Computation Methods
in NeuroImage Analysis

Instructor: Moo K. Chung
mkchung@wisc.edu

Lecture 11
Compressed sensing

November 19, 2010
Title: Link communities reveal multiscale complexity in networks

Abstract: Networks have become a key approach to understanding systems of interacting objects, unifying the study of diverse phenomena including biological organisms and human society. One crucial step when studying the structure and dynamics of networks is to identify communities: groups of related nodes that correspond to functional subunits such as protein complexes or social spheres. Communities in networks often overlap such that nodes simultaneously belong to several groups. Meanwhile, many networks are known to possess hierarchical organization, where communities are recursively grouped into a hierarchical structure. However, the fact that many real networks have communities with pervasive overlap, where each and every node belongs to more than one group, has the consequence that a global hierarchy of nodes cannot capture the relationships between overlapping groups. Here we reinvent communities as groups of links rather than nodes and show that this unorthodox approach successfully reconciles the antagonistic organizing principles of overlapping communities and hierarchy. In contrast to the existing literature, which has entirely focused on grouping nodes, link communities naturally incorporate overlap while revealing hierarchical organization. We find relevant link communities in many networks, including major biological networks such as protein-protein interaction and metabolic networks, and show that a large social network contains hierarchically organized community structures spanning inner-city to regional scales while maintaining pervasive overlap. Our results imply that link communities are fundamental building blocks that reveal overlap and hierarchical organization in networks to be two aspects of the same phenomenon.
Final Exam will be emailed to you at exactly 5:00am on December 3. You have 7 hours to email back your solutions. I will only accept PDF. The solution has to be emailed to me by Noon. I won’t accept multiple submissions. The first submission will be graded so submit carefully. After Noon, 10% deduction/hour will apply. The final exam is 30-40% of your final grade. Depending on the performance I will fix the percentage.

If you followed my lectures carefully, it should be done by 3 hours. There are 6 problems. Three problems will involve MATLAB programming.
Compressed sensing, also known as compressive sampling and sparse sampling, is a technique for finding sparse solutions to underdetermined systems (wiki definition).
About 5000 articles in google scholar with word “compressed sensing” in it.
Underdetermined system

\[ b = Ax \]

There are more parameters than measurements.

There are infinite number of solutions.

However, if \( x \) is sparse, we can exactly recover \( x \).
L1 norm minimization

\[ \| x \|_1 = \sum_i |x_i| \]

Basis pursuit

\[ \min_{Ax=b} \| x \|_1 \]

If \( x \) is sufficiently sparse, the basis pursuit will find it.

Matlab demonstration
\[ b = Ax + \text{noise} \]

LASSO

Equivalent formulation

\[ \min \left\| x \right\|_1 \]
\[ \| b - Ax \|_2 \leq \epsilon \]

Equivalent formulation

\[ \min \left\| b - Ax \right\|_2^2 + \lambda \left\| x \right\|_1 \]

Matlab demonstration
In science and medical imaging in particular, it is usually assumed that the $i$-th functional measurement $f_i(x)$ at position $x \in \mathcal{M} \subset \mathbb{R}^d$ to follow

$$f_i(x) = \mu(x) + \epsilon_i(x),$$

(1)

where $\mu$ is the unknown mean signal to be estimated, $\epsilon_i$ are noise and $\mathcal{M}$ is the underlying manifold where the data is observed [1, 2, 3, 4, 5, 6]. The unknown signal is usually estimated by various spatial smoothing techniques over the manifold $\mathcal{M}$. $\mathcal{M}$ can be the brain cortical surface (Figure 1) or 3D brain network graphs (Figure 4).
Signal detection via random field theory

image \( f_1 \)  
image \( f_2 \)  
statistic \( T \)

excursion sets \( B_{1,h} \)  
excursion sets \( B_{2,h} \)  
excursion sets \( A_h \)  

threshold \( h \)
In order to compute the type-I error associated with $H_0$, we need to know the distribution of the supremum of the field $T(x)$, which is not straightforward. Hence a great deal of the imaging and statistical literature have been devoted to determining the distribution of $\sup_{x \in \mathcal{M}} T(x)$ [7, 8, 9, 6, 10]. Define the excursion set as

$$A_h = \{x \in \mathcal{M} : T(x) > h\}.$$  

It is known that

$$P\left(\sup_{x \in \mathcal{M}} T(x) > h\right) \approx E\chi(A_h), \tag{3}$$

the expectation of the Euler characteristic of the excursion set $A_h$ [11, 12, 13]. The relationship (3) reformulates the usual statistical inference as a topological problem. Figure 2 shows how the excursion set changes over increasing threshold $h$. 
The above framework is one way of detecting signal and not necessarily the best way to characterize complex multivariate functional data such as brain MRI. Instead of looking at the topological change of the excursion set of the statistic $T$, which is a function of $f_i$, we look at the topological change of measurements $f_i$ first. Let

$$B_{i,h} = \{x \in \mathcal{M} : f_i(x) > h\}$$

be the excursion set associated with the $i$-th measurement. Then we determine the topological structure of $B_{i,h}$ first, and perform a statistical inference later (Figure 2). The main tool for investigating the topological change of the excursion set is persistent homology.
Heat kernel smoothing of cortical thickness

Delaunay triangulation on critical values

Persistent homology
Network modeling. Let $p$ be the number of nodes in the network. In most applications, the number of nodes are expected to be larger than the number of observations $n$, which gives an underdetermined system. The $i$-th measurement $f_i$ are then discretely sampled at $p$ nodes, which we will simply index by integers. To simplify the notation, we denote $x_{ij} = f_i(j)$. At node $j$, we have random variable $x_j$, which is realized by $x_{1j}, \ldots, x_{nj}$. We will denote this realization as $\mathbf{x}_j = (x_{1j}, \ldots, x_{nj})'$. The measurement $x_j$ are assumed to be distributed with mean zero and covariance $\Sigma = (\sigma_{ij})$. i.e.

$$\mathbb{E}x_i = 0, \quad \mathbb{E}(x_i x_j) = \sigma_{ij}.$$  

If $\mathbb{E}x_i \neq 0$, we can always center the data by translation. The correlation $\gamma_{ij}$ between the two nodes $i$ and $j$ is given by

$$\gamma_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii} \sigma_{jj}}}.$$
If we denote the inverse covariance matrix as $\Sigma^{-1} = (\sigma^{-1}_{ij})$, the partial correlation between the nodes $i$ and $j$ while factoring out the effect of all other nodes is given by

$$\rho_{ij} = -\frac{\sigma_{ij}}{\sqrt{\sigma_{ii}\sigma_{jj}}}.$$ (4)

Equivalently, we can compute the partial correlation via a linear model as follows. Consider a linear model of correlating measurement at node $i$ to all other nodes:

$$x_i = \sum_{j \neq i} \beta_{ij} x_j + \epsilon_i.$$ (5)

The parameters $\beta_{ij}$ are estimated by minimizing the sum of squared residual of (5)

$$L(\beta) = \sum_{i=1}^{p} \left\| x_i - \sum_{j \neq i} \beta_{ij} x_j \right\|^2.$$ (6)
in a least squares fashion. If we denote the least squares estimation as $\hat{\beta}_{ij}$, the residuals are given by

$$r_i = x_i - \sum_{j \neq i} \hat{\beta}_{ij} x_j. \quad (7)$$

The partial correlation is then obtained by computing the correlation between the residuals of the model fit (5) [22, 27, 26]:

$$\rho_{ij} = \mathbb{E}[(r_i - \mathbb{E}r_i)(r_j - \mathbb{E}r_j)].$$
The minimization of (6) is exactly given by solving the normal equation:

\[ x_i = \sum_{j \neq i} \beta_{ij} x_j, \quad (8) \]

which can be made into a standard linear form \( y = A\beta \) via an algebraic manipulation [28]. Note that (8) can be written as

\[
\begin{align*}
  x_i &= \begin{pmatrix} x_1, \cdots, x_{i-1}, 0, x_{i+1}, \cdots, x_p \end{pmatrix}^T, \\
  A &= \begin{pmatrix} \beta_{i1} \\ \beta_{i2} \\ \vdots \\ \beta_{ip} \end{pmatrix}, \\
  \beta &= \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix}.
\end{align*}
\]
where $0_{n \times 1}$ is a column vector of all zero entries. Then we have

$$
\begin{pmatrix}
    x_1 \\
    x_2 \\
    \vdots \\
    x_p
\end{pmatrix}_{y_{np \times 1}} =
\begin{pmatrix}
    X_{-1} & 0 & \cdots & 0 \\
    0 & X_{-2} & \cdots & 0 \\
    \vdots & \vdots & \ddots & \vdots \\
    0 & 0 & \cdots & X_{-p}
\end{pmatrix}_{A_{np \times p^2}}
\begin{pmatrix}
    \beta_1 \\
    \beta_2 \\
    \vdots \\
    \beta_p
\end{pmatrix}_{\beta_{p^2 \times 1}},
$$

(9)

where $A$ is a block diagonal matrix and $0_{n \times p}$ is a matrix of all zero entries.

When $n=30$, $p=169$, A matrix is of size $5577 \times 28561$.
When $p > 1000$, it becomes a challenging computational problem.
\[ J = \sum_{i,j} |\beta_{ij}|. \]

The sparse estimation of \( \beta_{ij} \) is then given by minimizing \( L + \lambda J \). Since there is dependency between \( y \) and \( A \), (9) is not exactly a standard compressed sensing problem but it is reasonable to treat it as one by simply ignoring the dependency [26, 28]. It should be intuitively understood that the sparsity makes the linear equation (8) less underdetermined. The larger the value of \( \lambda \), more sparse the underlying topological structure gets. Since

\[ \rho_{ij} = \beta_{ij} \sqrt{\frac{\sigma_{ii}}{\sigma_{jj}}}, \]

the sparsity of \( \beta_{ij} \) directly corresponds to sparsity of \( \rho_{ij} \), which is the strength of the link between nodes \( i \) and \( j \) [26, 28]. Once the
Partial correlation matrix

Least squares estimation        LASSO with $\lambda=100$
Sparse brain network obtained with different lambda

Matlab demonstration
Equivalent formulation

Then we estimate either $\beta_{ij}$ by solving

$$\min_{\beta} L(\beta) + \lambda J(\beta)$$

(10.3)

for some tuning parameter $\lambda \leq 0$. The larger the value of $\lambda$, more sparsity constraint we are enforcing. (10.3) can be equivalently formulated as

$$\min_{\beta} L(\beta) \text{ subject to } J(\beta) \leq \epsilon$$
Connection to likelihood approach

Others have proposed the likelihood methods. The Gaussian log-likelihood of data $X$ with the covariance matrix $\Sigma$ is given by

$$L(\Sigma^{-1}) = \log \det \Sigma^{-1} - \text{tr}(SS^{-1}) - \rho \|\Sigma^{-1}\|_1,$$

where $S$ is the sample covariance, $\| \cdot \|_1$ is the sum of the absolute value of the matrix entries and $\rho > 0$ controls the sparsity of solution (Banerjee et al., 2006, 2008; Friedman et al., 2008). This needs to be maximized over all positive-definite matrices numerically:

$$\hat{\Sigma}^{-1} = \arg \max_{\Sigma > 0} \log \det \Sigma^{-1} - \text{tr}(SS^{-1}) - \rho \|\Sigma^{-1}\|_1$$  \hspace{1cm} (10.4)

The relationship of (10.4) to LASSO framework is explored in (Friedman et al., 2008).
Computational bottleneck for large $p$. What do we do with $p = 10000$?

nodes. One practical solution is to modify (5) so that the measurement at node $i$ is represented more sparsely over some possible index set $S_i$:

$$x_i = \sum_{S_i} \beta_{ij} x_j + \epsilon_i.$$  

making the problem substantially smaller.
Lecture 12 Topics – last lecture

More on network complexity, complexity in general, fractional dimension (FD).

Read

ahn.2010.nature.linknetwork.pdf
rubinov.2010.Nl.network.pdf
esteban.2007.Nl.fractal.pdf