• L12 - Introduction to Protein Structure; Structure Comparison & Classification
• L13 - Predicting protein structure
• L14 - Predicting protein interactions
• L15 - Gene Regulatory Networks
• L16 - Protein Interaction Networks
• L17 - Computable Network Models
Wisdom of crowds for robust gene network inference

Daniel Marbach, James C Costello, Robert Küffner, Nicole M Vega, Robert J Prill, Diogo M Camacho, Kyle R Allison, The DREAM5 Consortium, Manolis Kellis, James J Collins & Gustavo Stolovitzky

Affiliations  |  Contributions  |  Corresponding author

Received 31 October 2011  |  Accepted 22 May 2012  |  Published online 15 July 2012
Wisdom of crowds for robust gene network inference
Wisdom of crowds for robust gene network inference
AUPR = area under precision-recall curve

Wisdom of crowds for robust gene network inference

Courtesy of Macmillan Publishers Limited. Used with permission.
AUPR = area under precision-recall curve

Wisdom of crowds for robust gene network inference
Wisdom of crowds for robust gene network inference
Wisdom of crowds for robust gene network inference

Wisdom of crowds for robust gene network inference
Wisdom of crowds for robust gene network inference
Wisdom of crowds for robust gene network inference
Thoughts on Gene Expression Data

• Useful for classification and clustering
• Not sufficient for reconstructing regulatory networks in yeast
• Can we infer levels of proteins from gene expression?
Approach

mRNA levels do not predict protein levels

\[ R^2 = 0.22, \quad R_s = 0.46 \]

1,000 fold range of protein concentrations

---


Global quantification of mammalian gene expression control.
Schwanhäusser B1, Busse D, Li N, Dittmar G, Schuchhardt J, Wolf J, Chen W, Selbach M.
Global quantification of mammalian gene expression control.
Schwanhäusser B1, Busse D, Li N, Dittmar G, Schuchhardt J, Wolf J, Chen W, Selbach M.
Global quantification of mammalian gene expression control.

Schwanhäusser B1, Busse D, Li N, Dittmar G, Schuchhardt J, Wolf J, Chen W, Selbach M.
Global quantification of mammalian gene expression control.
Schwanhäusser B1, Busse D, Li N, Dittmar G, Schuchhardt J, Wolf J, Chen W, Selbach M.
Global quantification of mammalian gene expression control.
Schwanhäusser B1, Busse D, Li N, Dittmar G, Schuchhardt J, Wolf J, Chen W, Selbach M.
Global quantification of mammalian gene expression control.
Schwanhäusser B1, Busse D, Li N, Dittmar G, Schuchhardt J, Wolf J, Chen W, Selbach M.

Strategies:
1. Use expression to infer upstream events
2. Explicitly model downstream steps
L18 Chromatin and DNase-seq Analysis

Mutant

Wild-type

Sequence Analysis

Sequence Analysis

Logo

- M01282
- M01201
- M01214
Move upstream of transcription

Network integration

Epigenomic Data & Sequence Analysis

DNA-binding proteins

mRNA
Strategies:

1. Use expression to infer upstream events
2. Explicitly model downstream steps
Inference of patient-specific pathway activities from multi-dimensional cancer genomics data using PARADIGM

Charles J. Vaske¹,†, Stephen C. Benz²,†, J. Zachary Sanborn², Dent Earl², Christopher Szeto², Jingchun Zhu², David Haussler¹,² and Joshua M. Stuart²,*

¹Howard Hughes Medical Institute and ²Department of Biomolecular Engineering and Center for Biomolecular Science and Engineering, UC Santa Cruz, CA, USA
Overview of the PARADIGM method.

Courtesy of Vaske et al. License: CC-BY.

© The Author(s) 2010. Published by Oxford University Press.
Factor graphs generalize Bayesian networks.

Bayesian network

Factor graph
Factor graphs

• Bipartite graph
  (means there are two types of nodes)
• Describes how a global function can be factored into a product of local functions
• Bayesian networks are a type of factor graph
Factor graphs

Global function of the variables: \( g(x_1, x_2, x_3) = \prod_{j \in J} f_j(X_j) \)

- Variable node, \( x \)
- Factor node, \( f \)
- Edge exists iff \( x \) is an argument of \( f \)

Factor graph
Factor graphs

• A node for:
  - $x_1$ — every variable and
  - $f$ — every function $f_j(X_j)$

• Node $x_i$ is connected to factor $f_j$ iff
  the variable $x_i$ appears as a term in $f_j$

\[
g(x_1, x_2, x_3) = \prod_{j \in J} f_j(X_j)
\]
In our setting

Joint probability function: \( P(x_1, x_2, x_3) = \prod_{j \in J} f_j(X_j) \)

- Variable node, \( x = \) state of gene/protein/pathway
- Factor node, \( f \) describes relationships
- Edge exists iff \( x \) is an argument of \( f \)

Factor graph
Global function: \( g(x_1, x_2, x_3, x_4, x_5) \)

Marginal \( g_i(a) \): sum \( g(x_1, x_2, x_3, x_4, x_5) \) over all configurations of the variables with \( x_i = a \)

What is the probability that MYC/MAX is active? \( P(x_i = \text{active}) \)

Factor graphs provide a method to compute such marginals.

© American Association for Cancer Research. All rights reserved. This content is excluded from our Creative Commons license. For more information, see http://ocw.mit.edu/help/faq-fair-use/.
Global function:
\[ g(x_1, x_2, x_3, x_4, x_5) = f_A(x_1) f_B(x_2) f_C(x_1, x_2, x_3) f_D(x_3, x_4) f_E(x_3, x_5) \]

Marginal \( g_i(a) \): sum \( g(x_1, x_2, x_3, x_4, x_5) \)
over all configurations of the variables with \( x_i=a \)

\[ g_1(x_1) = f_A(x_1) \times \left( \sum_{x_2} f_B(x_2) \left( \sum_{x_3} f_C(x_1, x_2, x_3) \left( \sum_{x_4} f_D(x_3, x_4) \left( \sum_{x_5} f_E(x_3, x_5) \right) \right) \right) \right) \]
Global function:
\[
g(x_1, x_2, x_3, x_4, x_5) = f_A(x_1)f_B(x_2)f_C(x_1, x_2, x_3)f_D(x_3, x_4)f_E(x_3, x_5)
\]

Marginal \( g_i(a) \) : sum \( g(x_1, x_2, x_3, x_4, x_5) \)
over all configurations of the variables with \( x_i=a \)

\[
g_i(x_i) = \sum_{\sim\{x_i\}} g(x_1, x_2, x_3, x_4, x_5)
\]

“not-sum” or summary over all values of \( x_j \neq i \)
Global function:
\[ g(x_1, x_2, x_3, x_4, x_5) = f_A(x_1) f_B(x_2) f_C(x_1, x_2, x_3) f_D(x_3, x_4) f_E(x_3, x_5) \]

Marginal \( g_i(a) \) : sum \( g(x_1, x_2, x_3, x_4, x_5) \) over all configurations of the variables with \( x_i=a \)

\[ g_1(x_1) = f_A(x_1) \times \left( \sum_{x_2} f_B(x_2) \left( \sum_{x_3} f_C(x_1, x_2, x_3) \left( \sum_{x_4} f_D(x_3, x_4) \left( \sum_{x_5} f_E(x_3, x_5) \right) \right) \right) \right) \]

\[ g_1(x_1) = f_A(x_1) \times \sum_{\sim \{x_1\}} \left( f_B(x_2) f_C(x_1, x_2, x_3) \left( \sum_{\sim \{x_3\}} f_D(x_3, x_4) \left( \sum_{\sim \{x_3\}} f_E(x_3, x_5) \right) \right) \right) \]
Global function:
\[ g(x_1, x_2, x_3, x_4, x_5) = f_A(x_1) f_B(x_2) f_C(x_1, x_2, x_3) f_D(x_3, x_4) f_E(x_3, x_5) \]

How do we find the marginal for any factor graph?
To compute the marginal with respect to variable $x_i$:
draw the factor graph as a tree with root $x_i$.
Marginal:

\[
g_1(x_1) = f_A(x_1) \times \sum_{x_1} \left( f_B(x_2) f_C(x_1, x_2, x_3) \left( \sum_{x_3} f_D(x_3, x_4) \right) \left( \sum_{x_3} f_E(x_3, x_5) \right) \right)\]
Compute product of “summary” function for parent variable

Compute “summary” function for parent variable

Marginal:
\[ g_1(x_1) = f_A(x_1) \times \]
\[ \sum_{\sim \{x_1\}} \left( f_B(x_2) f_C(x_1, x_2, x_3) \left( \sum_{\sim \{x_3\}} f_D(x_3, x_4) \right) \left( \sum_{\sim \{x_3\}} f_E(x_3, x_5) \right) \right) \]
Messages flow up from leaves:
• Each vertex waits for messages from all children before computing message to send to parents
• Variable nodes send product of messages from children
• Factor nodes with parent x send the “summary” for x of the product of the children’s functions.

Belief propagation:
An algorithm known as “Sum-Product” can be used to simultaneously compute all marginals!
See citation for details

Factor graphs in PARADIGM

- Variable node, $x$: three states:
  - 1 activated
  - 0 nominal
  - -1 deactivated

- Factor node, $f$

Edge exists iff $x$ is an argument of $f$

Courtesy of Vaske et al. License: CC-BY.

© The Author(s) 2010. Published by Oxford University Press.
Goal:

- Estimate probability that pathways are active
- Use log likelihood ratio

\[
L(i, a) = \log \left( \frac{P(D, x_i = a | \Phi)}{P(D, x_i \neq a | \Phi)} \right) - \log \left( \frac{P(x_i = a | \Phi)}{P(x_i \neq a | \Phi)} \right)
\]

\[
= \log \left( \frac{P(D | x_i = a, \Phi)}{P(D | x_i \neq a, \Phi)} \right).
\]

Parameters estimated by EM from experimental data

Manually constructed

Known pathways:
• Convert to a directed graph
• Each edge is labeled as either positive or negative based on influence
• Define joint probability

Courtesy of Vaske et al. License: CC-BY.
Defining joint probability

Expected state:
• Majority vote of parent variables
• If a parent is connected by a positive edge it contributes a vote of +1 times its own state to the value of the factor.
• If the parent is connected by a negative edge, then the variable votes −1 times its own state.

\[
\phi_i(x_i, \text{Parents}(x_i)) = \begin{cases} 
1 - \epsilon & \text{if } x_i \text{ is the expected state from Parents}(x_i) \\
\frac{\epsilon}{2} & \text{otherwise.}
\end{cases}
\]

\(\epsilon\) was set to 0.001
Defining factors manually

\[ \phi_i(x_i, \text{Parents}(x_i)) = \begin{cases} 
1 - \epsilon & \text{if } x_i \text{ is the expected state from Parents}(x_i) \\
\epsilon / 2 & \text{otherwise.} 
\end{cases} \]

\( \epsilon \) was set to 0.001

Courtesy of Vaske et al. License: CC-BY.

Logic:

• **AND**: The variables connected to \( x_i \) by an edge labeled ‘minimum’ get a single vote, and that vote's value is the minimum value of these variables.

• **OR**: The variables connected to \( x_i \) by an edge labeled ‘maximum’ get a single vote, and that vote's value is the maximum value of these variables, creating an OR-like connection.

• Votes of zero are treated as abstained votes.

• If there are no votes the expected state is zero. Otherwise, the majority vote is the expected state, and a tie between 1 and −1 results in an expected state of −1 to give more importance to repressors and deletions.
Defining factors manually

\[ \phi_i(x_i, \text{Parents}(x_i)) = \begin{cases} 1 - \epsilon & \text{if } x_i \text{ is the expected state from Parents}(x_i) \\ \epsilon/2 & \text{otherwise} \end{cases} \]

\( \epsilon \) was set to \( 0.001 \)

Logic:

• **AND**: The variables connected to \( x_i \) by an edge labeled ‘minimum’ get a single vote, and that vote's value is the minimum value of these variables

• **OR**: The variables connected to \( x_i \) by an edge labeled ‘maximum’ get a single vote, and that vote's value is the maximum value of these variables, creating an OR-like connection.

Compared to Bayesian networks, factor graphs provide an more intuitive way to represent these regulatory steps.
Joint probability of graph

\[ \phi_i(x_i, \text{Parents}(x_i)) = \begin{cases} 
1 - \epsilon & \text{if } x_i \text{ is the expected state from Parents}(x_i) \\
\frac{\epsilon}{2} & \text{otherwise.} 
\end{cases} \]

\[ P(X) = \frac{1}{Z} \prod_{j=1}^{m} \phi_j(X_j), \]

\[ Z = \prod_j \sum_{S \subseteq X_j} \phi_j(S) \]

Setting of variables = possible values
Marginal

\[ P(x_i = a | \Phi) = \frac{1}{Z} \prod_{j=1}^{m} \sum_{S \sqcap_{A_i(a)} X_j} \phi_j(S) \]

\( \{S \sqcap_{D} X\} \) Set of all possible assignments to the variables X consistent with data D

\( A_i(a) \) represents the singleton assignment set \( \{x_i = a\} \)

\( \Phi \) Full specified factor graph

Likelihood

\[ P(x_i = a, D | \Phi) = \frac{1}{Z} \prod_{j=1}^{m} \sum_{S \sqcap_{A_i(a) \cup D} X_j} \phi_j(S) \]
• genomic copies (G)
• epigenetic promoter state (E)
• mRNA transcripts (T)
• peptide (P)
• active protein (A).
• Regulation gene expression
  • transcriptional (RT)
  • translational (RP)
  • post-translational (RA)
© American Association for Cancer Research. All rights reserved. This content is excluded from our Creative Commons license. For more information, see http://ocw.mit.edu/help/faq-fair-use/.
"MYC/MAX ... is active because one of its known activated targets (CCNB1) is highly expressed while one of its repressed targets (WNT5A) has lower expression."

What about ENO1, which should be increasing? Note lack of epigenetic change.
increased
normal
lower
Reasoning on curated pathways

Courtesy of Vaske et al. License: CC-BY.
Network Models

• Structure of network
  – Coexpression
  – Mutual information
  – Physical/genetic interactions

• Analysis of network
  – Ad hoc
  – Shortest path
  – Clustering
  – Optimization
Graph Algorithms for Interaction Networks

• Rich area of computer science

• Applications to Interaction Networks:
  – Distances:
    • Finding kinase substrates
  – Clustering
    • PPI->Protein complexes, functional annotation
    • Coexpression -> Modules
    • Blast ->Protein families
  – Active subnetworks
    • Finding hidden components of processes
If I know a protein has been phosphorylated, can I determine the kinase?


**Step 1: Use sequence motifs to determine family of kinase**

![Diagram showing matching of sequence motifs for kinase families](image)


Step 1: Use sequence motifs to determine family of kinase

Step 2: Use Interactome data to find most likely family member


Linding et al. (2007) Cell. doi:10.1016/j.cell.2007.05.052
Which is best?

- High-throughput Two-hybrid assay
- High confidence interaction

Motif match
High confidence interaction
How do we find the closest kinase?

• Many efficient algorithms exist once we treat our problem as one in Graph Theory.
Graph Terminology

• $G=(V,E)$

• Undirected vs. directed

• Weights – numbers assigned to each edge

• Degree($v$) – number of edges incident on $v$
  - In-degree and out-degree

• Path from $a$ to $b$ is a series of vertices $<a, v_0, ..., b>$ where edges exist between sequential vertices

• Path length = sum of edges weights (or number of edges) on path.
Data Structure

Adjacency Matrix

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Weights can represent our confidence in the link.

Weighted graph: \( a_{ij} = w_{ij} \) if edge exists; 0 otherwise.

Adjacency Matrix:

\[
\begin{array}{ccccc}
 & 1 & 2 & 3 & 4 & 5 \\
1 & 0 & .5 & 0 & 0 & 0 \\
2 & .5 & 0 & 1 & 0 & 1 \\
3 & 0 & 1 & 0 & .2 & 0 \\
4 & 0 & 0 & .2 & 0 & 0 \\
5 & 0 & 1 & 0 & 0 & 0 \\
\end{array}
\]
Shortest Path Algorithms

• Efficient Algorithms for
  – single pair \((u,v)\)
  – single source/destination to all other nodes
  – all-pairs
Reliability of edges

- Assign weight to each edge based on reliability.
- Total distance in network = sum of edge weights
- If $\text{weight}_{ij} = -\log(P_{ij})$:
  \[
  \min \sum w_{ij} = \min (-\log \prod P_{ij})
  = \max \text{ (joint probability)}
  = \text{most probable path}
  \]
Interaction Weights

• How do we assign reliability of edges?
A Bayesian Networks Approach for Predicting Protein-Protein Interactions from Genomic Data

Ronald Jansen,1* Haiyuan Yu,1 Dov Greenbaum,1 Yuval Kluger,1 Nevan J. Krogan,4 Sambath Chung,1,2 Andrew Emili,4 Michael Snyder,2 Jack F. Greenblatt,4 Mark Gerstein1,3†

B

In vivo pull-down

Gavin
Ho
Uetz
Y2H
Ito

Fully connected Bayes

\[\text{PIE}\]

Naïve Bayes

\[\text{PIT}\]

Probabilistic interactome (PI)
Integration process
Data source

mRNA co-exp:

Rosetta
Cell cycle
GO process
MIPS function
Essentiality

© American Association for the Advancement of Science. All rights reserved. This content is excluded from our Creative Commons license. For more information, see http://ocw.mit.edu/help/faq-fair-use/. Source: Jansen, Ronald, Haiyuan Yu, et al. "A Bayesian Networks Approach for Predicting Protein-Protein Interactions from Genomic Data." Science 302, no. 5644 (2003): 449-53.
PSICQUIC and PSISCORE: accessing and scoring molecular interactions
Nature Methods 8, 528–529 (2011)
doi:10.1038/nmeth.1637

Human Proteome Organization Proteomics Standards Initiative (HUPO-PSI) released the PSI molecular interaction (MI) XML format

PSI common query interface (PSICQUIC), a community standard for computational access to molecular-interaction data data resources.

http://www.nature.com/nmeth/journal/v8/n7/full/nmeth.1637.html
Miscore is a normalized score between 0 and 1 that takes into account several variables:

- Number of publications
- Experimental detection methods found for the interaction
- Interaction types found for the interaction

Each of these variables is also represented by a score between 0 and 1. The importance of each variable in the main equation can be adjusted using a weight factor.
Miscore algorithm

\[ S_{MI} = \frac{K_p \times S_p(n) + K_m \times S_m(cv) + K_t \times S_t(cv)}{K_p - K_m + K_t} \]

Depends on

- Number of publications
- Experimental method (biophys.; imaging; genetic)
- Annotation of interaction type (physical, genetic)
Finding Modules

- **Topological module:**
  - locally dense
  - more connections among nodes in module than with nodes outside module

- **Functional module:**
  - high density of functionally related nodes

Courtesy of Macmillan Publishers Limited. Used with permission.
Can we use networks to predict function

Network-based prediction of protein function
Roded Sharan, Igor Ulitsky & Ron Shamir
doi:10.1038/msb4100129

Based on the Entrez Gene and the WormBase databases as of September 2006
Can we use networks to predict function

Network-based prediction of protein function
Roded Sharan, Igor Ulitsky & Ron Shamir
doi:10.1038/msb4100129
Systematically deduce the annotation of unknown nodes $u$ from the known (filled) nodes
“Direct” method for gene annotation

• K-nearest neighbors
  – assume that a node has the same function as its neighbors
Should $u$ and $v$ have the same annotation?
Advantages of kNN approach:
very easy to compute

Disadvantages:
how do you choose the best annotation?
“Direct”

Local search (Karaoz[2004]):

• For each annotation:
  – $S_v = 1$ if $v$ has the annotation, -1 otherwise
  – Procedure: for each unassigned node $u$, set $S_u$
    maximize $\sum S_u S_v$ for all edges $(u,v)$
  – iterate until convergence
Local search may not find some good solutions.
ΣS_uS_v does not improve if I only change A or C. Changing only B makes the score worse.
Can’t get there by a local optimization
How can we move away from a locally optimal solution?
Simulated Annealing Solution:

- Initialize $T$ and subgraph $G_n$ with score $S_n$.
- Repeat while
  - Pick a neighboring node $v$ to add to the subgraph.
  - Score new subgraph $\rightarrow S_{\text{test}}$.
  - If $S_n < S_{\text{test}}$: keep new subgraph.
  - Else keep new subgraph with
    \[ P = \exp\left[-\frac{(S_{\text{test}} - S_n)}{T}\right] \]
  - Modify $T$ according to “cooling schedule.”
Clustering Graphs

Goal: divide the graph into subgraphs each of which has lots of internal connections and few connections to the rest of the graph

Schaeffer Computer Science Review (2007) http://dx.doi.org/10.1016/j.cosrev.2007.05.001
Clustering Graphs

Two algorithms:
- edge betweenness
- markov clustering


Schaeffer Computer Science Review (2007) http://dx.doi.org/10.1016/j.cosrev.2007.05.001
Betweeness clustering

• Edge betweenness = number (or summed weight) of shortest paths between all pairs of vertices that pass through the edge.
  – Take a weighted average if there are >1 shortest paths for the same pair of nodes.

Betweeness clustering

- Repeat until max(betweenness) < threshold:
  - Compute betweenness
  - Remove edge with highest betweenness


Markov clustering (MCL)

• Goal: produce sharp partitions
• Intuition: A random walk will spend more time within a cluster than passing between clusters.

## Adjacency Matrix

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Adjacency Matrix

\[ A^N: a_{ij} = m \text{ iff there exist exactly } m \text{ paths of length } N \text{ between } i \text{ and } j. \]
MCL clustering

- Stochastic Matrix: each element $M_{ij}$ represents a probability of moving from $i$ to $j$ (this is a “Column Stochastic Matrix”).


MCL clustering

• Stochastic Matrix: each element $M_{ij}$ represents a probability of moving from $i$ to $j$ (this is a “Column Stochastic Matrix”).

• Therefore, $\sum_j p_{i,j} = 1$

• The probability of moving from $i$ to $j$ in two steps is given by

$$(M^2)_{i,j} = \sum_k p_{i,k} p_{k,j}$$


If we keep multiplying the stochastic matrix by itself, we compute the probabilities of longer and longer walks – we expect that the transitions will occur more frequently within a natural cluster than between them.

• This procedure won’t produce discrete clusters, so the algorithm includes an “inflation” step that exaggerates these effects: raise each element of the matrix to the power $r$ and renormalize.

$$p_A = 0.9$$

$$p_B = 0.1$$

$$p_A \rightarrow \frac{.81}{.81 + .01} = .99$$

$$p_B \rightarrow \frac{.01}{.81 + .01} = .01$$

$$(\Gamma_r M)_{pq} = (M_{pq})^r \div \sum_{i=1}^{k} (M_{iq})^r.$$
\( G \) is a graph
add loops to \( G \) \hspace{1cm} \# needed for a prob. of no transition
set \( \Gamma \) to some value \hspace{1cm} \# affects granularity
set \( \mathbf{M}_1 \) to be the matrix of random walks on \( G \)
while (change) {
\begin{align*}
\mathbf{M}_2 &= \mathbf{M}_1 \times \mathbf{M}_1 \hspace{1cm} \# \text{ expansion} \\
\mathbf{M}_1 &= \Gamma(\mathbf{M}_2) \hspace{1cm} \# \text{ inflation} \\
\text{change} &= \text{difference}(\mathbf{M}_1, \mathbf{M}_2)
\end{align*}
}
set CLUSTERING as the components of \( \mathbf{M}_1 \)
Example

- Identifying protein families
- BLAST will identify proteins with shared domains, but these might not be very similar otherwise (eg: SH2, SH3 domains)
Extremely fast, since it only requires matrix operations.

Copyright restrictions may apply.
Distinct clusters identified by MCL can still share a common domain
Example

• Clustering expression data for 61 mouse tissues
• Nodes = genes
• Edges = Pearson correlation coefficient > threshold
• Network gives an overview of connections not obvious from hierarchical clustering
Nodes=genes
Edges=pearson
Correlation of
Expression in
Mouse Tissues
Clustered by
MCL

Freeman, et
al. (2007) PLoS
Comput Biol
3(10): e206.
doi:10.1371/journal.pcbi.0030206

Courtesy of Freeman et al. License: CC-BY.
Largest clusters are gamete-specific

Cluster 4 = liver specific
Cluster 6 = kidney specific
Cluster 5 = both liver and kidney

Largest clusters are gamete-specific

How do we decide which function to assign to members of a cluster?
How do we decide which function to assign to members of a cluster?

- Consensus
- Significant by hypergeometric

Network Models

• Structure of network
  – Coexpression
  – Mutual information
  – Physical/genetic interactions

• Analysis of network
  – Ad hoc
  – Shortest path
  – Clustering
  – Optimization
How do we find modules associated with specific data?
Example: paint a PPI network with expression data. Try to find connected components that have overall high expression. (Example: Ideker et al. (2002) Bioinformatics).
Active subgraph problem:

Can reveal hidden components of a biological response.
Where did we see something similar?
• The annotation problem attempts to label the entire graph.
• The active subnet problem searches for a part of the graph that is enriched in a label.
• **Steiner Tree Problem:** Find the smallest tree connecting all the vertices of in a set of interest (terminals).

• **Downside:** will include all terminals, including false positives.
Experimental hits

Not all hits are real
Not all edges are real
Not all edges are known
Avoiding False Positives

Steiner tree is forced to include this node

- terminals
- no data
Network Models

- Structure of network
  - Coexpression
  - Mutual information
  - Physical/genetic interactions

- Analysis of network
  - Ad hoc
  - Shortest path
  - Clustering
  - Optimization
Prize Collecting Steiner Tree

- **Collect a prize** for each data point included

Don’t Include All Data

- Pay a **penalty** for excluding nodes

\[
\sum_{\nu \text{ not in } T} \beta \text{penalty}(\nu) + \sum_{e \text{ in } T} \text{cost}(e)
\]

Courtesy of Huang et al. Used with permission.
Source: Huang, Shao-shan Carol, David C. Clarke, et al. "Linking Proteomic and Transcriptional Data through the Interactome and Epigenome Reveals a Map of Oncogene-induced Signaling."

Avoid Unlikely Interactions

- Pay a cost for including edges based on probability

\[ \sum_{\nu \text{ not in } T} \beta \text{penalty}(\nu) + \sum_{e \text{ in } T} \text{cost}(e) \]
Balanced Objective Function

\[ \sum_{\nu \text{ not in } T} \beta \text{penalty}(\nu) + \sum_{e \text{ in } T} \text{cost}(e) \]

Does the node penalty justify the edge costs?

Courtesy of Huang et al. Used with permission.
Source: Huang, Shao-shan Carol, David C. Clarke, et al. "Linking Proteomic and Transcriptional Data through the Interactome and Epigenome Reveals a Map of Oncogene-induced Signaling."
Optimization methods:


\[ \sum_{v \text{ not in } T} \beta \text{penalty}(v) + \sum_{e \text{ in } T} \text{cost}(e) \]

Does the node penalty justify the edge costs?

![Diagram of a network with nodes labeled as phosphoproteins, TF, target gene, and no data.]

Courtesy of Huang et al. Used with permission.
Naïve Methods

- >2,500 nearest neighbors of phosphoproteins
- >4,500 nearest neighbors of phosphoproteins + transcription factors
Linking Proteomic and Transcriptional Data through the Interactome and Epigenome Reveals a Map of Oncogene-induced Signaling.

Can we find drug targets?

Rank every node by weighted distance to all prize-collecting Steiner tree nodes

High rank targets

Control targets

Courtesy of Huang et al. Used with permission.
Rank <27 out of 11,637

Lower Rank Targets 193 to 3,582 out of 11,637

Courtesy of Huang et al. Used with permission.
Data Integration
Approach

mRNA levels do not predict protein levels

1,000 fold range of protein concentrations


<table>
<thead>
<tr>
<th></th>
<th>SpectrumMill</th>
<th>msInspect</th>
<th>msBID</th>
<th>NSAF</th>
<th>RPKM</th>
<th>Microarray</th>
</tr>
</thead>
<tbody>
<tr>
<td>SpectrumMill</td>
<td>-</td>
<td>0.91 (0.92)</td>
<td>0.91 (0.91)</td>
<td>0.90 (0.90)</td>
<td>0.49 (0.51)</td>
<td>0.36 (0.40)</td>
</tr>
<tr>
<td>msInspect</td>
<td>0.91 (0.92)</td>
<td>-</td>
<td>0.89 (0.91)</td>
<td>0.87 (0.88)</td>
<td>0.51 (0.53)</td>
<td>0.40 (0.44)</td>
</tr>
<tr>
<td>msBID</td>
<td>0.91 (0.91)</td>
<td>0.89 (0.91)</td>
<td>-</td>
<td>0.84 (0.89)</td>
<td>0.54 (0.54)</td>
<td>0.41 (0.42)</td>
</tr>
<tr>
<td>NSAF</td>
<td>0.90 (0.90)</td>
<td>0.87 (0.88)</td>
<td>0.84 (0.89)</td>
<td>-</td>
<td>0.51 (0.53)</td>
<td>0.42 (0.44)</td>
</tr>
</tbody>
</table>


L18 Chromatin and DNase-seq Analysis

Mutant

Wild-type

Sequence Analysis

Logo

<table>
<thead>
<tr>
<th>Logo</th>
</tr>
</thead>
<tbody>
<tr>
<td>M0851</td>
</tr>
<tr>
<td>TGAC-TCA</td>
</tr>
<tr>
<td>M01262</td>
</tr>
<tr>
<td>TGAGTCA</td>
</tr>
<tr>
<td>M014001</td>
</tr>
<tr>
<td>ACAAAGC</td>
</tr>
<tr>
<td>M01214</td>
</tr>
<tr>
<td>TTTCCTG</td>
</tr>
</tbody>
</table>

Sequence Analysis
Move upstream of transcription

Network integration

Interactome

Epigenomic Data & Sequence Analysis

DNA-binding proteins

mRNA
‘Omic data don’t agree

Toxic Compound, Mutation, Environmental Change
Genetic vs. Expression Data

<table>
<thead>
<tr>
<th>Perturbation</th>
<th>Differentially expressed genes</th>
<th>Genetic hits</th>
<th>Number of overlapping genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Growth arrest (Hydroxyurea)</td>
<td>59</td>
<td>86</td>
<td>0</td>
</tr>
<tr>
<td>DNA damage (MMS)</td>
<td>198</td>
<td>1448</td>
<td>43</td>
</tr>
<tr>
<td>Protein biosynthesis block</td>
<td>20</td>
<td>164</td>
<td>0</td>
</tr>
<tr>
<td>(Cycloheximide)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ER stress (Tunicamycin)</td>
<td>200</td>
<td>127</td>
<td>5</td>
</tr>
<tr>
<td>ATP synthesis block (Arsenic)</td>
<td>828</td>
<td>50</td>
<td>9</td>
</tr>
<tr>
<td>Fatty acid metabolism</td>
<td>269</td>
<td>103</td>
<td>9</td>
</tr>
<tr>
<td>(oleate)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gene inactivation</td>
<td>27</td>
<td>130</td>
<td>0</td>
</tr>
<tr>
<td>(24 datasets, median shown)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Bridging high-throughput genetic and transcriptional data reveals cellular responses to alpha-synuclein toxicity

Nature Genetics Published online: 22 February 2009
For 156 perturbations:

Genetic Data Enriched for:
• Transcriptional regulation
• Signal transduction

Expression Data Enriched for:
Metabolic Processes
e.g., organic acid metabolic process, oxidoreductase activities

Bridging high-throughput genetic and transcriptional data reveals cellular responses to alpha-synuclein toxicity
Nature Genetics Published online: 22 February 2009
Bridging high-throughput genetic and transcriptional data reveals cellular responses to alpha-synuclein toxicity

Nature Genetics Published online: 22 February 2009
DNA Damage

RAD53 = CHK2
MEC1 = ATM

Cell cycle arrest
DNA repair

Bridging high-throughput genetic and transcriptional data reveals cellular responses to alpha-synuclein toxicity
Nature Genetics Published online: 22 February 2009
Bridging high-throughput genetic and transcriptional data reveals cellular responses to alpha-synuclein toxicity
Nature Genetics Published online: 22 February 2009
Test case: Perturbing pheromone response pathway

Perturbing Ste5

20 genes rescue mating phenotype (SGD)

12 genes differentially expressed (Rosetta compendium)
Δste5: Naïve approach
Paths limited to length 3

Genetic Data

Expression Data

193 nodes, 778 edges

Bridging high-throughput genetic and transcriptional data reveals cellular responses to alpha-synuclein toxicity

Nature Genetics Published online: 22 February 2009
Maximize the connectivity via reliable paths

Goal: find paths that maximize product of $P_{ij}$

Assign probabilities using a Bayesian approach based on reliability of underlying data type:


Bridging high-throughput genetic and transcriptional data reveals cellular responses to alpha-synuclein toxicity
Nature Genetics Published online: 22 February 2009
Maximize the connectivity via reliable paths

Source

Minimum cost flow problem

Sink

Flow

Low probability

High probability

Bridging high-throughput genetic and transcriptional data reveals cellular responses to alpha-synuclein toxicity

Nature Genetics Published online: 22 February 2009
Maximize the connectivity via reliable paths

Minimum cost flow problem

Flow

Proteins ranked by their incoming flow:

Less important  More important
Maximize the connectivity via reliable paths

Minimum cost flow problem

Maximize flow: source to sink

Minimize cost (e_{ij}) = f_{ij} \cdot (-\log P_{ij})

\[
\min (\sum \text{cost}(e_{ij}) - \gamma \sum f_{Sj})
\]

f_{ij} = \text{flow through } e_{ij}

c_{ij} = \text{capacity of } e_{ij} = 1 \text{ for all } e_{ij}

Proteins ranked by their incoming flow:

Less important

More important
Test case: Perturbing pheromone response pathway

Perturbing Ste5

20 genes rescue mating phenotype (*SGD*)

12 genes differentially expressed
(Rosetta compendium)

Bridging high-throughput genetic and transcriptional data reveals cellular responses to alpha-synuclein toxicity
*Nature Genetics* Published online: 22 February 2009
Enriched for pheromone response $p<10^{-18}$

49 nodes, 96 edges

Predicted genes

Importance
Network Models

• Structure of network
  - Coexpression
  - Mutual information
  - Physical/genetic interactions

• Analysis of network
  - Ad hoc
  - Shortest path
  - Clustering
  - Optimization
Physical Relationships

Known Components
- Differential equations
- Boolean logic, decision trees
- Bayesian networks

Statistical Relationships

Unknown Components
- Interactome Models
- Mutual information
- Regression, clustering