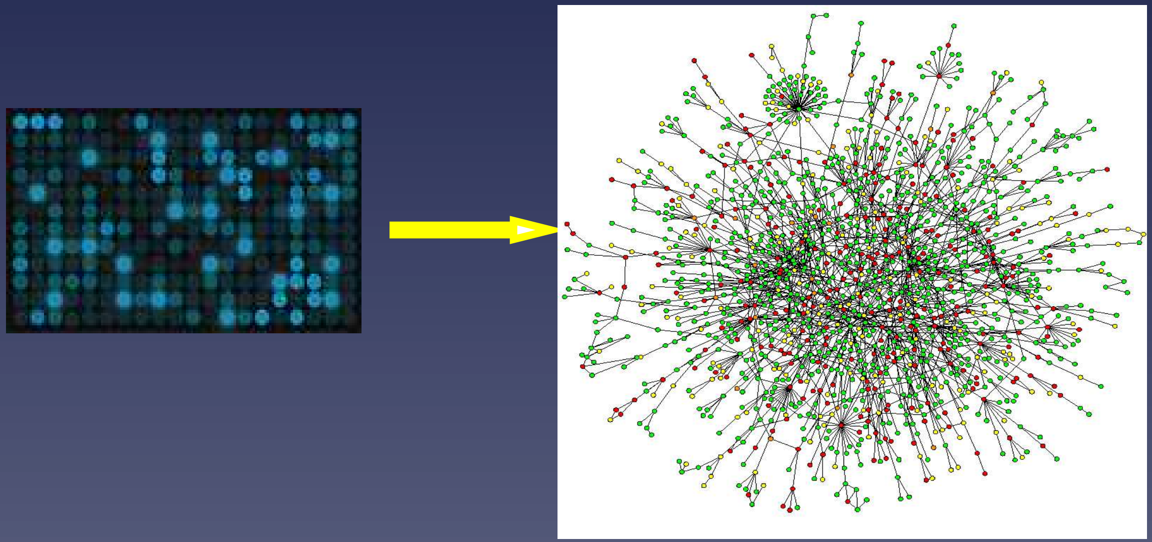
Supervised Graph Inference

Jean-Philippe Vert
Ecole des Mines de Paris, Computational Biology group
`Jean-Philippe.Vert@mines.org`

Yoshihiro Yamanishi
Kyoto University, Bioinformatics Center
`yoshi@kuicr.kyoto-u.ac.jp`

NIPS, December 15, 2004.

Motivations: systems biology



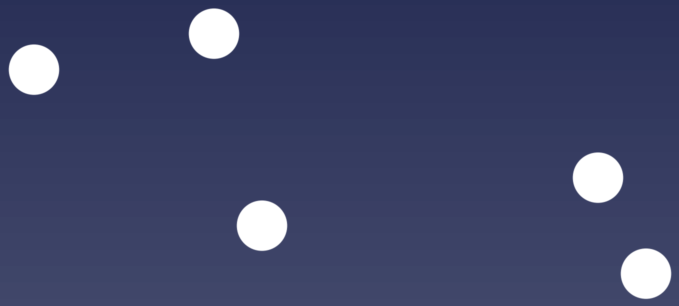
- Gene expression
- Sequence
- Protein structure
- Protein localization, etc...

- Regulatory network
- Signaling pathways
- Metabolic pathways
- Interaction network, etc...

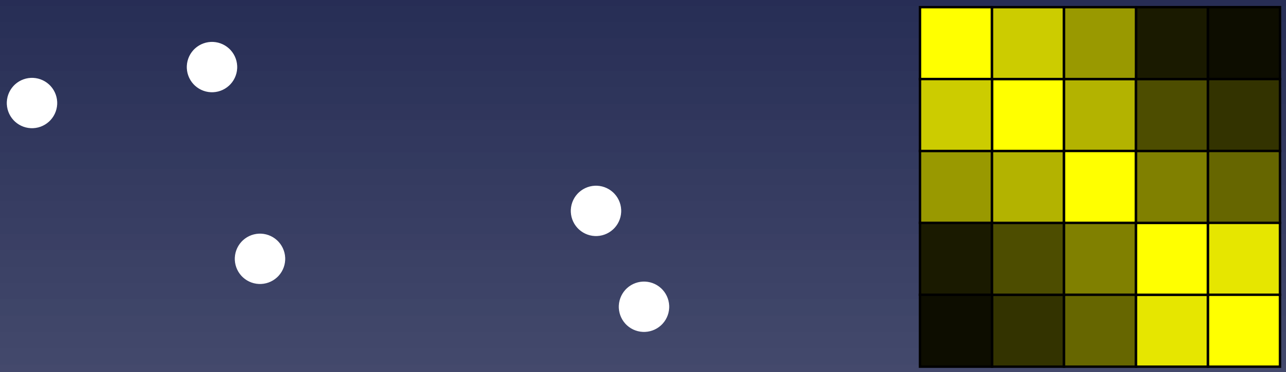
Related approaches

- Bayesian nets for regulatory networks (Friedman et al. 2000)
- Boolean networks (Akutsu, 2000)
- Nearest neighbors method (Marcotte et al, 1999)

Example: nearest neighbors method



Example: nearest neighbors method



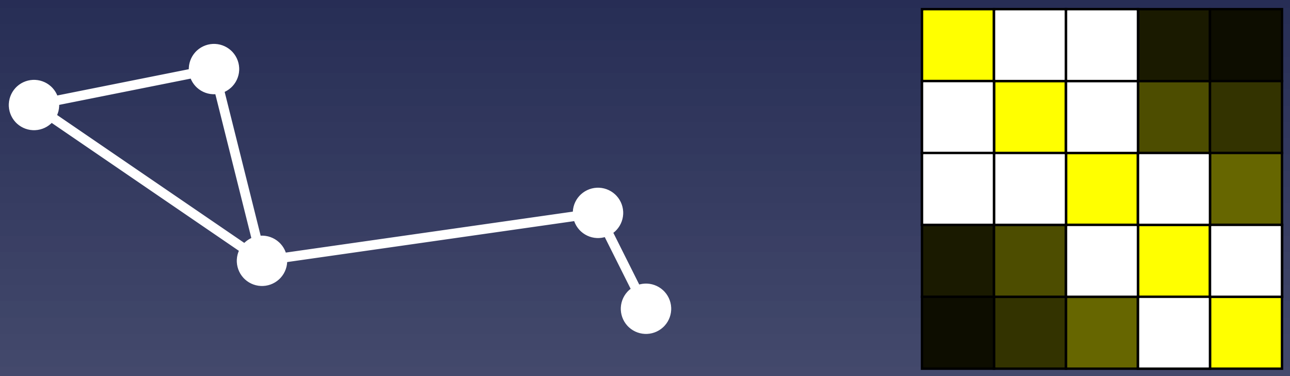
Example: nearest neighbors method

Example: nearest neighbors method

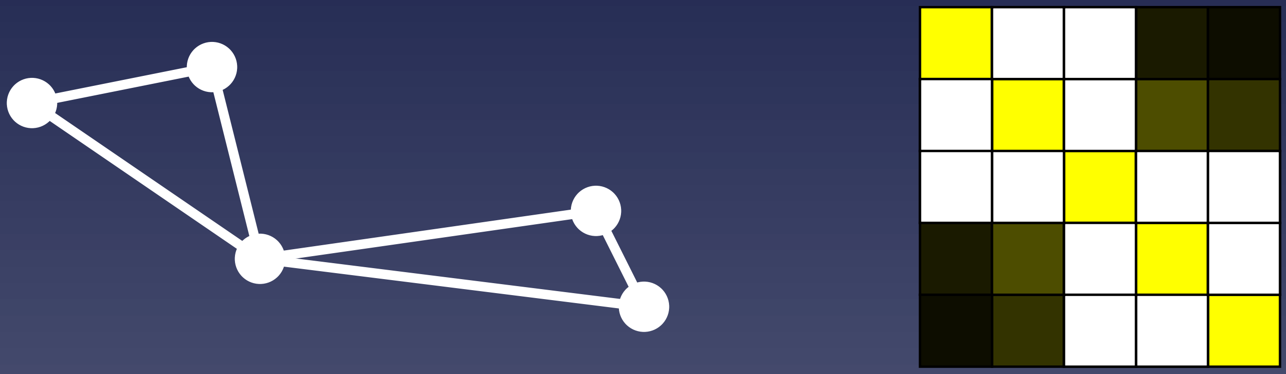
Example: nearest neighbors method

Example: nearest neighbors method

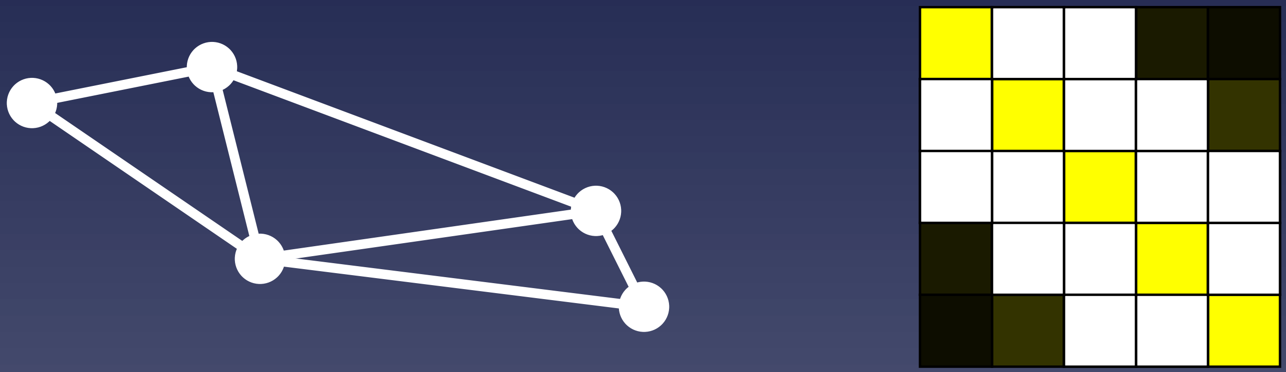
Example: nearest neighbors method



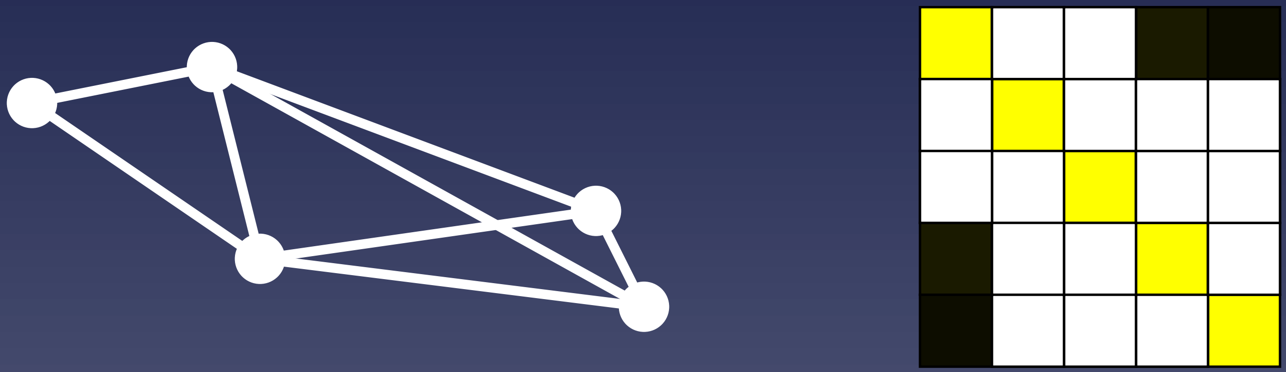
Example: nearest neighbors method



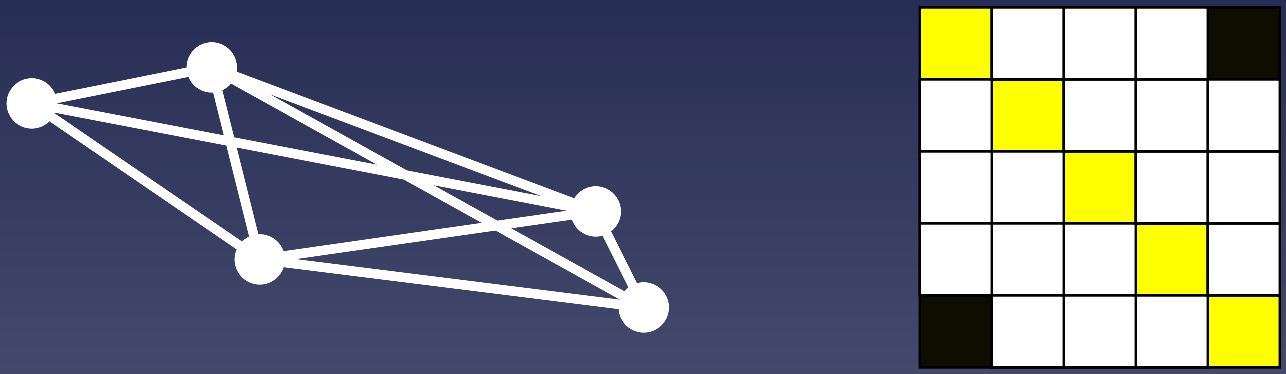
Example: nearest neighbors method



Example: nearest neighbors method

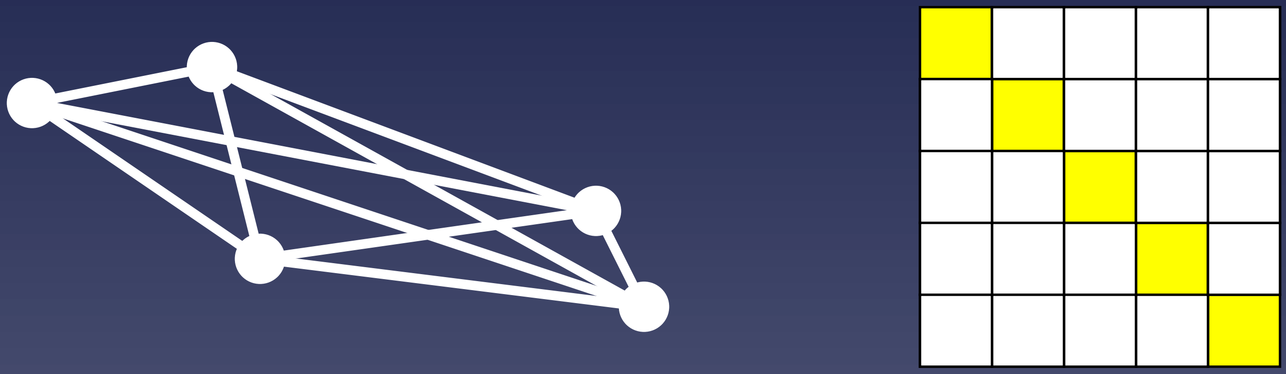


Example: nearest neighbors method

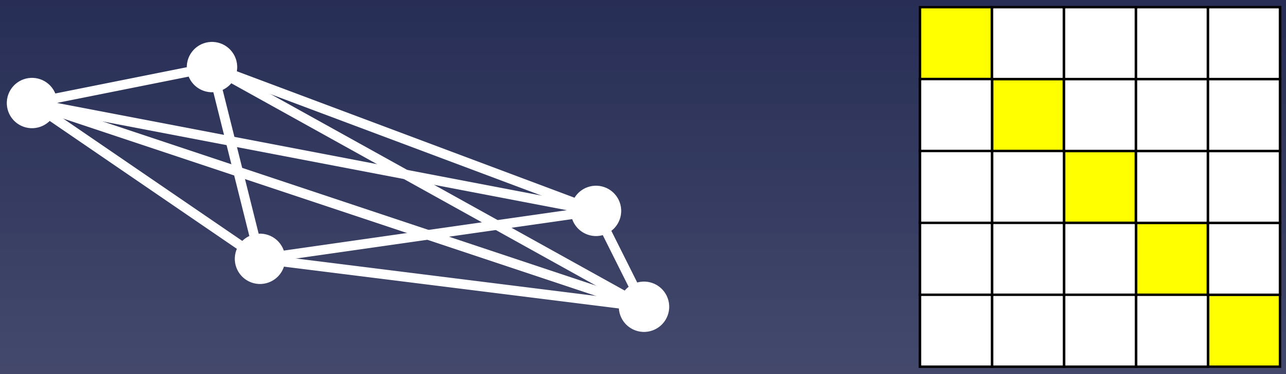


🟨	⬜	⬜	⬜	⬛
⬜	🟨	⬜	⬜	⬜
⬜	⬜	🟨	⬜	⬜
⬜	⬜	⬜	🟨	⬜
⬛	⬜	⬜	⬜	🟨

Example: nearest neighbors method

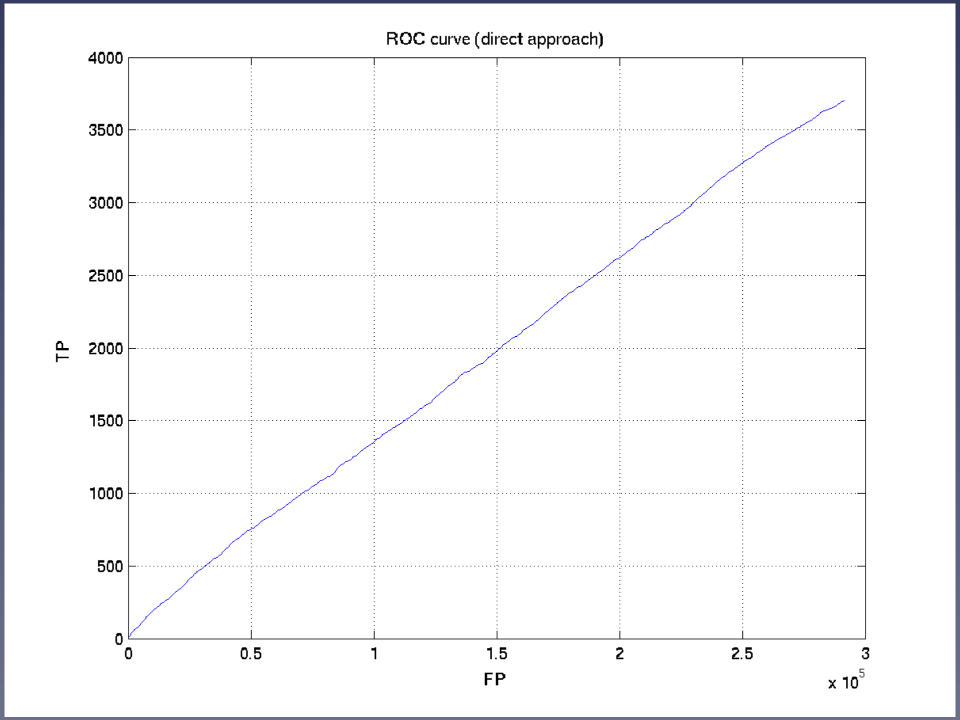


Example: nearest neighbors method



Application: metabolic network reconstruction

The metabolic network of the yeast involves 769 genes. Each gene is represented by 157 expression measurements. (ROC=0.52)



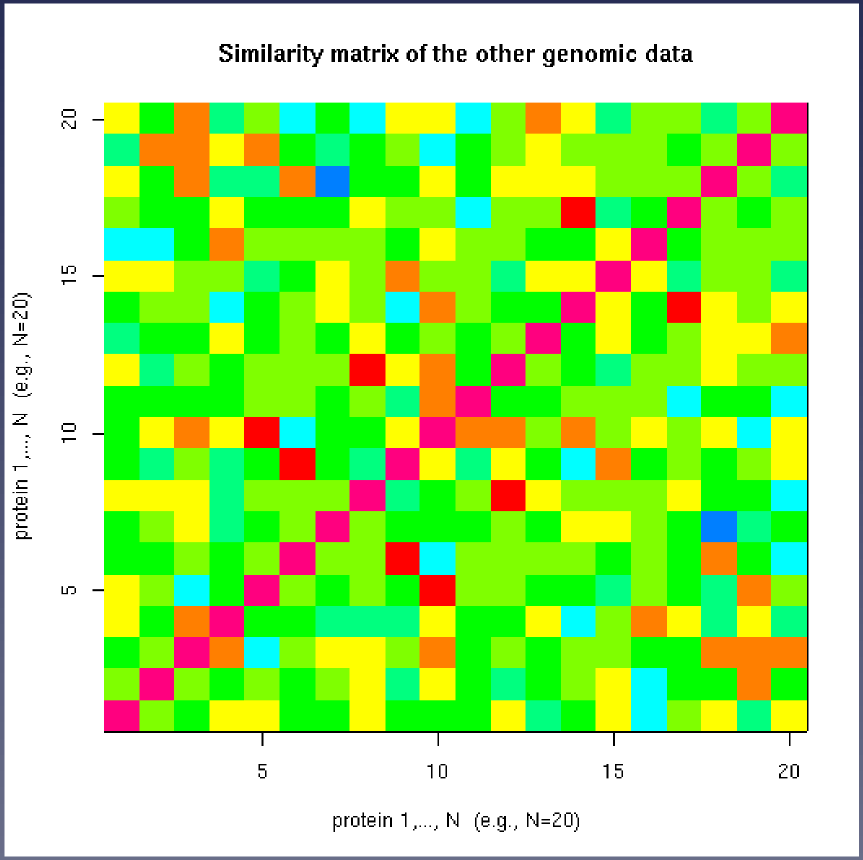
What is wrong?

- What similarity measure between profiles should be use?

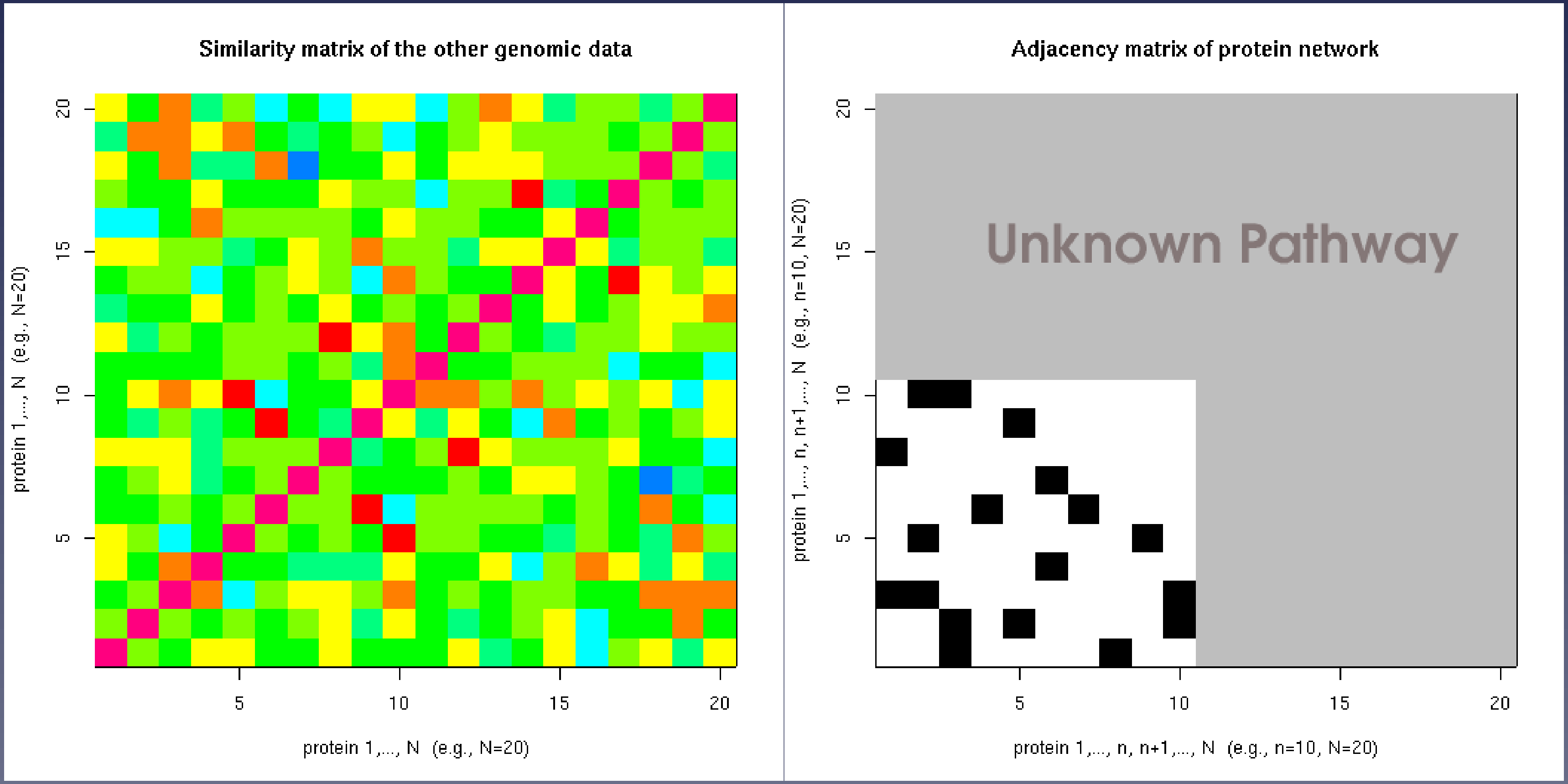
What is wrong?

- What similarity measure between profiles should be use?
- Which network are we expecting to recover?

The supervised gene inference problem

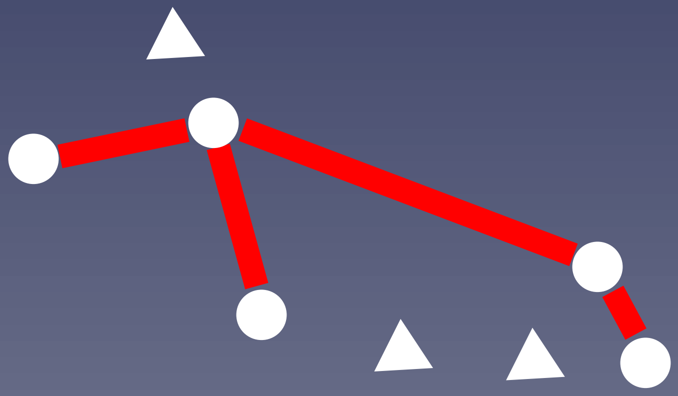


The supervised gene inference problem



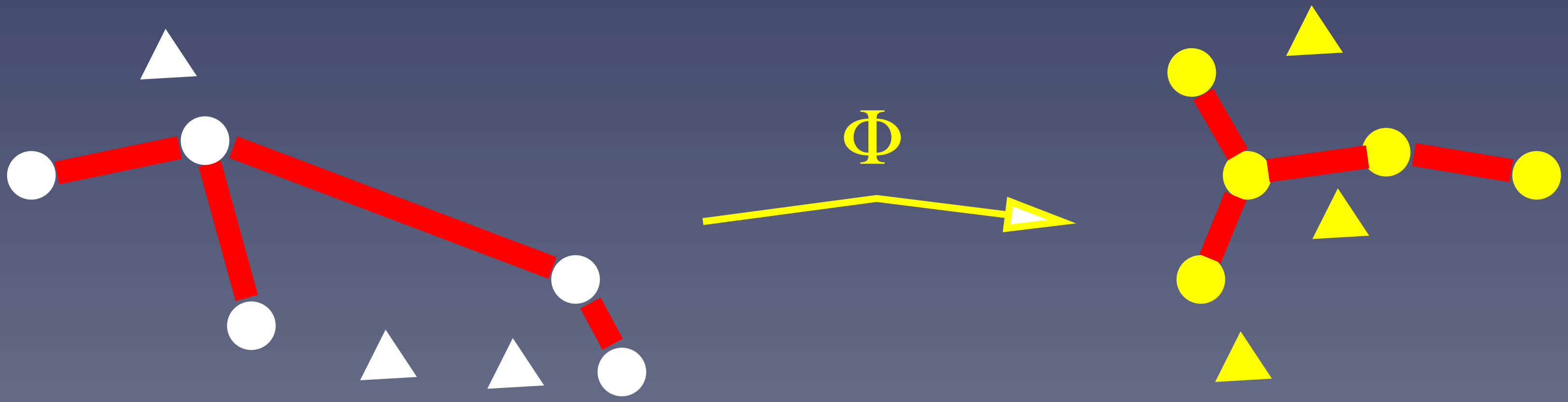
The main idea

Supervised graph inference
through
distance metric learning



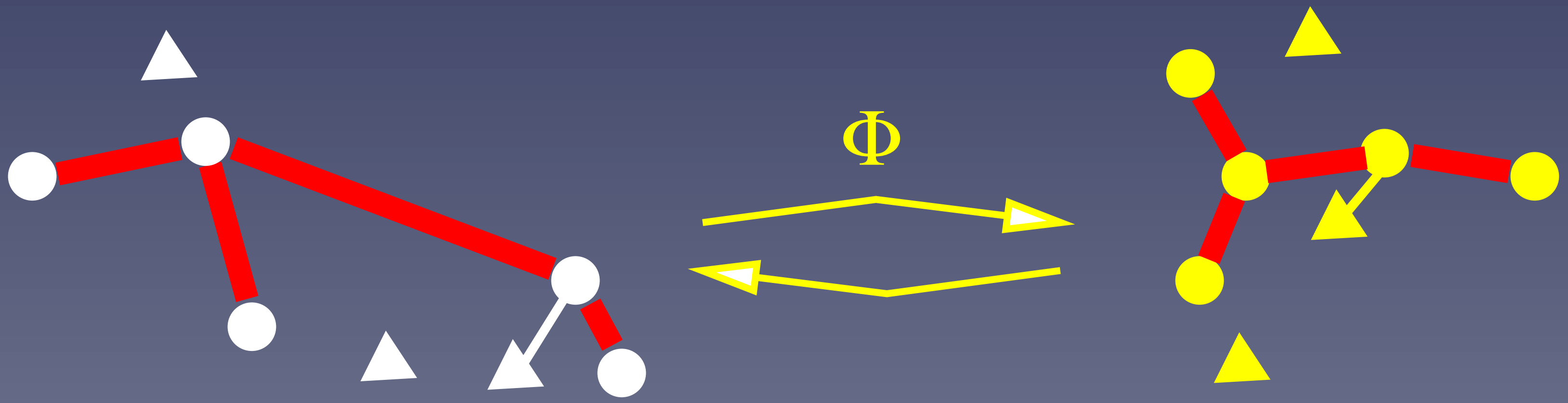
The main idea

Supervised graph inference
through
distance metric learning



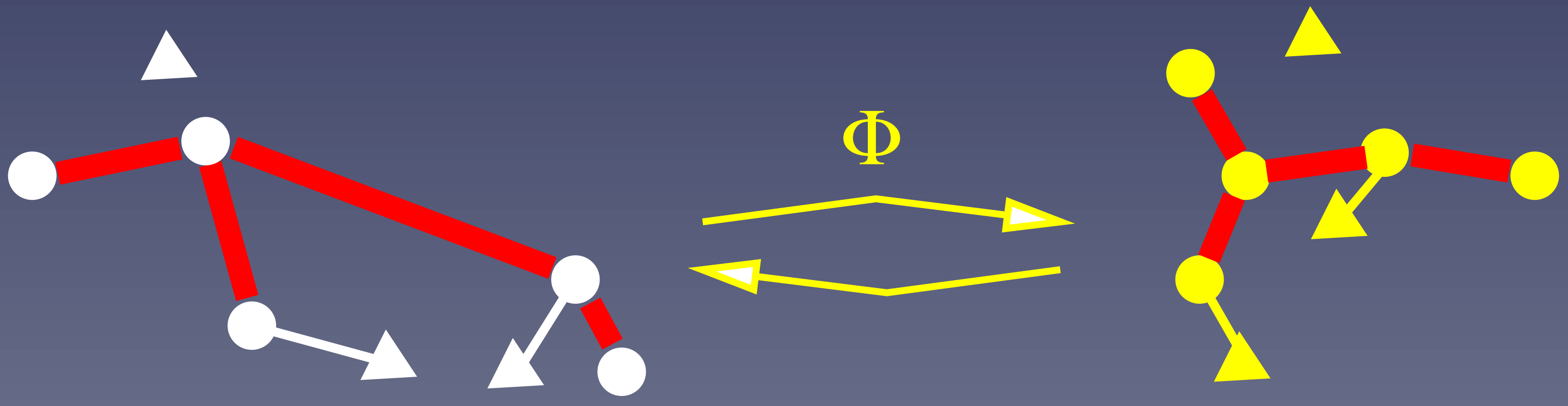
The main idea

Supervised graph inference
through
distance metric learning



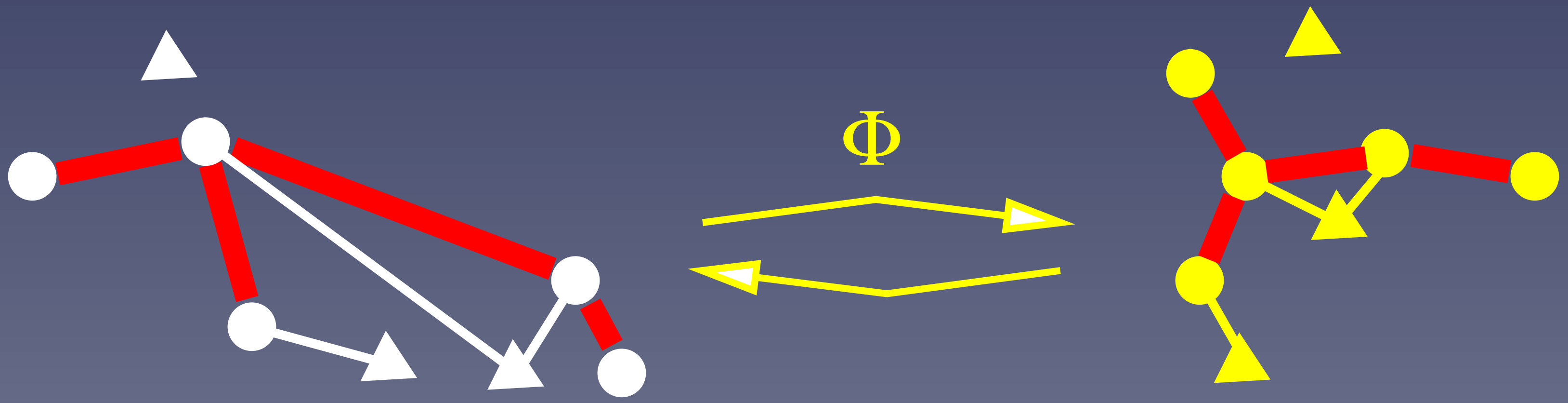
The main idea

Supervised graph inference
through
distance metric learning



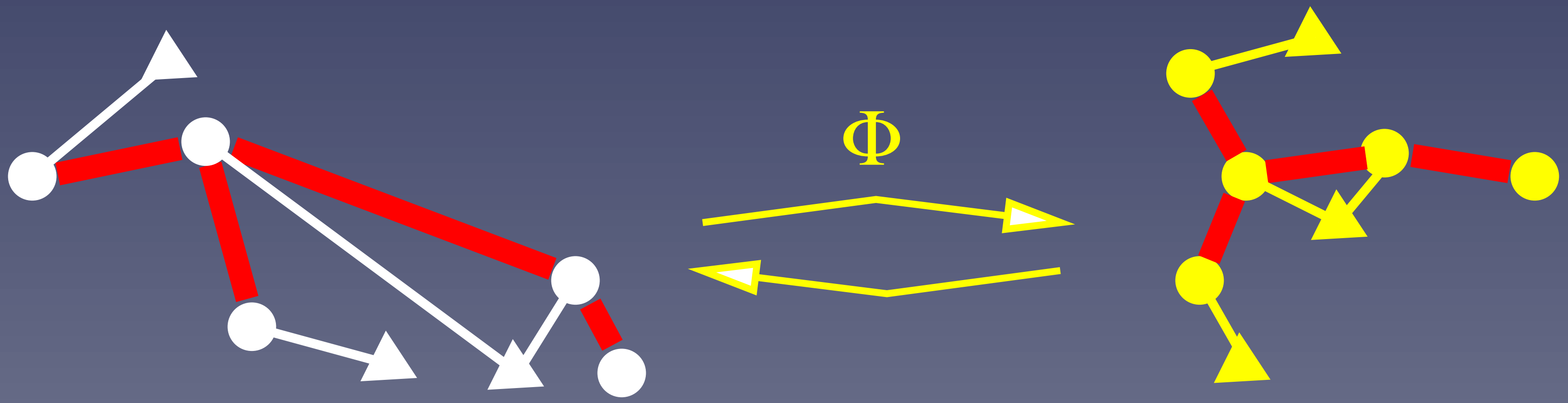
The main idea

Supervised graph inference
through
distance metric learning



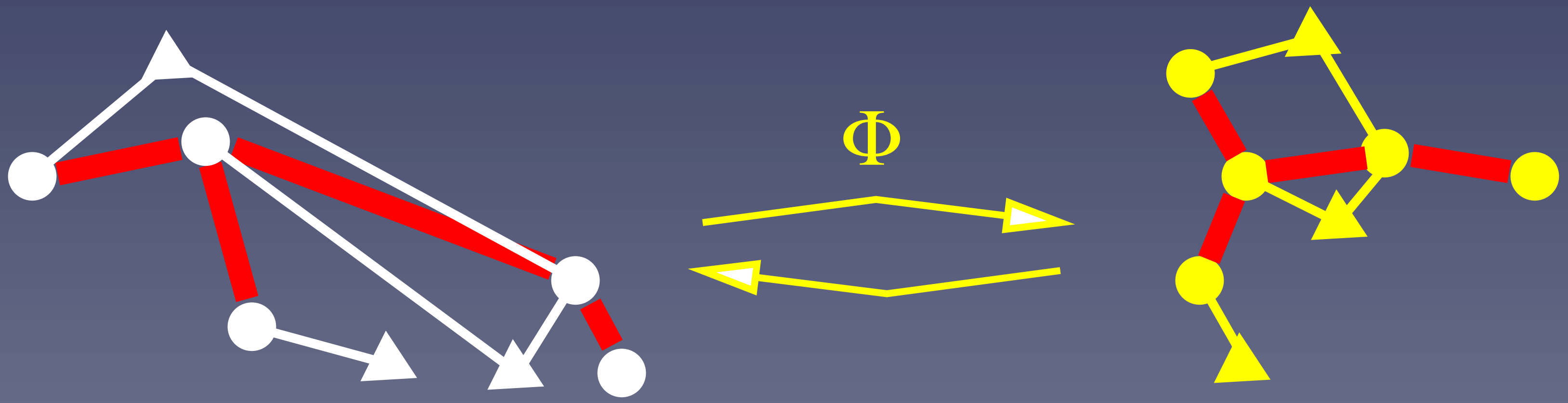
The main idea

Supervised graph inference
through
distance metric learning



The main idea

Supervised graph inference
through
distance metric learning



Learning the mapping Φ

- Let us consider mappings $\mathcal{X} \to \mathbb{R}^d$ ($\mathcal{X}$ being endowed with a p.d. kernel K):

$$\Phi(x) = (f_1(x), \ldots, f_d(x))' \in \mathbb{R}^d$$

  made of orthogonal features $f_i \in \mathcal{H}_K$ in the RKHS

Learning the mapping Φ

- Let us consider mappings $\mathcal{X} \rightarrow \mathbb{R}^d$ ($\mathcal{X}$ being endowed with a p.d. kernel K):

$$\Phi(x) = (f_1(x), \ldots, f_d(x))' \in \mathbb{R}^d$$

made of orthogonal features $f_i \in \mathcal{H}_K$ in the RKHS

- A possible criterion to ensure that connected genes in the known network have similar value is to minimize:

$$\min_{f \in \mathcal{H}_K} \frac{\sum_{(i,j) \in E} \left(f(x_i) - f(x_j)\right)^2 - \sum_{(i,j) \notin E} \left(f(x_i) - f(x_j)\right)^2}{\sum_{i=1}^{n} f(x_i)^2}$$

Regularized risk

- If the data are centered ($\sum_i x_i = 0$), then this is equivalent to minimizing:

$$\min_{f \in \mathcal{H}_K} \frac{\sum_{i \sim j} \left(f(x_i) - f(x_j) \right)^2}{\sum_{i=1}^{n} f(x_i)^2}.$$

Regularized risk

- If the data are centered ($\sum_i x_i = 0$), then this is equivalent to minimizing:

$$\min_{f \in \mathcal{H}_K} \frac{\sum_{i \sim j} \left(f(x_i) - f(x_j) \right)^2}{\sum_{i=1}^n f(x_i)^2}.$$

- For statistical reasons (particularly in large dimension), it is safer to minimize:

$$\min_{f \in \mathcal{H}_K} \frac{\sum_{i \sim j} \left(f(x_i) - f(x_j) \right)^2 + \lambda || f ||^2}{\sum_{i=1}^n f(x_i)^2}$$

Influence of λ

- $\lambda \to +\infty$: kernel PCA
  - Useful for noisy, high-dimensional data.
  - Used in spectral clustering. The graph does not play any role (unsupervised)
- $\lambda \to 0$: second smallest eigenvector of the graph
  - Useful to embed the graph in a Euclidean space (used in graph partitioning)
  - Sensitive to noise. Mapping of points outside of the graph unstable (overfitting)

Extracting successive features

- Successive features to form Φ can be obtained by:

$$f_i = \underset{f \perp \{f_1, \ldots, f_{i-1}\}}{\arg\min} \left\{ \frac{\sum_{i \sim j} \left(f(x_i) - f(x_j)\right)^2 + \lambda ||f||^2}{\sum_{i=1}^{n} f(x_i)^2} \right\}.$$

Extracting successive features

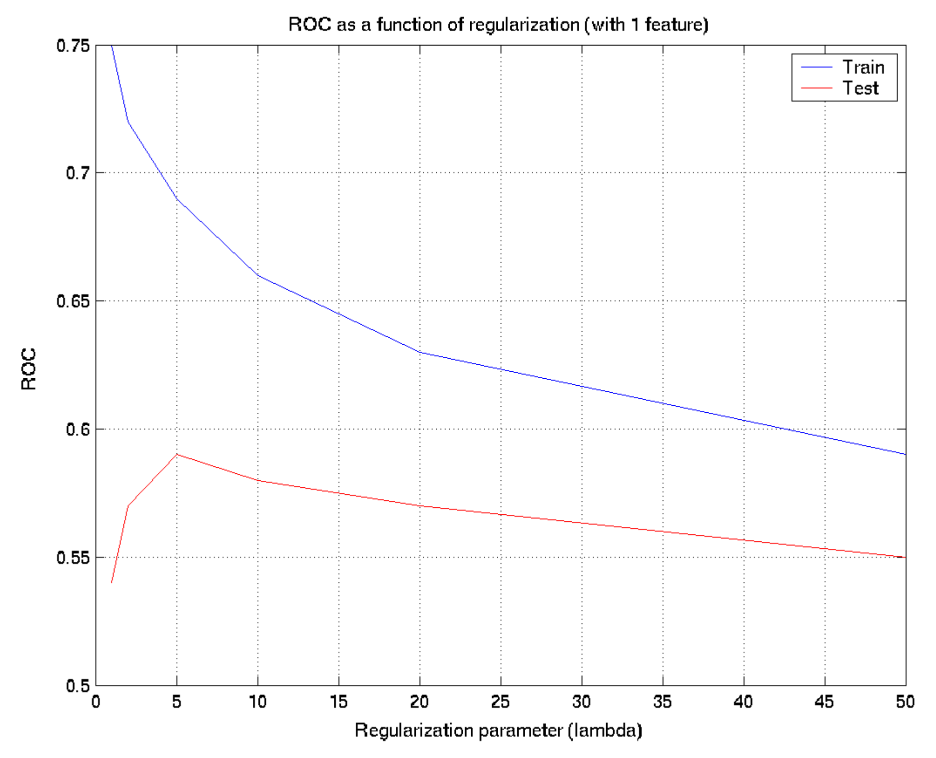
- Successive features to form Φ can be obtained by:

$$f_i = \underset{f \perp \{f_1, \ldots, f_{i-1}\}}{\arg\min} \left\{ \frac{\sum_{i \sim j} \left(f(x_i) - f(x_j)\right)^2 + \lambda ||f||^2}{\sum_{i=1}^{n} f(x_i)^2} \right\}.$$

- The solution satisfies $f_i(x) = \sum_j \alpha_{i,j} K(x_j, x)$, where α_i are the successive generalized eigenvectors of

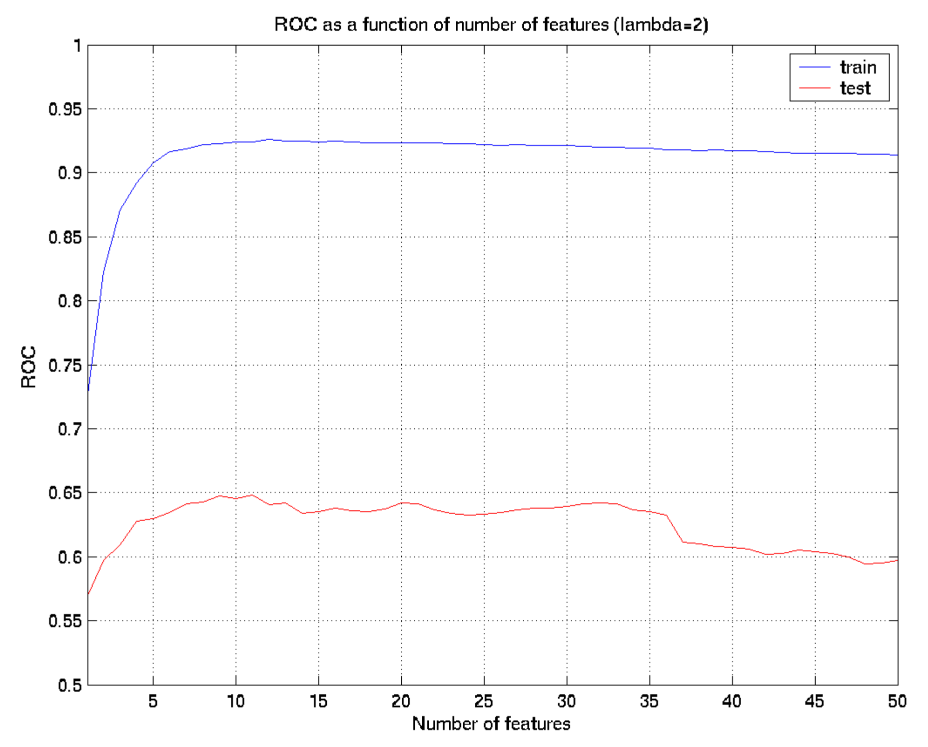
$$(LK_V + \lambda I)\alpha = \mu K_V \alpha.$$

Evaluation of the supervised approach: effect of λ



Metabolic network, 10-fold cross-validation, 1 feature

Evaluation of the supervised approach: number of features ($\lambda = 2$)

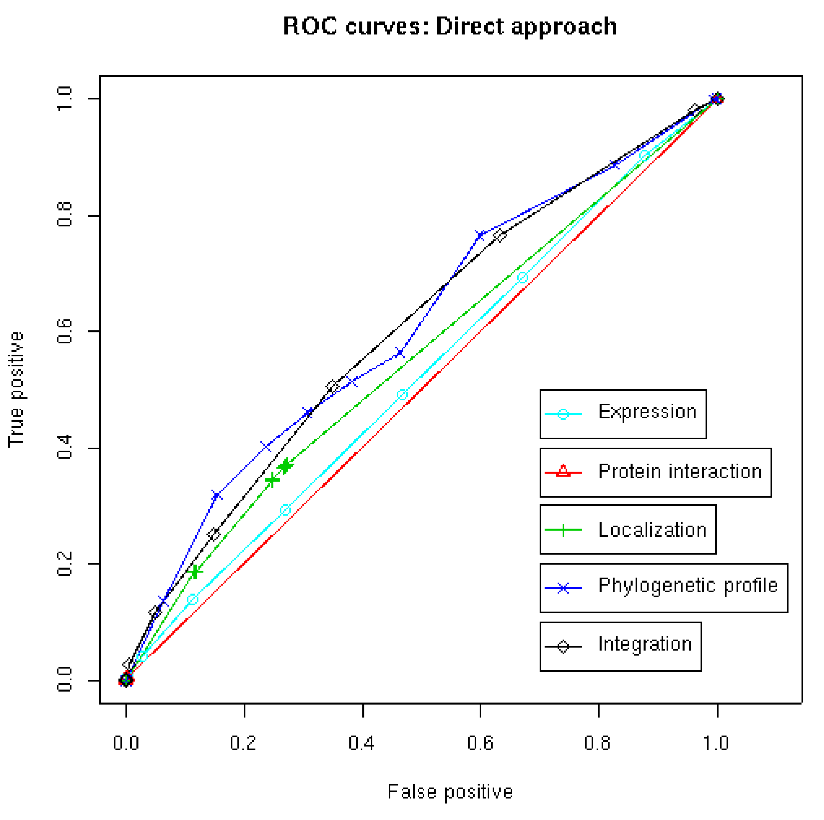


Learning from heterogeneous data

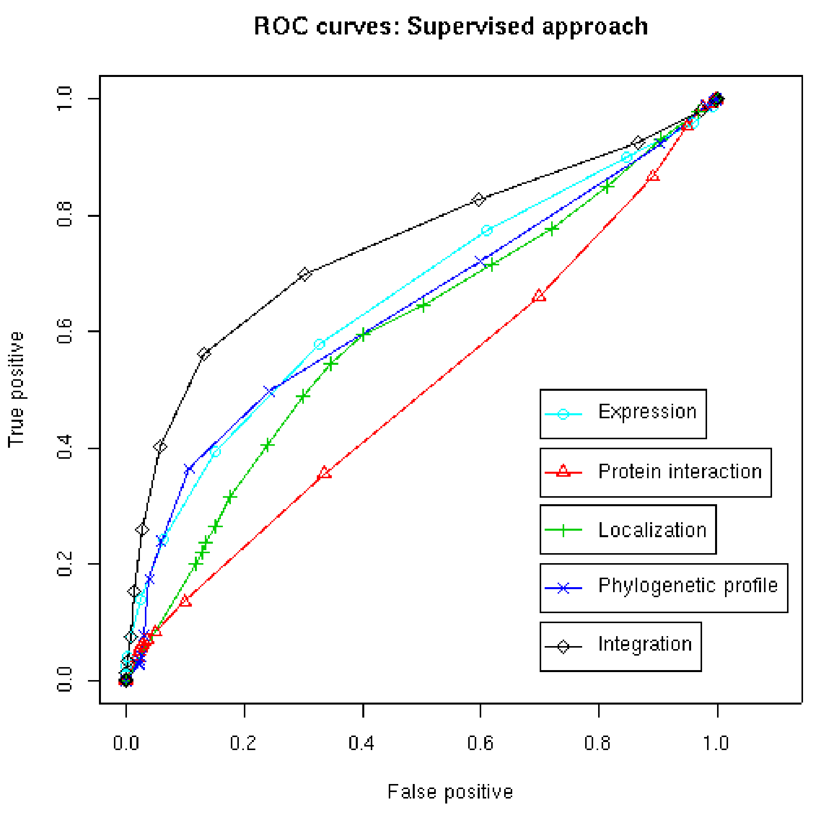
- Suppose several data are available about the genes, e.g., expression, localization, struture, predicted interaction etc...
- Each data can be represented by a positive definite similarity matrix $K_1, \ldots, K_p$ called kernels
- Kernel can be combined by various operations, e.g., addition:

$$K = \sum_{i=1}^{p} K_i$$

Learning from heterogeneous data (unsupervised)



Learning from heterogeneous data (supervised)



Conclusion

Conclusion

1. Supervised inference is better than unsupervised

Conclusion

1. Supervised inference is better than unsupervised
2. Supervised graph inference can be performed by distance metric learning

Conclusion

1. Supervised inference is better than unsupervised
2. Supervised graph inference can be performed by distance metric learning
3. Data integration with kernels is simple and powerful

See you at poster 49