Large-Scale Data Fusion by Collective Matrix Factorization

Marinka Žitnik & Blaž Zupan
University of Ljubljana, Slovenia

Tutorial at BC^2, Basel 2015
Tutorial Overview

❖ **Motivation:** search for bacterial-response genes, a case study
❖ **Warm up:** recommender systems
❖ **Data fusion:** tri-factorization and sharing of latent features
❖ **Examples:** movies & genes
❖ **Hands-on:** visual programming and scripting
❖ **Applications:** case studies in prioritization and classification
❖ **Other approaches:** related work
Motivation

Search of bacterial response genes in a social amoeba *Dictyostelium discoideum.*
Motivation: Bacterial-Response Genes in Dicty

Gad Shaulsky

Adam Kuspa

Baylor College of Medicine, Houston, USA

Dictyostelium discoideum
Search for Bacterial Response Genes

Dicty is bacterial predator!

50,000 clonal mutants

genetic screen

gene pool

12,000 genes

found

7 genes

workload

5 years

estimated

~200 genes

Gram+ defective: swp1, gpi, nagB1

Gram- defective: clkB, spc3, alyL, nip7
Now What?

50% coverage (100 genes) → 20 screens required!

80% coverage (160 genes) → 65 screens required!
### Alternative: A Data-Driven Approach

<table>
<thead>
<tr>
<th>Genes</th>
<th>Mutant Phenotypes</th>
</tr>
</thead>
<tbody>
<tr>
<td>spc3</td>
<td>Gram neg. defective</td>
</tr>
<tr>
<td>swp1</td>
<td>Aberrant spore color</td>
</tr>
<tr>
<td>kif9</td>
<td>Aberrant spore color</td>
</tr>
<tr>
<td>alyL</td>
<td>Decreased chemotaxis</td>
</tr>
<tr>
<td>nagB1</td>
<td>Gram pos. defective</td>
</tr>
<tr>
<td>gpi</td>
<td></td>
</tr>
<tr>
<td>shkA</td>
<td></td>
</tr>
<tr>
<td>nip7</td>
<td></td>
</tr>
</tbody>
</table>
So Much More Data …
Actually, in Biomedicine, Matrices Abound ...
Actually, in Biomedicine, Matrices Abound …

Gene Interactions
Actually, in Biomedicine, Matrices Abound …

Part of N-Glycan biosynthesis pathway

Fructose and mannose metabolism

GPI-anchor biosynthesis

Gene Pathways
Actually, in Biomedicine, Matrices Abound …

```
<table>
<thead>
<tr>
<th>Response to stress</th>
<th>Response to external stimulus</th>
<th>Response to biotic stimulus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defense response</td>
<td>Response to other organisms</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Defense response to other organism</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Defense response to bacterium</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Gene Ontology terms

```
<table>
<thead>
<tr>
<th>Response to stress</th>
<th>Response to other organisms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response to bacterium</td>
<td>Defense response</td>
</tr>
</tbody>
</table>
```

Ontologies
Actually, in Biomedicine, Matrices Abound ...

Ontologies of Controlled Vocabularies
Actually, in Biomedicine, Matrices Abound ...

Part of N-Glycan biosynthesis pathway

Orthology
Dolichol kinase (K00902)
Alpha-mannosidase II (K01231)
Oligosaccharyltransferase complex (K12668)

Ontology
GO:0004168
GO:0004572
GO:0008250

Cross-Links of Controlled Entities
BC2 Tutorial on
Large-scale data fusion by collective matrix factorization

Warm-Up

Working with matrices: recommender systems, two-factorization and tri-factorization.
Back to the Problem: Recommender Systems

Mutant Phenotypes
- spc3
- swp1
- kif9
- alyL
- nagB1
- gpi
- shkA
- nip7

Genes
- Gram negative defective
- Aberrant spore color
- Aberrant spore color
- Decreased chemotaxis
- Gram positive defective

Users
- John
- Kate
- Alex
- Jerry
- Jenny
- Mike
- Morgan
- Nick

Movies
- Passengers
- War of the Worlds
- Bride Wars
- The Matrix Reloaded
- The Godfather
- The Dark Knight
- Pulp Fiction
- Schindler's List

Diagram showing the relationships between genes, phenotypes, users, and movies in a recommender system context.
Recommender Systems
Netflix Prize, 2009
**Example: A Small Relation Matrix**

<table>
<thead>
<tr>
<th></th>
<th>Passengers</th>
<th>War of the Worlds</th>
<th>Bride Wars</th>
<th>The Matrix Reloaded</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kate</td>
<td>5</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alex</td>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mike</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Matrix Two-Factorization

<table>
<thead>
<tr>
<th></th>
<th>L1</th>
<th>L2</th>
<th>L1</th>
<th>L2</th>
<th>L1</th>
<th>L2</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>0.2</td>
<td>0</td>
<td>6.3</td>
<td>0</td>
<td>3.9</td>
<td>10.7</td>
</tr>
<tr>
<td>Kate</td>
<td>0</td>
<td>0.5</td>
<td>0</td>
<td>8</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Alex</td>
<td>0.5</td>
<td>0</td>
<td>1.1</td>
<td>0</td>
<td>3.3</td>
<td>3.3</td>
</tr>
<tr>
<td>Mike</td>
<td>0</td>
<td>0.5</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Matrix Two-Factorization

<table>
<thead>
<tr>
<th></th>
<th>Passengers</th>
<th>War of the Worlds</th>
<th>Bride Wars</th>
<th>The Matrix Reloaded</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>1.3</td>
</tr>
<tr>
<td>Kate</td>
<td>5</td>
<td>4</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Alex</td>
<td>4</td>
<td>5</td>
<td>0</td>
<td>3.2</td>
</tr>
<tr>
<td>Mike</td>
<td>4</td>
<td>5</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

≈

<table>
<thead>
<tr>
<th></th>
<th>Passengers</th>
<th>War of the Worlds</th>
<th>Bride Wars</th>
<th>The Matrix Reloaded</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>1.3</td>
<td>0</td>
<td>0.2</td>
<td>1.6</td>
</tr>
<tr>
<td>Kate</td>
<td>2</td>
<td>5.4</td>
<td>0</td>
<td>1.7</td>
</tr>
<tr>
<td>Alex</td>
<td>3.2</td>
<td>0</td>
<td>0.6</td>
<td>4</td>
</tr>
<tr>
<td>Mike</td>
<td>2</td>
<td>5.4</td>
<td>0</td>
<td>1.7</td>
</tr>
</tbody>
</table>
# Matrix Tri-Factorization

<table>
<thead>
<tr>
<th></th>
<th>U1</th>
<th>U2</th>
<th>M1</th>
<th>M2</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>0.2</td>
<td>0.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kate</td>
<td>0.8</td>
<td>0.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alex</td>
<td>0.7</td>
<td>1.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mike</td>
<td>0.8</td>
<td>1.2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>U1</th>
<th>U2</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>-4.4</td>
<td>9.1</td>
</tr>
<tr>
<td>U2</td>
<td>6.7</td>
<td>-5.8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>M1</th>
<th>M2</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>0.9</td>
<td>0.2</td>
</tr>
<tr>
<td>M2</td>
<td>0.6</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Note: The values represent preferences or ratings for each user and movie.
Matrix Tri-Factorization

<table>
<thead>
<tr>
<th></th>
<th>Passengers</th>
<th>War of the Worlds</th>
<th>Bride Wars</th>
<th>The Matrix Reloaded</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>2</td>
<td></td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Kate</td>
<td>5</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alex</td>
<td>4</td>
<td></td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Mike</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ \begin{pmatrix} 1.1 & 0.3 & 0.2 & 1.2 \\ 1.7 & 4.5 & 0.2 & 2.1 \\ 4.1 & 0.5 & 0.9 & 4.5 \\ 1.5 & 4.8 & 0.1 & 1.9 \end{pmatrix} \approx \begin{pmatrix} 2 & 3 & 4 \end{pmatrix} \begin{pmatrix} 4 & 5 \end{pmatrix} \begin{pmatrix} 0.2 & 0.3 & 0.4 & 0.5 \end{pmatrix} \]
Hands-On: Matrix Tri-Factorization
BC2 Tutorial on
Large-scale data fusion by collective matrix factorization

Data Fusion

Collective matrix tri-factorization.
Latent factor sharing.
Data Fusion Graph: One Data Matrix

Tri-factorization of matrix A-B

Recipe matrix of A

Backbone matrix of A-B

Recipe matrix of B

Reconstructed matrix A-B

\[ A \times B \times \approx A-B \]
Data Fusion Graph: Two Data Matrices

Collective tri-factorization of matrices A-B and A-E

Recipe matrix of A
Recipe matrix of B
Backbone matrix of A-E

Reconstructed matrix A-B
Reconstructed matrix A-E
Data Fusion Graph: All Together Now
Sharing of Latent Matrices
\[ \mathbf{R} = \begin{bmatrix} * & \mathbf{R}_{12} & \ldots & \mathbf{R}_{1r} \\
\mathbf{R}_{21} & * & \ldots & \mathbf{R}_{2r} \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{R}_{r1} & \mathbf{R}_{r2} & \ldots & * \end{bmatrix} \quad \mathbf{S} = \begin{bmatrix} * & \mathbf{S}_{k_1 \times k_2} & \ldots & \mathbf{S}_{k_1 \times k_r} \\
\mathbf{S}_{k_2 \times k_1} & * & \ldots & \mathbf{S}_{k_2 \times k_r} \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{S}_{k_r \times k_1} & \mathbf{S}_{k_r \times k_2} & \ldots & * \end{bmatrix} \quad \Theta^{(t)} = \text{Diag}(\Theta_1^{(t)}, \Theta_2^{(t)}, \ldots, \Theta_r^{(t)}) \]

\[ \min_{\mathbf{G} \geq 0} J(\mathbf{G}; \mathbf{S}) = \sum_{\mathbf{R}_{ij} \in \mathcal{R}} \| \mathbf{R}_{ij} - \mathbf{G}_i \mathbf{S}_{ij} \mathbf{G}_j^T \|^2 + \sum_{t=1}^{\max_i t_i} \text{tr}(\mathbf{G}_t^T \Theta^{(t)} \mathbf{G}_t) \]
Theorem 2 (Convergence of DFMF algorithm): Due to nonnegativity of auxiliary function \( \Phi \), we have seen that using such a rule, at convergence, the updates of all matrices on right-hand sides converge to a local optimum which is equivalent to using Eq. (10)–(12) converge to a local optimum which is equivalent to

\[
\begin{align*}
G_i^{(e)} &= (R_{ij} G_j S_{ij})^+ + G_i (S_{ij} G_j S_{ij})^-
G_i^{(d)} &= (R_{ij} G_j S_{ij})^- + G_i (S_{ij} G_j S_{ij})^+
G_j^{(e)} &= (R_{ij} G_i S_{ij})^+ + G_j (S_{ij} G_i S_{ij})^-
G_j^{(d)} &= (R_{ij} G_i S_{ij})^- + G_j (S_{ij} G_i S_{ij})^+
\end{align*}
\]

(10)

For \( t = 1, 2, \ldots, \max_i t_i \):

\[
\begin{align*}
G_i^{(e)} &= [\Theta_i^{(t)}]^- G_i \quad \text{for } i = 1, 2, \ldots, r \\
G_i^{(d)} &= [\Theta_i^{(t)}]^- G_i \quad \text{for } i = 1, 2, \ldots, r 
\end{align*}
\]

(11)

Construct \( \mathbf{G} \) as:

\[
\mathbf{G} \leftarrow \mathbf{G} \odot \text{Diag}(\sqrt{G_1^{(e)}}, \sqrt{G_2^{(e)}}, \ldots, \sqrt{G_r^{(e)}}), \sqrt{G_1^{(d)}}, \sqrt{G_2^{(d)}}, \ldots, \sqrt{G_r^{(d)}}), \]

(12)

where \( \odot \) denotes the Hadamard product. The \( \sqrt{\cdot} \) and \( \cdot^{-} \) are entry-wise operations.
Theorem 1 (Correctness of the algorithm): If the update rules for matrix factors G and S from the Algorithm converge, then the final solution satisfies the KKT conditions of optimality.

Theorem 2 (Convergence of the algorithm): The objective function $J(G; S)$ is nonincreasing under the updating rules for matrix factors G and S given by the Algorithm.
Hands-On: Collective Matrix Factorization
BC2 Tutorial on
Large-scale data fusion by collective matrix factorization

Scoring

Data sampling.
Completion scoring.
Data Sampling & Completion

\[
\text{RMSE}(y, \hat{y}) = \sqrt{\frac{\sum_i (y_i - \hat{y}_i)^2}{n}}
\]
Hands-On: Performance Evaluation
Hands-On: Latent Profiling
BC2 Tutorial on
Large-scale data fusion by collective matrix factorization

The Yeast Case Study

A large data compendium. Functional genomics.
The Yeast Case Study
Hands-On: The Yeast Case Study
Latent Matrix Chaining

Gene literature → Literature → Topics of yeast biology → Literature Topic

Gene → Literature topic

Gene profile matrix = Gene literature × Literature matrix × Topics of yeast biology matrix = Gene profile matrix
Hands-On: Latent Matrix Chaining

Latent chains

1165x166  Gene  →  Ontology term
1165x166  Gene  →  Literature  →  Ontology term

Complete chain to:
- latent space
- feature space
BC2 Tutorial on
Large-scale data fusion by collective matrix factorization

Case Studies

Dictyostelium bacterial gene hunt.
Functional genomics.
Drug-induced liver injury.
Survival analysis.
Dictyostelium Bacterial Gene Hunt

4 seed genes

14 data sets, only one of which was directly related to bacterial response
Gene Hunt: Object Profiling by Latent Chaining

Latent chains

\[ G_1, G_1S_{1,7}, G_1S_{1,8}, G_1S_{1,9}, G_1S_{1,10}, G_1S_{1,2}, G_1S_{1,6}, G_1S_{1,5}, G_1S_{1,4}, G_1S_{1,2}S_{2,3}, G_1S_{1,6}S_{6,5}, G_1S_{1,6}S_{6,4}, G_1S_{1,2}S_{2,4}, G_1S_{1,5}S_{5,4} \] and \( G_1S_{1,6}S_{6,5}S_{5,4} \).
Gene Hunt: Similarity Estimation and Ranking

Similarity estimation
- Seed genes
- Candidate gene

Gene ranking
- Candidate genes
- Scored candidate genes

Similarity scoring
- Similarity score matrices
- Similarity score aggregation

Seed genes
- Similarity score matrix
- Scored candidate gene

Chains
Gene Hunt: Validation of Predictions

The impact of data selection. The table lists the top 30 candidate genes obtained by prioritization by data fusion of 14, 7, 4, 3 and 2 data sets from the data fusion graphs in Supplementary Fig. 3. Genes in bold are the ones selected for the experimental study.

<table>
<thead>
<tr>
<th>Gene</th>
<th>Day 2</th>
<th>Day 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>cf50-1</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>smlA</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>acbA</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>pirA</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>rps10</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>abpC</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>tirA</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>pikB</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>vps46</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>pikA</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>swp1</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>ggtA</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>DDB_G0272184</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pten</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>DDB_G0288519</td>
<td></td>
<td></td>
</tr>
<tr>
<td>abpC</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>DDB_G0288519</td>
<td></td>
<td></td>
</tr>
<tr>
<td>modA</td>
<td>AX4</td>
<td>AX4</td>
</tr>
<tr>
<td>DDB_G0287399</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tirA</td>
<td>AX4</td>
<td>AX4</td>
</tr>
</tbody>
</table>
### Functional Genomics

The image contains a network diagram with nodes labeled as Gene, PMID, GO Term, KEGG Pathway, MeSH Descriptor, Experimental Condition, Pharmacologic Action, Chemical, Depositor, and Substructure Fingerprint. The diagram illustrates the relationships between these nodes through various edges with associated expressions.

#### Table 1: Predictive Performance

<table>
<thead>
<tr>
<th>Prediction task</th>
<th>DFMF</th>
<th>MKL</th>
<th>RF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$F_1$</td>
<td>AUC</td>
<td>$F_1$</td>
</tr>
<tr>
<td>100 <em>D. discoideum</em> genes</td>
<td>0.799</td>
<td>0.801</td>
<td>0.781</td>
</tr>
<tr>
<td>1000 <em>D. discoideum</em> genes</td>
<td>0.826</td>
<td>0.823</td>
<td>0.787</td>
</tr>
<tr>
<td>Whole <em>D. discoideum</em> genome</td>
<td>0.831</td>
<td>0.849</td>
<td>0.800</td>
</tr>
<tr>
<td>Pharmacologic actions</td>
<td>0.663</td>
<td>0.834</td>
<td>0.639</td>
</tr>
</tbody>
</table>
Drug Toxicity Prediction

Table 3. Predictive performance of fusing various subsets of assays for DILI potential prediction.

<table>
<thead>
<tr>
<th>Data fusion studies</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>In vivo studies</td>
<td>0.819</td>
</tr>
<tr>
<td>In vitro studies</td>
<td>0.790</td>
</tr>
<tr>
<td>Human in vitro study</td>
<td>0.793</td>
</tr>
<tr>
<td>Animal in vitro study</td>
<td>0.799</td>
</tr>
<tr>
<td>Animal studies</td>
<td>0.811</td>
</tr>
<tr>
<td>Human studies</td>
<td>0.792</td>
</tr>
<tr>
<td>All studies</td>
<td>0.810</td>
</tr>
</tbody>
</table>
Survival Analysis
Survival Analysis
Disease-Disease Association Discovery

Example of an identified disease class

- **Interstitial nephritis**
  - **Crescentic glomerulonephritis**
  - **Acute proliferative glomerulonephritis**
  - **Nephritis**
  - **Urinary system disease**
  - **Kidney failure**
  - **Kidney disease**

**Sources and relations**
- **DO terms**: nodes represent four types of objects, i.e., genes, GO terms, DO terms, and drugs.
- **Arcs**: denote data sources that relate objects of different types (relation matrices).

**Table 2**

<table>
<thead>
<tr>
<th>Data source</th>
<th>Size</th>
<th>Edge density</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapped GeneRIF</td>
<td>5,267/1,536</td>
<td>2%</td>
</tr>
<tr>
<td>DrugBank v3.0</td>
<td>4,477/2,182</td>
<td>6%</td>
</tr>
<tr>
<td>Gene Ontology</td>
<td>10,360</td>
<td>9%</td>
</tr>
<tr>
<td>KEGG</td>
<td>12,050</td>
<td>9%</td>
</tr>
<tr>
<td>Prieto et al.</td>
<td>17,852</td>
<td>6%</td>
</tr>
<tr>
<td>BioGRID v3.1.94</td>
<td>21,821</td>
<td>6%</td>
</tr>
<tr>
<td>Managed GeneRIF</td>
<td>22,084</td>
<td>8%</td>
</tr>
</tbody>
</table>

Relative contribution of each data source to the fused model:

1. Gene-disease relationships: 15% increase
2. GO semantic structure: 2% decrease
3. Drug interaction data: 9% decrease
4. Cell signalling data: 9% decrease
5. Gene co-expression: 100% increase

**Mechanisms**

- From the configuration given, the quality of the resulting fused model drops by 0.00273.
- The column labelled "R" indicates that only drug side-effects information was removed.
Disease-Disease Association Discovery

Root layer

Level 1
- cancer
- inherited metabolic disorders
- nervous system diseases
- respiratory system diseases
- cardiovascular system disease

Level 2
- immune system diseases
- cognitive disorders
- acquired metabolic diseases
- metabolic diseases
- cancer

Level 3

Disease class size:
- a single disease
- two diseases
- three or more diseases
- eighteen diseases
Related Approaches

Network-based methods.
Kernel-based methods.
Probabilistic graphical models.
Collective latent factor models.
Data Integration Strategies

Early integration

Intermediate integration

Late integration
**Network-Based Approaches**

Figure 3. A graphical example that depicts the network integration algorithm used in GeneMANIA. In this example, we are assuming that we have a hypothetical organism with five genes (the nodes are labeled as a, b, c, d, and e). Given a query list with three genes (nodes a, b, and c), we would like to assign weights to interaction networks I and II in order to construct a weighted combination of the two networks.

We refer to a link between nodes a and b as \((a, b)\). To assign network weights, we first construct an ideal network whereby pairs of query genes are connected together by a positively weighted link and pairs of query and nonquery genes are connected together by a negatively weighted link. We then construct the vector \(\vec{t}\) whereby each element represents a link in the ideal network. Similarly, we represent networks I and II by vectors \(\vec{x}_1\) and \(\vec{x}_2\), where we only consider links between query–query genes and query to nonquery genes. Finally, to find the network weights, we attempt to reconstruct \(\vec{t}\) from a weighted combination of an intercept term \((\alpha_0)\), \(\vec{x}_1\), and \(\vec{x}_2\).

Appending a vector of all 1's to \(X\) \((\vec{x}_0 = \vec{1})\) and discards this intercept term when constructing the composite network. Here, \(\vec{t}\) is the vectorized representation of the ideal network. Note that this cost function does not include a link between pairs of nonquery genes in \(X\) or \(t\); this is because links between pairs of nonquery genes have lesser importance when assessing network relevance to the query list.

4.3.3 GO-dependent weighting

When the queried list of gene is very small (e.g. less than five), the query-dependent weighting scheme can overfit and inaccurately assess weights to the interaction networks. In the GO-dependent network weighting (described in [48]), GeneMANIA constructs a combined network where the relevance of each network is predetermined based on a large number of GO functions. In this approach, GeneMANIA solves a similar regression problem whereby network weights are simultaneously optimized for predicting multiple GO functions using the cost function:

\[ \sum c (\vec{t}_c - \vec{x}_1 / H9251) \Rightarrow \vec{x}_2 \],

where \(\vec{t}_c\) is the vectorized representation of an ideal network (as per the query-dependent weighting above) with all genes annotation with GO term \(c\) used as the query list. The summation in this equation is over a predefined set of GO terms.

In the GeneMANIA webserver and in the Cytoscape plugin, the user has the option to choose between three predefined sets of GO terms: those from the BP, CC, or MF hierarchies. These sets only contain GO terms that have between 3 and 300 annotations.

4.4 GeneMANIA label propagation

Formally, we are given a composite network \(W^*\), and a vector of labels \(\vec{y} = \{0, 1\}^n\), where the query genes (also referred to as the positive genes) are represented by +1, and all other genes (unlabeled) are represented by 0, the goal is to score the nonquery genes according to their likelihood of being positives. GeneMANIA's output is a vector of discriminant scores \(\vec{f} = \{0, 1\}^n\), where a high value of \(f_i\) indicates a higher confidence.
Network-Based Approaches

(a) Original data
(b) Patient similarity matrices

DNA methylation

(c) Patient similarity networks

(d) Fusion iterations

(e) Fused patient similarity network

Wang et al., Nat Methods 2014; Greene et al., Nat Genet 2015.
Multiple Kernel Learning

\[
\text{maximize } \omega(K) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j k(x_i, x_j)
\]

with respect to \( \alpha \in \mathbb{R}_+^N \)

subject to \( \sum_{i=1}^{N} \alpha_i y_i = 0 \)

\( C \geq \alpha_i \geq 0 \quad \forall i. \)

\[\mathcal{K}_L = \left\{ K : K = \sum_{m=1}^{P} \eta_m K_m, \; K \succeq 0, \; \text{tr}(K) \leq c \right\}\]

Lanckriet et al., Bioinformatics 2004; Yu et al., BMC Bioinformatics 2010; Gonen et al., JMLR 2012.
Probabilistic Graphical Models

Bayesian integration of heterogeneous data

This component uses a Bayesian network to integrate diverse data to derive a probabilistic linkage map among proteins.

Functional genomic input data

We have collected a diverse set of evidence from over 950 publications from several databases, including complete physical and genetic interaction data from the GRID and BIND databases (downloaded on 6/25/04), which contain both high-throughput interaction data sets and some interactions from individual experiments curated from the literature [35,40,41]. We also make use of cellular localization data [42], curated sequence data in the form of shared transcription factor binding sites from the Saccharomyces cerevisiae Promoter Database (SCPD) [43], and biological complex curated literature from the Saccharomyces Genome Database (SGD) [35]. Additionally, we have collected gene expression data from 10 different microarray studies, totaling more than 300 arrays and 29 distinct biological conditions [31,33,34,44-50]. Pearson correlation between genes across each set of related conditions is used as a measure of similarity. Correlation coefficients in each dataset are converted to Z-scores and combined across datasets. References to all sources of genomic data are listed in [51].

Bayesian network structure and conditional probabilities

Given these diverse data, we can answer questions about pairwise protein relationships using a Bayesian network that leverages our previous work [2]. A Bayesian network essentially weights each evidence type according to a measure of confidence. This Bayesian integration produces a graph with confidence-weighted relationships between each gene pair (characterized in supplemental Figure S1 in [15]). Based on this integrated network graph and a user-defined query set of proteins of interest (b), the network prediction algorithm identifies novel network components by finding proteins with the maximum expected number of direct and indirect relationships with the query set (c). The resulting network is then displayed to the user using a spring model layout, such that the geometric proximity of genes reflects how related they are to each other, and the edge color reflects the confidence of pair-wise connections (d). Details of each component are presented in Materials and methods.

Collective Latent Factor models

**Relation heterogeneity** — borrow consistent patterns across many potentially heterogeneous input spaces.

**Object type heterogeneity** — leverage heterogeneous types of features to improve the learning performance in each task.

**Task heterogeneity** — exploit related prediction tasks to transfer knowledge between data views.
BC2 Tutorial on
Large-scale data fusion by collective matrix factorization

Conclusion

Current challenges.
Universal data fusion.
Multiple Types of Data Heterogeneity

Multislice Data

Multiscale Data


Multiplex & Multirelational Data
Data Fusion of Everything!
Thanks to Bioinformatics Laboratory

University of Ljubljana
Faculty of Computer and Information Science