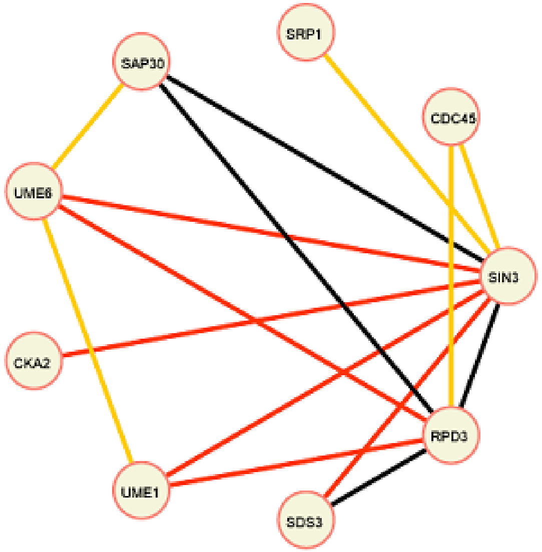
# Protein Complex Identification by Supervised Graph Clustering

Yanjun Qi[1], Fernanda Balem[2], Christos Faloutsos[1], Judith Klein-Seetharaman[1,2], Ziv Bar-Joseph[1]

[1]School of Computer Science, Carnegie Mellon University,
[2]University of Pittsburgh School of Medicine

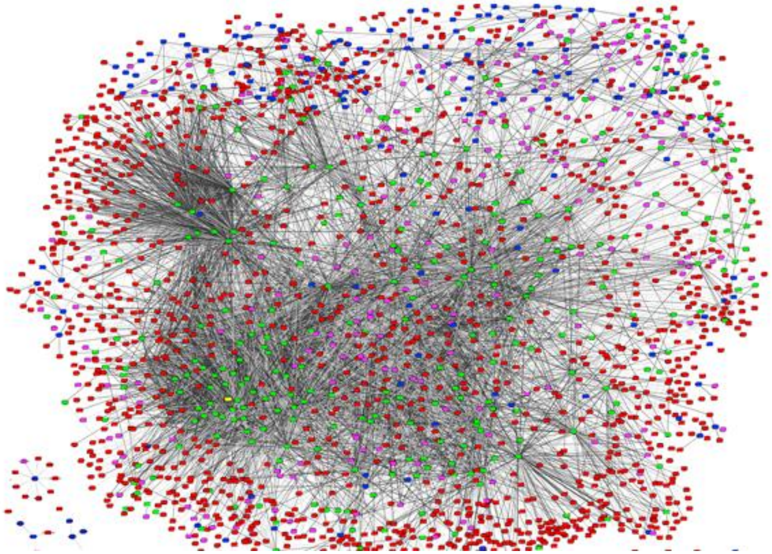


# Protein-Protein Interaction (PPI)

- Involved in most activities in the cell
  - Can be used to infer function
  - Combined to infer pathways and complexes

- Lots of experimental work
  - Mass spectrometry (Gavin et al 2006; Krogan et al 2006)
  - Yeast two-hybrid (Rual et al 2005)

- Lots of computational work
  - Bayesian networks (Jansen et al 2003)
  - Random Forest (Qi et al 2006)

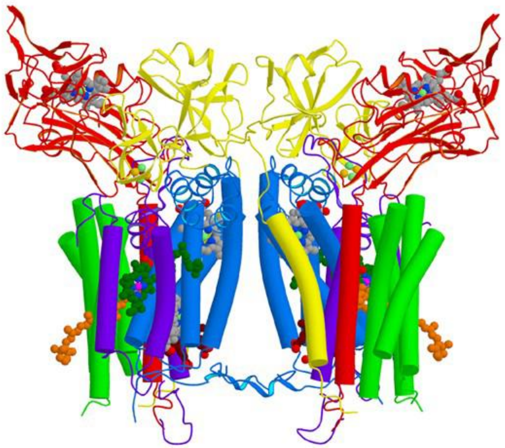


# Protein Complexes

- A set of proteins working together as a ‘super machine’

- Complex member interacts with all or part of the group

- Correct identification leads to better understanding of function and mechanisms

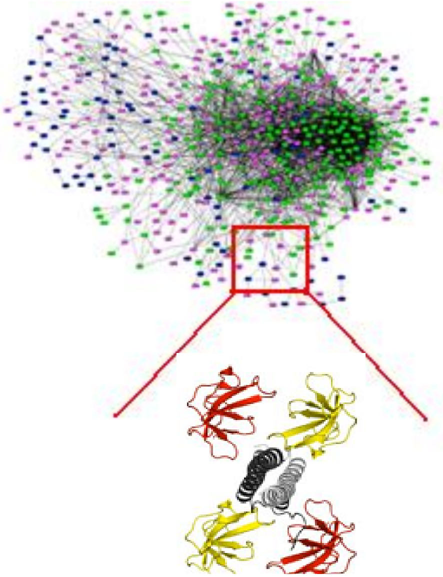


# Identifying complexes in a PPI graph

- Problem statement: Given a PPI graph identify the subsets of interacting proteins that form complexes

- Algorithms for addressing this problem were used in the high throughput mass spec papers
- Many other algorithms suggested for this task

***PPI Network***

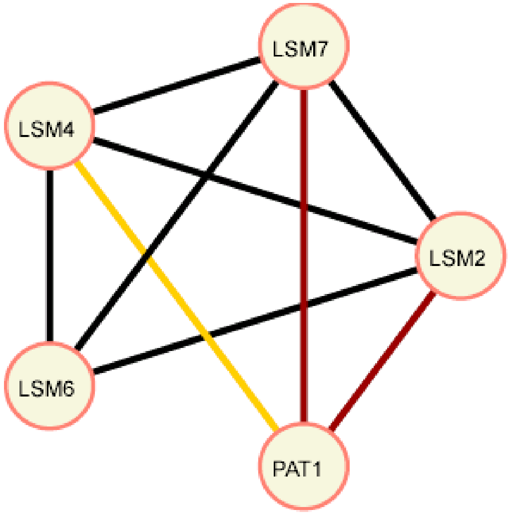


***Protein Complex***

# Methods for identifying complexes in PPI graph focus on cliques …

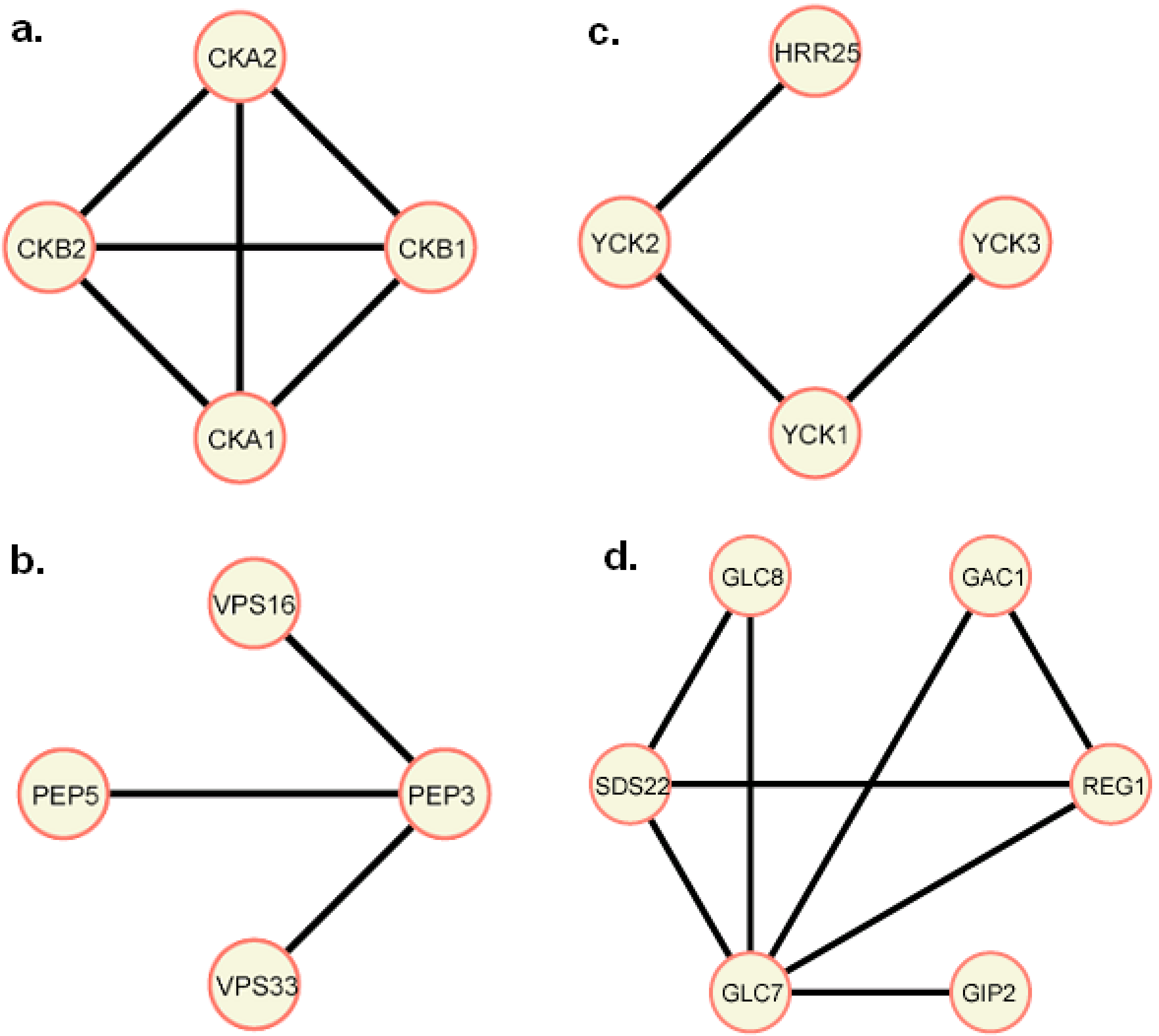
- Prior methods looked for dense subgraphs (cliques)
- Methods mainly differed in how the graph was segmented
- Most methods treated the graph as a binary graph ignoring weight on the edges

  - such weight can be obtained from both, computational predictions and experimental data
- Example: MCODE (Bader et al 2003) detects densely connected regions in PPI networks using vertex weights representing local neighborhood density



# … while many other topological structures are present

Projecting computationally predicted PPI graph on curated MIPS complexes



*……. More*

# Method

# Key ideas for our method

- Utilize available data for training
  - Supervised instead of unsupervised methods
- Summarize properties of the possible topological structures
  - Use common subgraph features
- Take into account the biological properties of complexes
  - Use information about the weight and size of proteins

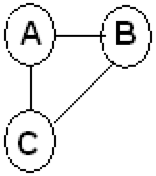
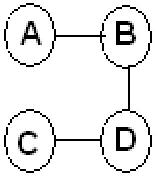
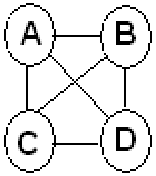
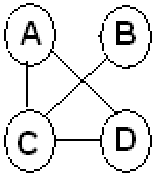
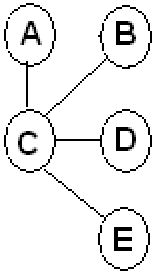
# Features used to model subgraphs

Systems
Biology
Group

- Subgraph properties as features
  - –Various topological properties from graph
  - –Biological attributes of complexes
- Can be computed on projections of known complexes on our PPI graph

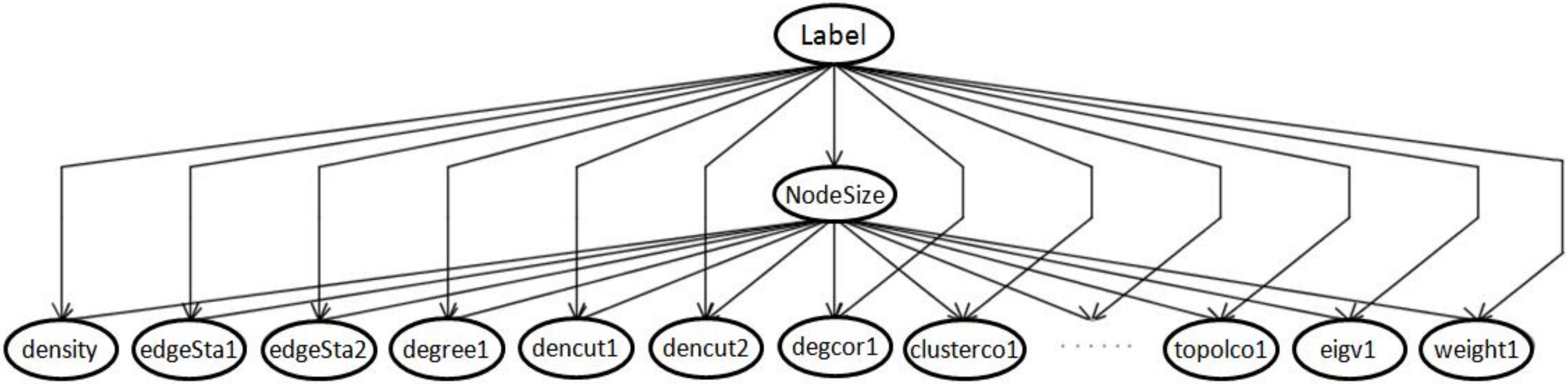
| **No.** | **Sub-Graph Property** |
|---|---|
| 1 | Vertex Size |
| 2 | Graph Density |
| 3 | Edge Weight Ave / Var |
| 4 | Node degree Ave / Max |
| 5 | Degree Correlation Ave / Max |
| 6 | Clustering Coefficient Ave / Max |
| 7 | Topological Coefficient Ave / Max |
| 8 | First Two Eigen Value |
| 9 | Fraction of Edge Weight > Certain Cutoff |
| | |
| 10 | Complex Member Protein Size Ave / Max |
| 11 | Complex Member Protein Weight Ave / Max |

# Example

| |  |  |  |  |  |
|---|---|---|---|---|---|
| **Node Size** | 3 | 4 | 4 | 4 | 5 |
| **Density** | 1 | 0.5 | 1 | 0.667 | 0.4 |
| **……** | | | | | |

# Probabilistic model for complex features

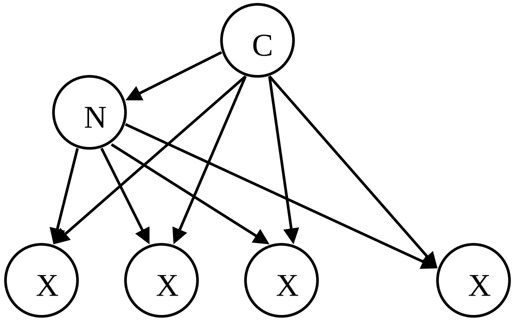
- We use a Bayesian Network to represent the joint probability distribution of the various features we use



# Log likelihood ratio for complexes

- We use a Bayesian Network to represent the joint probability distribution of the various features we use

- Bayesian Network (BN)
  - *C* : If this subgraph is a complex (1) or not (0)
  - *N* : Number of nodes in subgraph
  - *Xi* : Properties of subgraph



$$L = \log \frac{p(c=1 \mid n, x_1, x_2, \ldots, x_m)}{p(c=0 \mid n, x_1, x_2, \ldots, x_m)}$$

# Learning

- BN parameters were learned using MLE
  - Trained from known complexes and random sampled subgraphs with of the same size (non-complexes)
  - Discretize continuous features
  - Bayesian Prior to smooth multinomial parameters

- Evaluate candidate subgraphs with the log ratio score *L*

$$L = \log \frac{p(c=1 \mid n, x_1, x_2, \ldots, x_m)}{p(c=0 \mid n, x_1, x_2, \ldots, x_m)} = \log \frac{p(c=1)\, p(n \mid c=1) \prod_{k=1}^{m} p(x_k \mid n, c=1)}{p(c=0)\, p(n \mid c=0) \prod_{k=1}^{m} p(x_k \mid n, c=0)}$$

# Searching for high scoring complexes

- Given our likelihood function we would like to find high scoring complexes (maximizing the log likelihood ratio)

- Lemma: *Identifying the set of maximally scoring subgraphs in our PPI graph is NP-hard*

- We thus employ the iterated simulated annealing search on the log-ratio score

Systems Biology Group

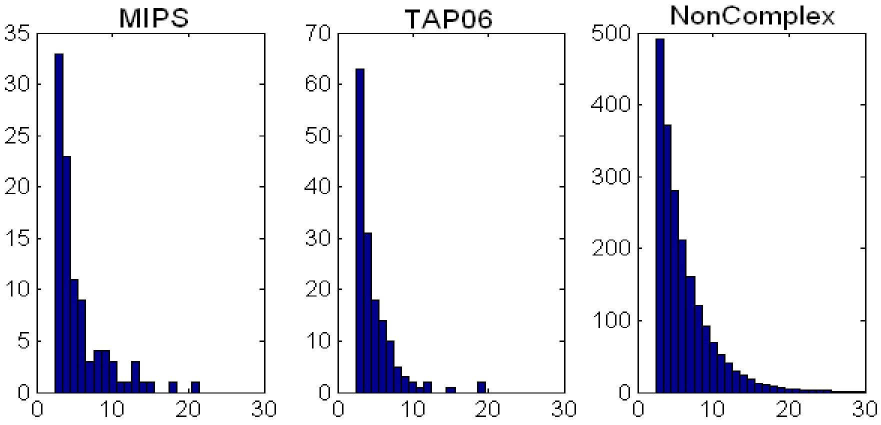
# Evaluation and Results

# Experimental Setup

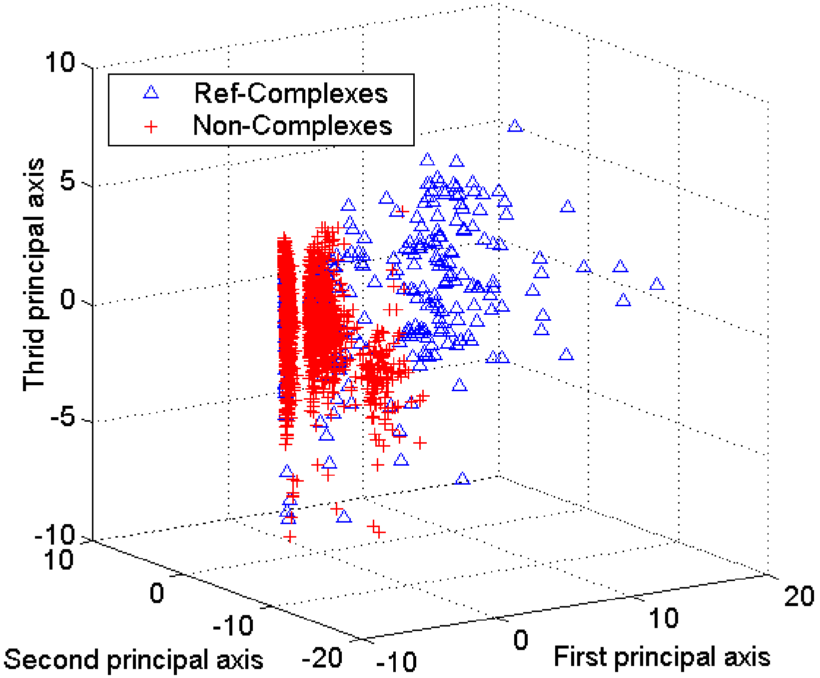
- Positive training data (complexes)
  - Set1: MIPS Yeast complex catalog: a curated set of ~100 protein complexes
  - Set2: TAP06 Yeast complex catalog (Gavin et al 2006): a reliable experimental set of ~150 complexes
  - Complex size (nodes' num.) follows a power law
- Negative training data (pseudo non-complexes)
  - Generated from randomly selected nodes in the graph
  - Size distribution is similar as the positive complexes (for each of the two sets)

# Data Distribution

**Node size distribution**

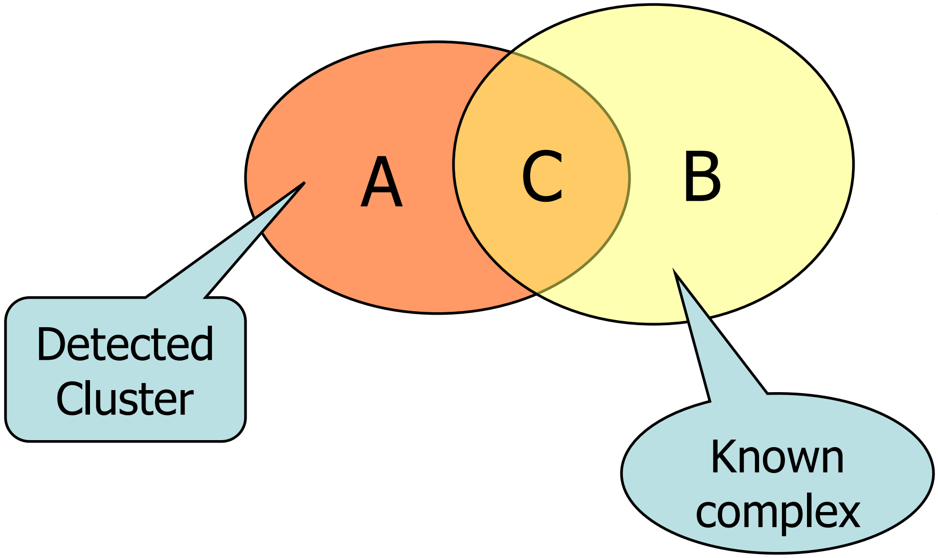


**Feature distribution**



# Evaluation

- Train on set1 and evaluate on set 2 and vice versa
- Precision / Recall / F1 measures
- A cluster “detects” a complex if



A : Number of proteins only in the cluster
B : Number of proteins only in the complex
C : Number of proteins shared between the two

We set the threshold (p) to be *50%*

$$\frac{C}{A+C} > p \quad \& \quad \frac{C}{B+C} > p$$

# Performance Comparison: Training on MIPS

- On yeast predicted PPI graph (Qi et al 2006, ~2000 nodes)
- Compared to three other methods:
  - MCODE which looks for highly interconnected regions (Bader et al 2003)
  - Search relying on density feature only
  - Same set of features using SVM rather than BN

- Training on MIPS, testing on TAP06

| Methods | Precision | Recall | F1 |
|---|---|---|---|
| **Density** | 0.217 | 0.409 | 0.283 |
| **MCODE** | 0.293 | 0.088 | 0.135 |
| **SVM** | 0.247 | 0.377 | 0.298 |
| **BN** | 0.312 | 0.489 | 0.381 |

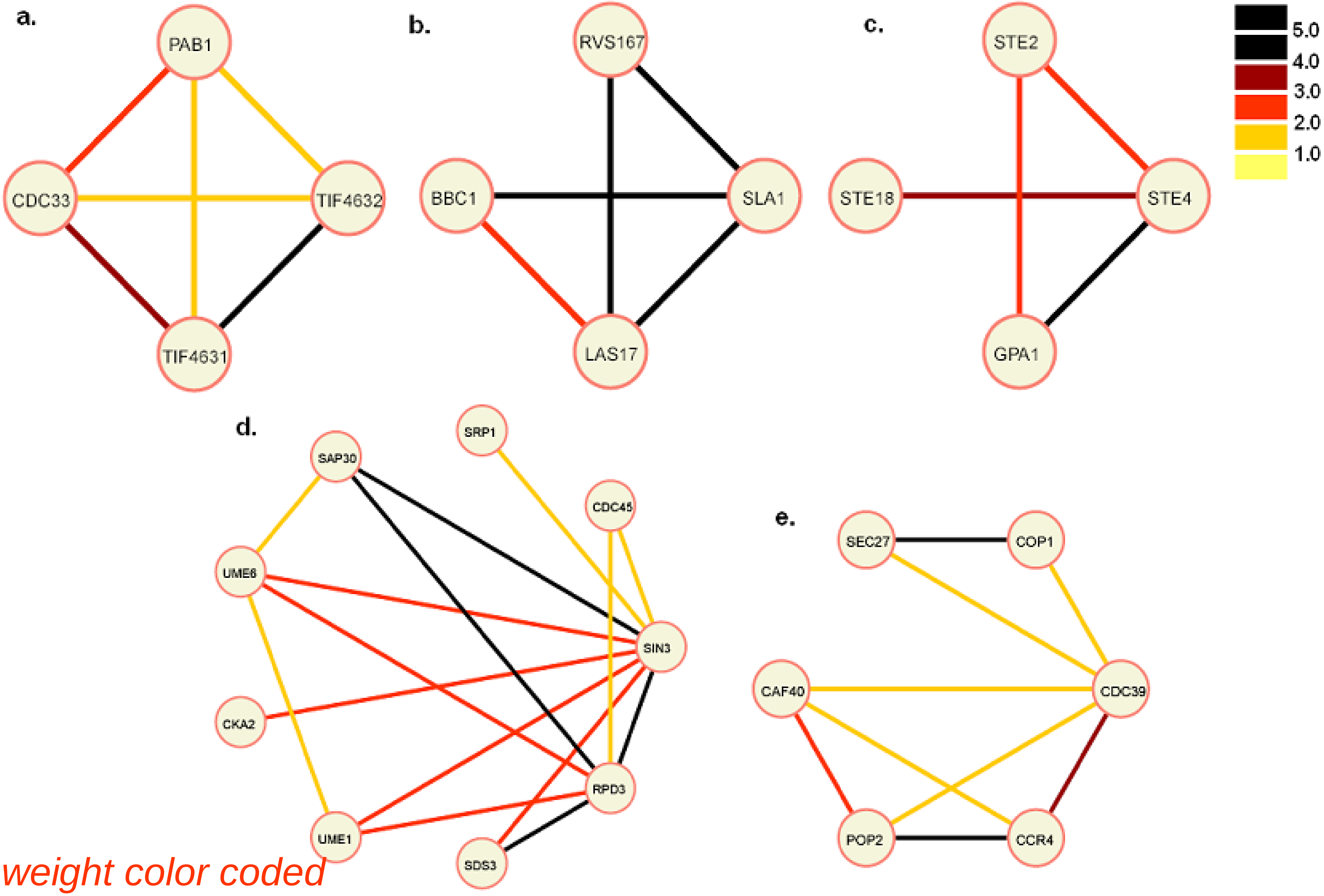
# Performance Comparison: Training on TAP

- Training on TAP06, testing on MIPS
- Compared to three other methods:

| **Methods** | **Precision** | **Recall** | **F1** |
|---|---|---|---|
| **Density** | 0.143 | 0.515 | 0.224 |
| **MCODE** | 0.146 | 0.063 | 0.088 |
| **SVM** | 0.176 | 0.379 | 0.240 |
| **BN** | 0.219 | 0.537 | 0.312 |

- MCODE tends to find a few big clusters …

# Examples of identified new complexes



*Edge weight color coded*

# Conclusions and future work

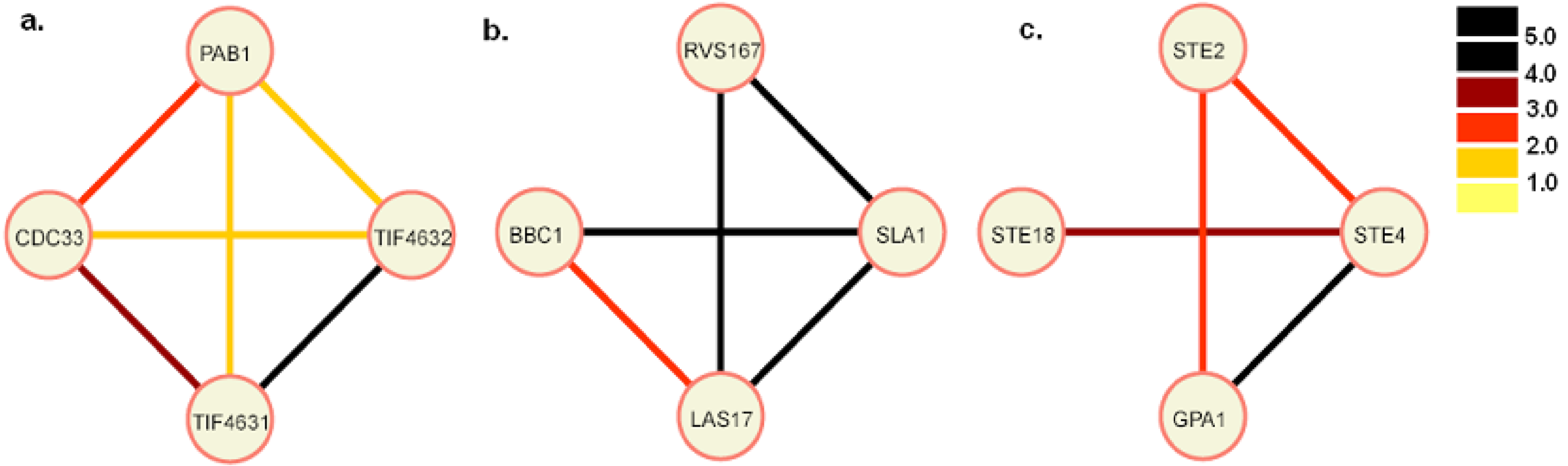
Systems Biology Group

- Supervised method can identify complexes that are missed when using strong topological assumptions
- Utilizing edge weight leads to higher predictions and recall
- Can be used whenever weight information is available
- Further improvements:
    - Better local search algorithm
    - Other features

# Acknowledgements

- Funding
  - NSF grants CAREER 0448453, CAREER CC044917, NIH NO1 AI-5001
  - ISMB Travel Fellowship (though unable to use it …)

**www.cs.cmu.edu/~qyj/SuperComplex/index.html**



# Heuristic Local Search

**Search**:

- Accept the new cluster candidate if with higher score
- If lower, accept with probability $exp(I' - I)/T$
- T : temperature parameter, decreasing by a scaling factor *alpha* after each round
- Accepted cluster must score higher than a threshold

**When to Stop:**

- $N( i, k+1) = \emptyset$ (*k-th* round)
- Number of round since the last score improvements larger than a specified number
- k is larger than a specified number

**Expand current cluster:**

- Generate a sub-set V* from all neighbors of current cluster
- Top M nodes ranked by their max-weight to current cluster

# Algorithm

Systems Biology Group

**Input**:

- Weighted protein-protein interaction network;
- A training set of complexes and non-complexes;

**Output:**

- Discovered list of detected clusters;

**Complex Model Parameter Estimation:**

- Extract features from positive and negative training examples;
- Calculate MLE parameters on the multinomial distributions;

**Search for Complexes:**

- Starting from the seeding nodes , apply simulated annealing search to identify candidate complexes;
- Output detected clusters ranked with log-ratio scores