Assessing a Nonlinear Dimensionality Reduction-Based Approach to Biological Network Reconstruction.

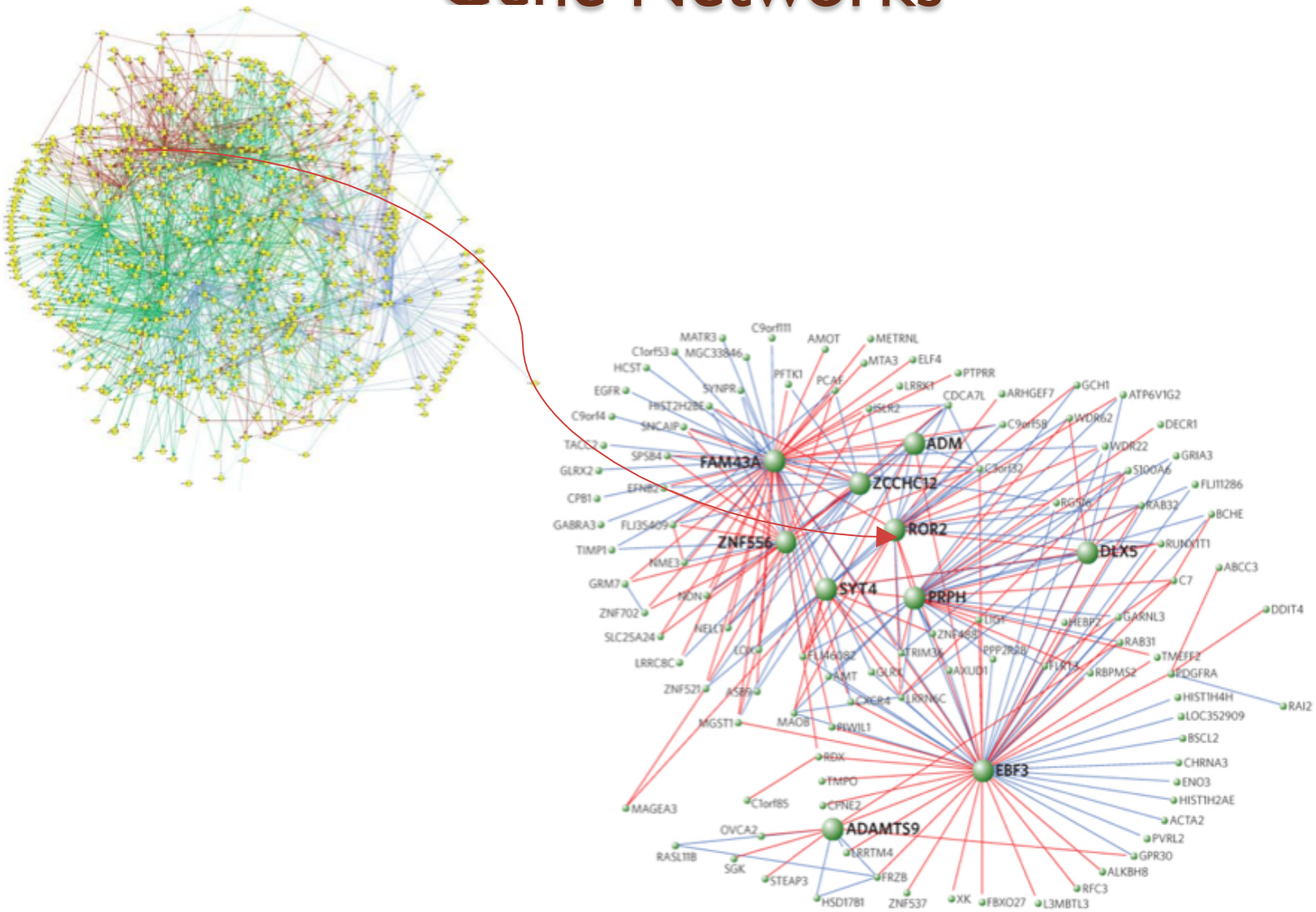
Vinodh N. Rajapakse (vinodh@math.umd.edu)

Advisor: Prof. Wojciech Czaja (wojtek@math.umd.edu)

Presentation Outline

- Problem Review
- Solution Approach
- Accomplished Work
- Main Results
- Future Work
- Questions and Comments

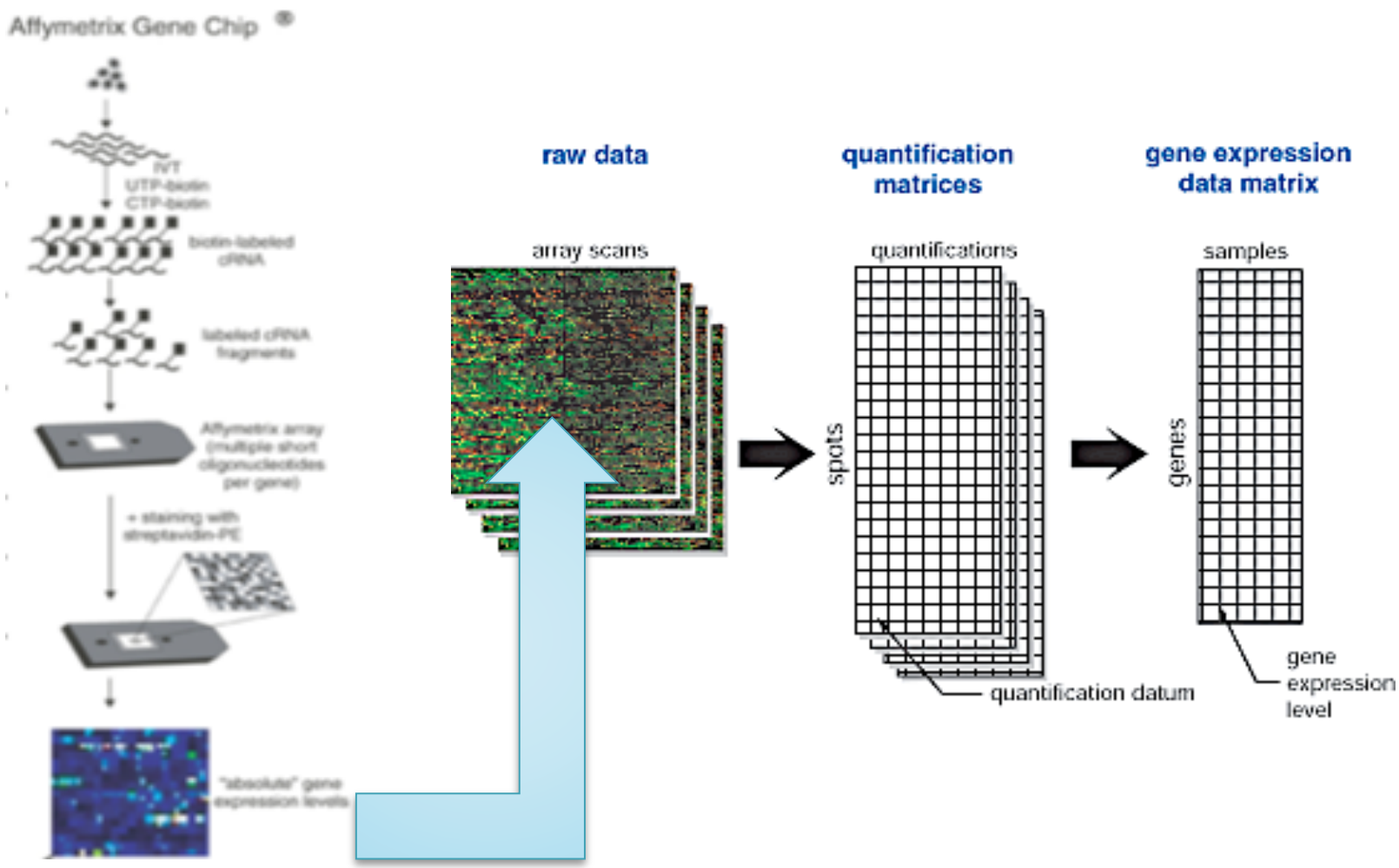
Gene Networks



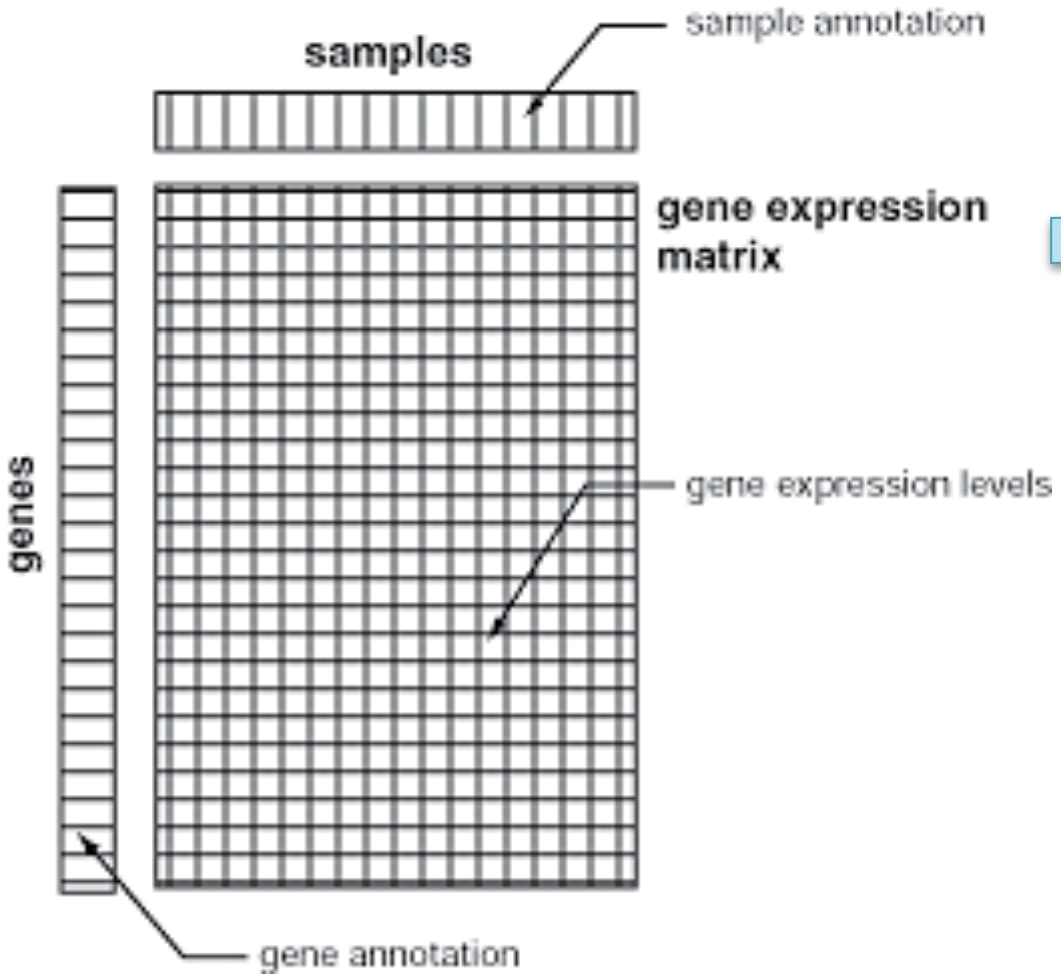
Motivation

- Gain a broader, systems level view of biological processes and their underlying functional elements
  - Avoid a narrow focus on a limited subset of driving elements
  - Incisively identify the most promising targets for experimental exploration (to derive focused data for iteratively refining models)

Gene Expression Microarrays



Starting Point: Gene Expression Data Matrix



$\mathbf{X}_{(N \times D)}$

N = 12,600
D = 336

Basic Network Construction Workflow

- Starting from from the N x D gene expression data matrix **X**, derive an N x d matrix **Y**, (d < D) using Laplacian Eigenmaps (or another dimensionality reduction technique).
- Construct an N x N matrix **W*** capturing pairwise Euclidean distances between row vectors of **Y** ('reduced gene profiles').
- Apply a threshold to the elements of **W*** to obtain a network (adjacency matrix) representation.

Laplacian Eigenmaps

- Input: **X** (N x D) → Output: **Y** (N x d)
  - Let **x** = ($x_1, x_2, \ldots, x_D$) denote a row of **X**
  - Let **y** = ($y_1, y_2, \ldots, y_d$) denote a row of **Y**
- Step 1: Model Data Point Relationships
  - Build a graph G, with nodes i and j connected if $\mathbf{x}_i$ is one of the k nearest neighbors of $\mathbf{x}_j$ or vice versa (Euclidean distances used, alternatives are possible)
  - k is a local structure resolution parameter

Laplacian Eigenmaps

- Step 2: Form Weight Matrix
  - Form a diffusion weight matrix W, with entry $W_{i,j} = \exp\{ - \| x_i - x_j \|^2 / \sigma \}$, if i and j are connected; 0 otherwise.
- Step 3: Solve Minimization Problem

$$\min_{(Y^T DY = I)} \frac{1}{2} \sum_{i,j} \| y_i - y_j \|^2 W_{i,j}$$

Laplacian Eigenmaps

- Step 3 (cont.) Solve Eigenvalue Problem
  - Given weight matrix **W**, let **D** be a N x N diagonal ('connectivity') matrix with entries recording the sum of edge weights for each data point-derived node
  - Let **L** = **D** – **W** denote the Laplacian matrix
  - We have:

$$\min_{(Y^T DY=I)} \frac{1}{2} \sum_{i,j} \| y_i - y_j \|^2 W_{i,j} = \min_{(Y^T DY=I)} \operatorname{trace}(Y^T LY),$$

Laplacian Eigenmaps

- Given: $\min_{(Y^T DY=I)} \frac{1}{2}\sum_{i,j} \| y_i - y_j \|^2 W_{i,j} = \min_{(Y^T DY=I)} \mathrm{trace}(Y^T LY),$
- Basic results from linear algebra show that the optimal mapping can be obtained by solving generalized eigenvalue problem **Lx** = λ **Dx**, under the above constraint.
- In particular, coordinates for the mapped vector y_i can be extracted from the ith coordinates of the d eigenvectors with smallest nonzero eigenvalues.

Laplacian Eigenmaps

- Additional Details:
  - $Lx = \lambda Dx \Leftrightarrow (D^{-1/2} L D^{-1/2})u = \lambda u$, where $u = D^{1/2}x$, $(D^{-1/2} L D^{-1/2}) = L_{sym}$
  - Estimation of intrinsic data dimensionality d
  - Selection of local neighborhood resolution parameter k
  - Selection of kernel width parameter σ

Basic Network Construction Workflow

- Starting from from the N x D gene expression data matrix **X**, derive an N x d matrix **Y**, (d < D) using Laplacian Eigenmaps (or another dimensionality reduction technique).
- Construct an N x N matrix **W*** capturing pairwise Euclidean distances between row vectors of **Y** ('reduced gene profiles').
- Apply a threshold to the elements of **W*** to obtain a network (adjacency matrix) representation.

Network Derivation

- Given:
  - Target (fractional) value α
  - N x N matrix **W*** capturing pairwise Euclidean distances between mapped gene expression profiles in reduced dimensional space.
- Estimate: Distance Threshold
  - Rank mapped data space pairwise distances.
  - Select distance threshold that excludes upper $(1-\alpha)$ fraction of observed values.

Basic Network Construction Workflow

- Starting from from the N x D gene expression data matrix **X**, derive an N x d matrix **Y**, (d < D) using Laplacian Eigenmaps (or another dimensionality reduction technique).
- Construct an N x N matrix **W*** capturing pairwise Euclidean distances between row vectors of **Y** ('reduced gene profiles').
- Apply a threshold to the elements of **W*** to obtain a network (adjacency matrix) representation.

Original Milestones for First Term

- Phase I: Laplacian Eigenmaps + Network Reconstruction
  - Target Date: Middle of December 2010
  - Milestones:
    - (C++) Implementation and focused validation of Laplacian Eigenmaps
    - Basic (distance matrix threshold-based) network reconstruction.

Work Accomplished During First Term

- Key Items:
  - Built clean, object-oriented interface to efficient low-level linear algebra routines (ARPACK, etc.)
  - Organized data structures and associated operations to conserve memory and support scalability.
  - Established basic correctness of implementation using standard assessment data sets.

Original Milestones for Second Term

- Phase II: Integrated Testing of Network Reconstruction + Possible Extensions
  - Target Date: end of March 2011
  - Milestones:
    - Integrated testing of network reconstruction
    - Comparison of results obtained using nonlinear dimensionality reduction (LE), linear dimensionality reduction (PCA), and original data; comparison with leading reconstruction method.
    - Parameter Tuning
    - Possible: Diffusion Maps
    - Possible: Approximate Nearest Neighbor Algorithm

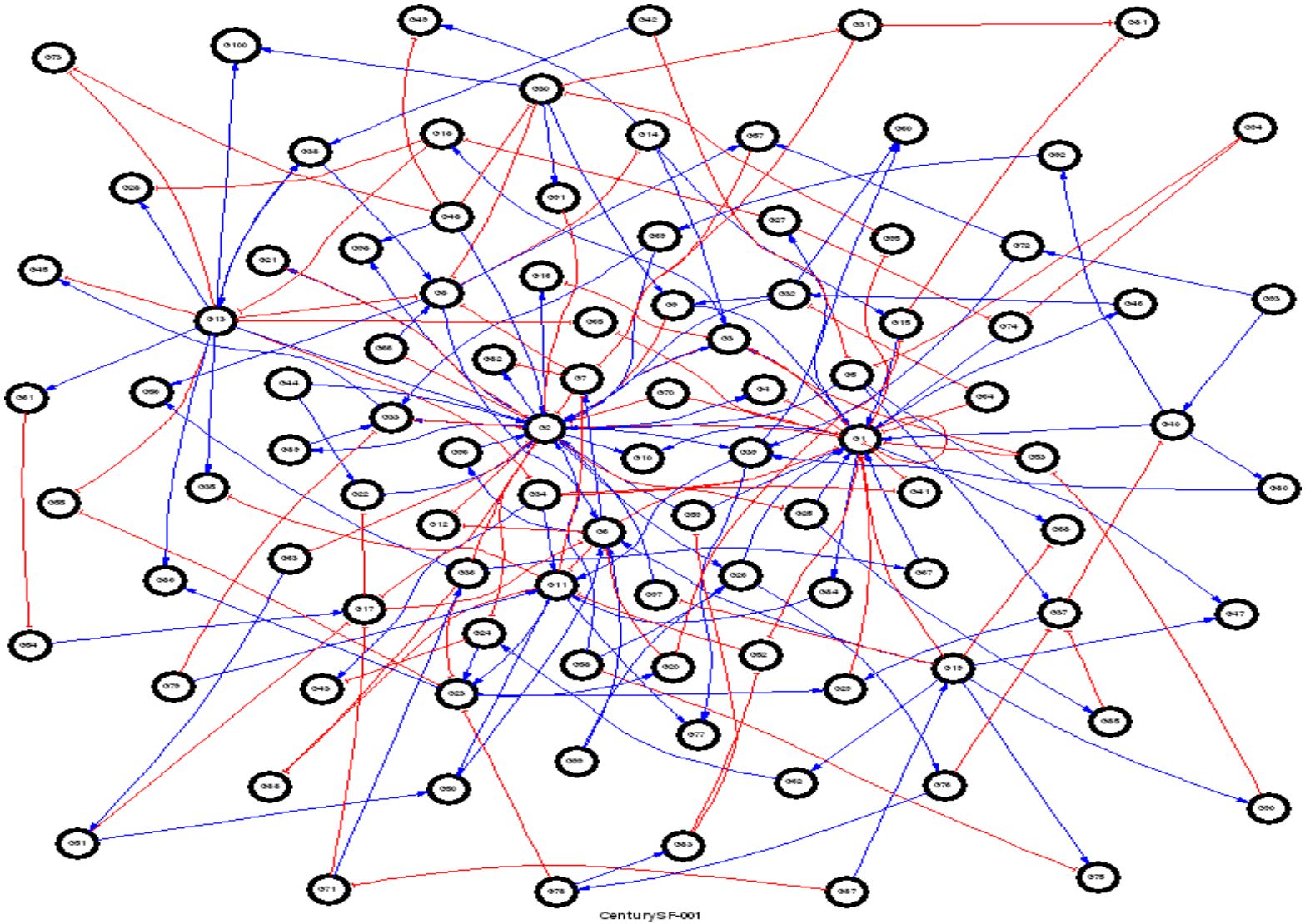
Current Term – Main Development Work

- Parameter Tuning: intrinsic dimensionality estimators (maximum likelihood, correlation).
- Parameter Tuning: nearest neighbor and kernel bandwidth selection by clustering in mapped data spaces (k-means algorithm, cluster quality assessment measures).
- Validation Support: Implementation of code to read/write network models, compare model results against each other and against known results.

Overview of Main Results

- Assessment using synthetic data from small simulated gene networks.
- Assessment using large gene expression data set with strong supporting publications:
  - Comparison with results obtained using leading network reconstruction method.
  - Comparison with results derived using original data and PCA-mapped data.

Synthetic Network Example



Synthetic Networks – Results Overview

- Aiming to assess recovery of positive interactions in 100 gene, 200 interaction network. Input data is 100 x 101 expression data matrix derived from (ODE) simulation.
- Average accuracy TP/(TP+FP) over 4 data sets.

Original Data	PCA-Mapped	LE-Mapped
0.46	0.47	0.44

Synthetic Networks - Discussion

- Results are not spectacular, but with consideration, somewhat unsurprising.
- Relatively direct methods based on a global distance threshold add many 'extra' edges in simple, prescribed networks because associations among co-regulated genes often appear stronger than the ones with their regulators. Such edges get selected first.
- LE focus on local neighborhoods may incur some global accuracy cost. Better parameter tuning may help, though hard in this case due to small data set sizes.

Synthetic Networks - Discussion

- More successful methods for global network recovery typically either:
  - Filter a larger pool of initially admitted edges using mathematical criteria, independent experimental data, or prior biological knowledge.
  - Try to fit parameters of explicit ODE-based models (currently only practical with simpler microbial gene networks).

Local Network Recovery

- Global network recovery is hard - essentially unrealistic without additional constraining data and other criteria, especially in more complex cells.
- Local network recovery around highly connected 'hubs' remains a more accessible and biomedically relevant problem.
- Many potential interactions are known, but challenge is to:
  - Identify which ones are active in a given cell type or disease context.
  - Prioritize novel interactions for further investigation.

Local Network Recovery

- Somewhat different challenges:
  - Much larger, noisier data sets with less prescribed biological contexts (relative to precise perturbation experiments simulated with synthetic data).
  - Relatively lower accuracy threshold relative to (still limited) validation data. 3-4 fold enrichment around hub gene (with respect to known targets) can be a solid, useful result – e.g., a starting point for prioritizing genes for labor-intensive lab investigation.

MYC Network - Background

- MYC is the most frequently deregulated gene in human cancers, largely due to its prominent role as a 'regulator of regulators'.
- Hundreds of validated direct targets, though far fewer are active in any given cell type.
- Major challenge is to fill in the many gaps – e.g., define cell/tumor type-specific target sets, identify novel targets, etc.

Validation/Assessment Approach

- Expand local (first-neighbor) network around MYC and match target genes against established database of biologically validated MYC (direct) targets.
- Leading network reconstruction method (ARACNE) recovers 56 direct targets using large cancer cell line data set.
  - 24 of 56 match target database (~42.9%).
  - Significant enrichment over ~ 10% expected by chance.
  - An additional set of 5 computationally identified targets was selected and biochemically validated.

Assessment/Validation Results

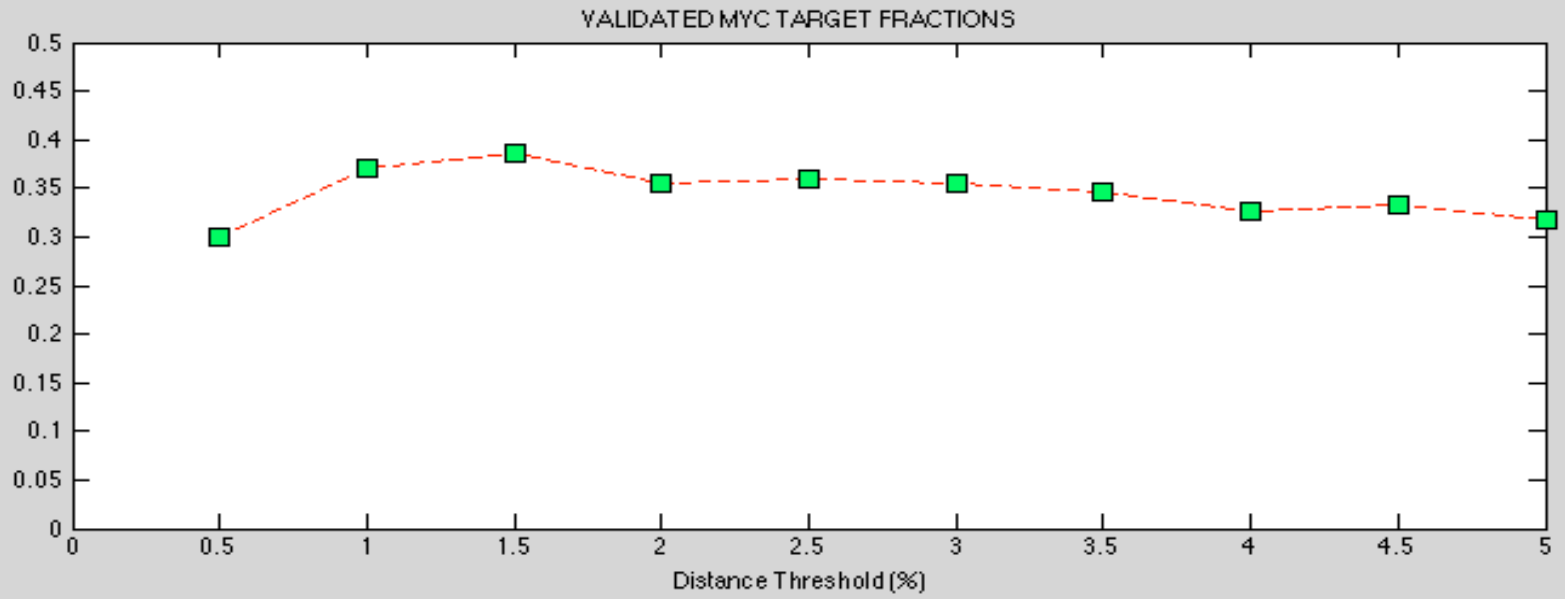
- With selected MYC network derived using Laplacian Eigenmaps-processed data, 23 of 61 direct targets match database.
  - ~37.7% versus ~42.9% for ARACNE method
  - Network derived by applying global threshold yielding edges for smallest 1% of distances.
  - Remark: ARACNE also applies thresholding strategy, but with mutual information-based measure, and subsequent edge pruning steps.

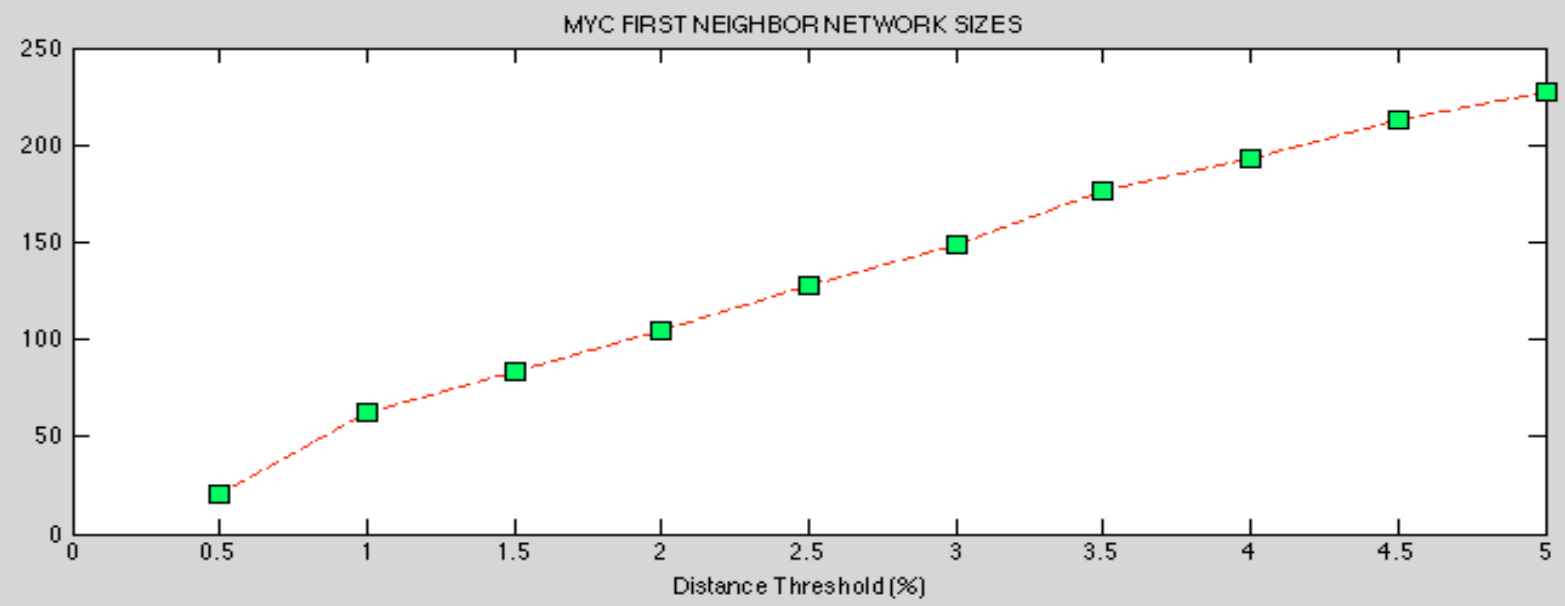
Assessment/Validation Details

- The 61 gene 1st neighbor network of the LE-based method overlaps with the corresponding 56 gene network of the ARACNE method by just 5 genes overall (and 3 genes in the validated groups).
- Remark: ARACNE developers indicate that their method does not aim to recover all possible edges, but rather a substantially enriched candidate set.
- Differences with respect to ARACNE could represent new information, but further assessment is required.

Assessment/Validation Details

- Parameter Selection
  - Target Dimensionality was set to 28, based on maximum over DR estimation methods.
  - NN and kernel bandwidth (σ) parameters were set by running small grid of typically applied parameter selections
    - NN = 8,12,16,20 σ = 0.5, 1
  - Selected results with best (k-means-derived) cluster structure in LE-mapped space:
    - NN = 16 σ = 0.5


VALIDATED MYC TARGET FRACTIONS
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
Distance Threshold (%)


MYC FIRST NEIGHBOR NETWORK SIZES
250
200
150
100
50
0
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
Distance Threshold (%)

Assessment/Validation Details

- Analogous network derivation strategy with original data and PCA-mapped data did not yield strong results. Percentage thresholds required to recover network edges around MYC gene:

LE-Mapped	PCA-Mapped	Original Data
1%	> 40%	> 50%

- Unreasonably high thresholds are required in original and PCA-mapped spaces.
- LE appears to produce meaningful distances over local network neighborhoods.

Original Milestones for Second Term

- Phase II: Integrated Testing of Network Reconstruction + Possible Extensions
  - Target Date: end of March 2011
  - Milestones:
    - Integrated testing of network reconstruction
    - Comparison of results obtained using nonlinear dimensionality reduction (LE), linear dimensionality reduction (PCA), and original data; comparison with leading reconstruction method.
    - Parameter Tuning
    - Possible: Diffusion Maps
      - Very limited implementation
      - Currently does not yield strong results, possibly due to steps taken to allow sparse eigenvalue problem. Further work required.
    - Possible: Approximate Nearest Neighbor Algorithm
      - Not implemented due to time constraints, limited motivation with manageable run times, and desire to focus on assessing and improving structure recovery without introducing potentially confounding factor.

Ongoing Work

- Improve Local Network Recovery
  - Consider broader measures of gene association, more sophisticated kernels.
  - Filter edges and prioritize targets using mathematical criteria, additional data sources.
- Further Assessment and Validation
  - Consider other known network hubs.
  - Broader biological database validation.
  - Work with lab collaborators to see if selected targets can be biochemically validated.

References

- **M. Belkin and P. Niyogi, *Laplacian Eigenmaps for dimensionality reduction and data representation, Neural Computation.* 15 (2004), No. 6, 1373-1396**
- **K Basso, A Margolin, G Stolovitzky, U Klein, R Dalla-Favera, A Califano: Reverse engineering of regulatory networks in human B cells. Nature Genetics 2005, 37(4):382-390.**
- M. Ehler, V. Rajapakse, B. Zeeberg, B. Brooks, J. Brown, W. Czaja, and R. F. Bonner, *Analysis of temporal-spatial co-variation within gene expression microarray data in an organogenesis model.* 6th International Symposium on Bioinformatics Research and Applications (ISBRA'10), Lecture Notes in Bioinformatics, Springer Verlag, 2010
- **A. A. Margolin, I Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R Dalla Favera, A. Califano, *ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context.* BMC Bioinformatics. 2006 Mar 20;7 Suppl 1:S7.**
- van der Maaten, Postma, van den Herik, *Dimensionality Reduction: A Comparative Review.* Tilburg Centre for Creative Computing Technical Report 2009-005
- von Luxburg, U. A Tutorial on Spectral Clustering. Statistics and Computing 17 (4), 395-416 (12 2007)
- **MYC TARGET DATABASE: http://www.myccancergene.org/**

Implementation Challenges

- Selecting and integrating appropriate (public domain) software to efficiently solve eigenvalue problem.

- Organizing data structures and operations to conserve memory and support scalability.

Linear Algebra Libraries

- Need to solve sparse, symmetric eigenvalue problem.
- Basic BLAS/LAPACK largely emphasize dense matrices.
- Evaluated several C++ packages
  - uBLAS (BLAS routines, relatively slow)
  - Armadillo++, Eigen (nice, almost MATLAB-like interface w/operator overloading, but meager support for sparse matrices/eigenproblems)

ARPACK

- ARnoldi PACKage: Fortran 77 library for solving large scale sparse eigenvalue problems
- Used by MATLAB (e.g., eigs function)
- For symmetric matrices, applies Lanczos Algorithm

ARPACK and Memory Management

- Reverse Communication Interface:
  - ARPACK routines do not operate directly on matrices
  - Instead: work with function defining matrix vector product. Allows matrices to be stored in any suitable format (or not at all).
- Implementation exploits this to represent matrices using compact adjacency lists, with fast 'in-place' operations where possible

Linear Algebra/ARPACK interface

- Organized ARPACK interface code, compressed matrix classes into convenient package, with overloaded operators and high-level, template-based methods.
- Basic, re-usable building block which will facilitate additional algorithm implementations.

Linear Algebra/ARPACK Interface

- ```
  extern "C" void dsaupd_(int *ido, char *bmat,
  int *n, char *which,int *nev, double *tol,
  double *resid, int *ncv, double *v, int *ldv,
  int *iparam, int *ipntr, double *workd,
  double *workl, int *lworkl, int *info);
  ```

- ```
  template<typename T>
  void sparseSymEigSolve(
  const CompressedMatrix<T>& M,
  const Matrix<T>& evecs,
  const NumVector<T>& evals);
  ```

Maximum Likelihood Estimate of Intrinsic Dimensionality

- Let $T_k(x)$ denote the Euclidean distance from a fixed point x to its k-th nearest neighbor in the sample of size N.
- Set
$$\hat{m}_k(x) = \left[\frac{1}{k-1} \sum_{j=1}^{k-1} \log \frac{T_k(x)}{T_j(x)} \right]^{-1}$$
- Set
$$\hat{m}_k = \frac{1}{N} \sum_{i=1}^{N} \hat{m}_k(x_i)$$
- Average above over k = 6 … 12