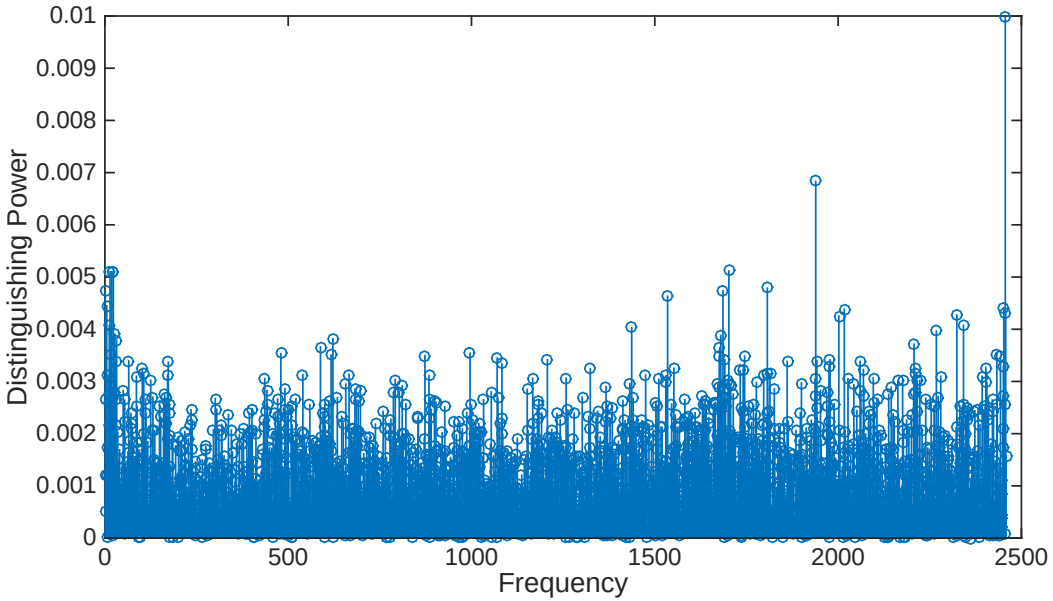


Distinguishing Power

- Define the distinguishing power of frequency $\mathbf{v}_k$ as

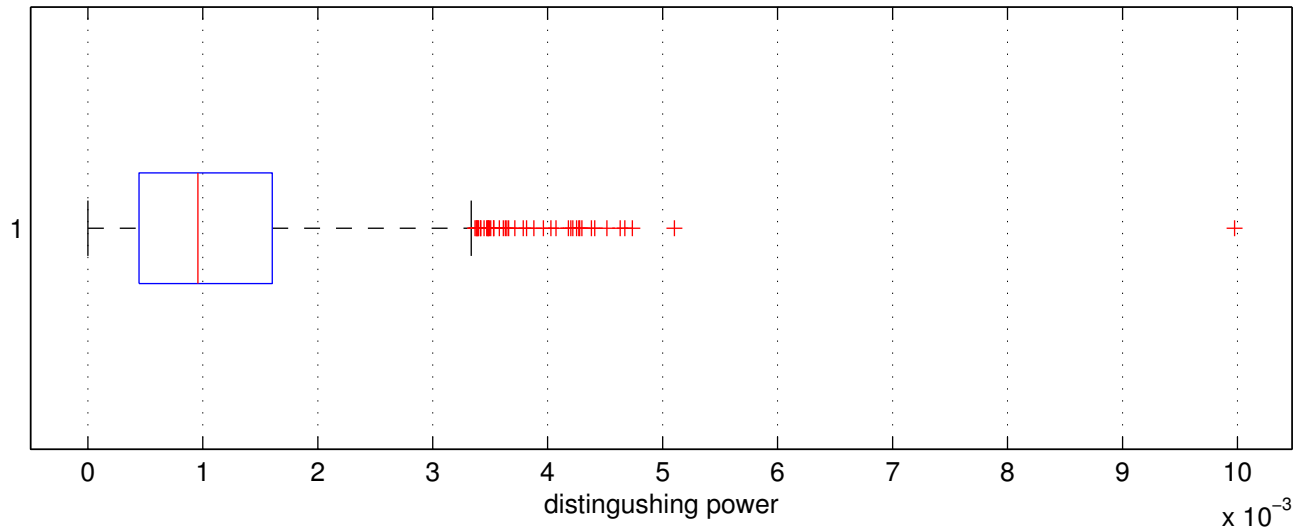
$$DP(\mathbf{v}_k) = \left| \frac{\sum_{i:y_i=1} \tilde{\mathbf{x}}_i(k)}{\sum_i \mathbf{1}\{y_i = 1\}} - \frac{\sum_{i:y_i=2} \tilde{\mathbf{x}}_i(k)}{\sum_i \mathbf{1}\{y_i = 2\}} \right| / \sum_i |\tilde{\mathbf{x}}_i(k)| ,$$

- Normalized difference between the mean GFT coefficient for $\mathbf{v}_k$
  $\Rightarrow$ Among patients with serous and endometrioid subtypes
- Distinguishing power is not equal across frequencies



Distribution of distinguishing powers

- The distribution of discriminating power



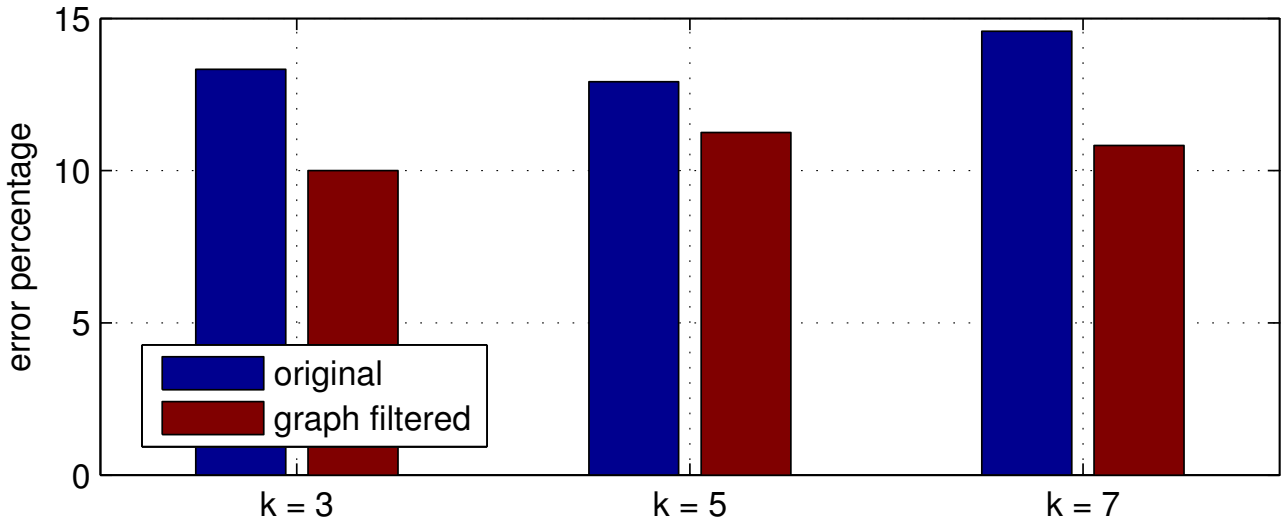
- Most frequencies have weak distinguishing power
  - $\Rightarrow$ A few frequencies have strong differentiating power
  - $\Rightarrow$ The most powerful frequency outperforms others siginificantly
- The distinguishing power defined is one of many proper heuristics

Increasing accuracy via graph filters

- Keeps only information in the most distinguishable frequency
- For the genetic profile $\mathbf{x}_i$ with its frequency representation $\tilde{\mathbf{x}}_i$
- Multiply $\tilde{\mathbf{x}}_i$ with graph filter H_1 having the frequency response

$$H_1(k) = \begin{cases} 1, & \text{if } k = \text{argmax}_k \, DP(\mathbf{v}_k); \\ 0, & \text{otherwise.} \end{cases}$$

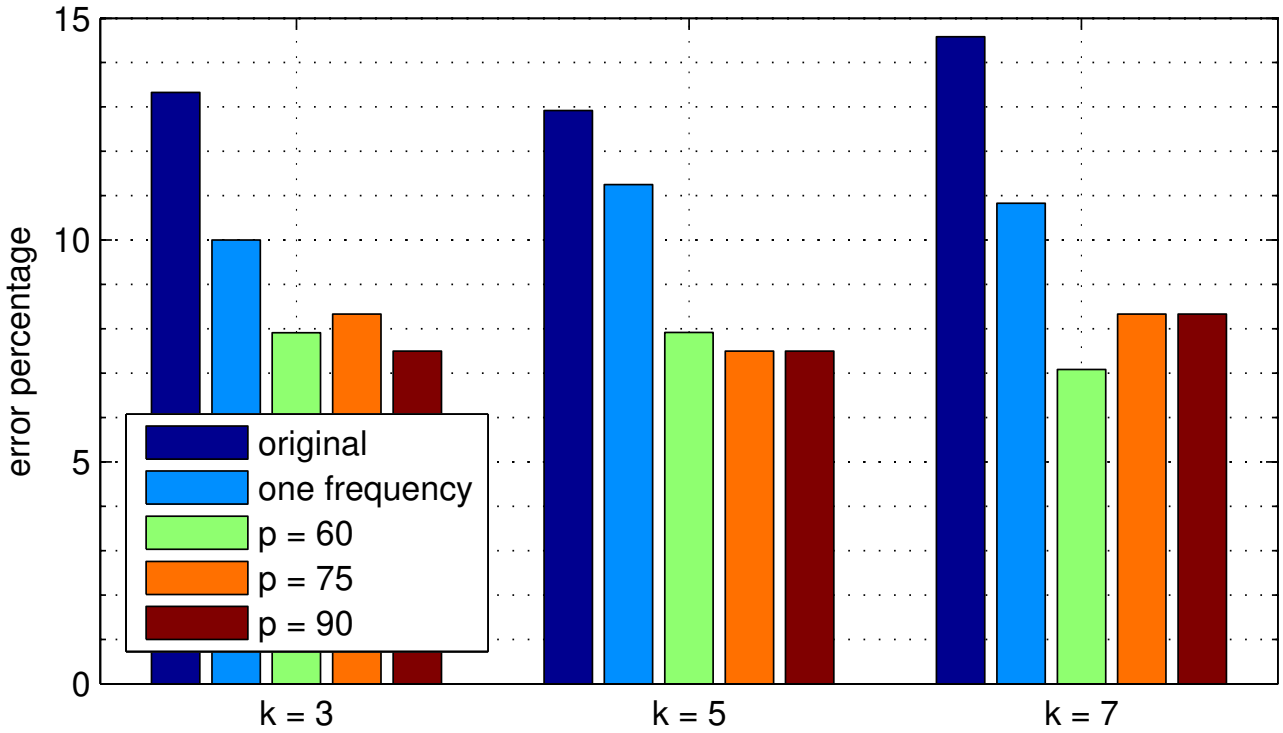
- Then perform inverse GFT to get the filtered graph signals $\hat{\mathbf{x}}_i$



Increasing accuracy via another graph filters

- Keeps information in frequencies with higher distinguishing power
- Multiply $\tilde{\mathbf{x}}_i$ with graph filter H_p having the frequency response

$$H_p(k) = \begin{cases} 1, & \text{if } DP(\mathbf{v}_k) \geq p\text{-th percentile of the distribution of } DP; \\ 0, & \text{otherwise,} \end{cases}$$



Information sciences at ESE

Graph Signals

Graph Laplacian

Graph Fourier Transform (GFT)

Ordering of frequencies

Inverse graph Fourier transform (iGFT)

Graph Filters

Application: Gene Network

Information sciences at ESE

Data Sciences (DS) Sequence

- Transforms, PCA mostly, are the brain of data analysis
  - $\Rightarrow$ But you still need a heart.
- Transforms concentrates information in a space of lower dimensionality
  - $\Rightarrow$ But you still have to extract that information
  - $\Rightarrow$ We did that with minimal sophistication (nearest neighbors)
- If you think you want to learn more about this, follow the DS sequence
  - $\Rightarrow$ ESE224: Signal and Information Processing
  - $\Rightarrow$ ESE305: Foundations of Data Science (V. Preciado)
  - $\Rightarrow$ ESE545: Data Mining: Learning From Massive Datasets (H. Hassani)
- Penn is also starting a masters of data science. Consider sub-matriculating

The systems and signal processing sequences

- To follow up on what we did on the first half of the course you can ...
  - $\Rightarrow$ Follow up literally by digging deeper intro signal processing
  - $\Rightarrow$ Follow up philosophically by studying systems
- Signal Processing sequence
  - $\Rightarrow$ ESE224: Signal and Information Processing
  - $\Rightarrow$ ESE325: Fourier Analysis and Applications
  - $\Rightarrow$ ESE531: Digital Signal Processing
- Systems analysis and design
  - $\Rightarrow$ ESE210: Introduction to Dynamic Systems (R. Ghrist)
  - $\Rightarrow$ ESE224: Signal and Information Processing
  - $\Rightarrow$ ESE303: Stochastic Systems Analysis and Simulation

Controls and Optimization sequences

- Once you have information you may want to something with it
- Controlling the state of a system
  - $\Rightarrow$ ESE406: Control of Systems
  - $\Rightarrow$ ESE500: Linear Systems Theory
- Making decisions that are good in some sense (optimal)
  - $\Rightarrow$ ESE204: Decision Models
  - $\Rightarrow$ ESE304: Optimization of Systems
  - $\Rightarrow$ ESE504: Introduction to Optimization Theory
  - $\Rightarrow$ ESE605: Modern Convex Optimization

Even more courses to consider

- At some point, you want to use what you've learned to do something
  - $\Rightarrow$ ESE290: Introduction to ESE Research Methodology
  - $\Rightarrow$ ESE350: Embedded Systems/Microcontroller Laboratory

Thanks

- It has been my pleasure. I am very happy abut how things turned out
- If you need my help at some point in the next 30 years, let me know
- I will be retired after that