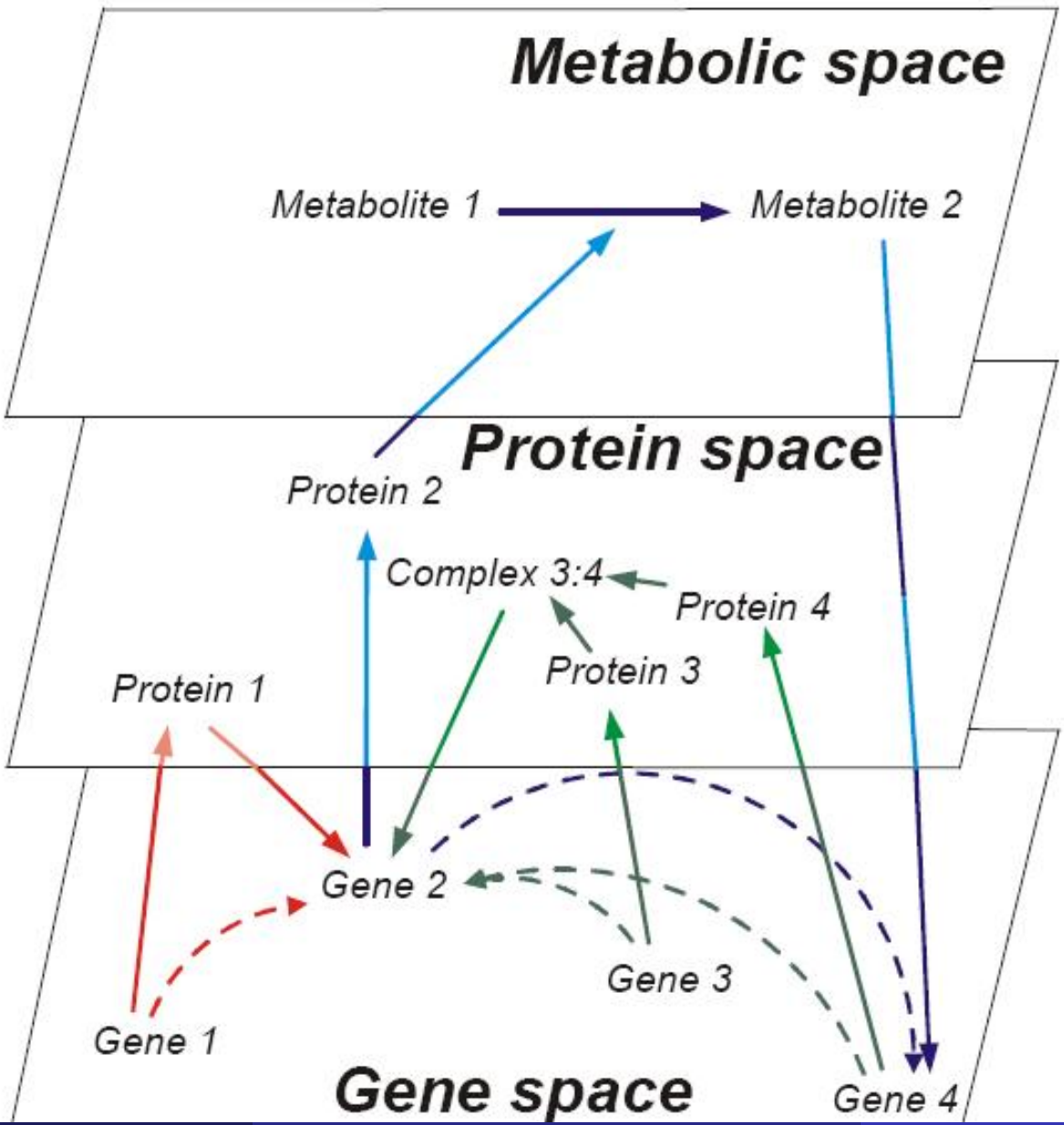
Statistical learning of biological networks: a brief overview

Florence d'Alché–Buc

IBISC CNRS, Université d'Evry, GENOPOLE, Evry, France
Email: florence.dalche@ibisc.fr

Biological networks



Motivation

- Identify and understand complex mechanisms at work in the cell
- Biological networks
  - signaling pathways
  - gene regulatory networks
  - protein-protein interaction networks
  - metabolic pathways
- Use experimental data and prior knowledge AND statistical inference to unravel biological networks and predict their behaviour

How to learn biological networks from data ?

- **Data-mining approaches** : extract co-expressed patterns and/or co-regulated patterns, reduce dimension [large scale data, often preliminary to more accurate modeling or prediction]
- **Modeling approaches** : model the network behavior, can be used to simulate and predict the network as a system [smaller scale data]
- **Predictive approaches** : predict (only) edges in an unsupervised or supervised way [large or medium scale data]

Learning (biological) networks

Data

- global data
- intervention data

Available knowledge

- edges distribution
- dynamics

labeled data, partial network

Modeling the behavior of the network
Unsupervised
Bayesian networks
Dynamical Bayesian networks
And state-space models
Ordinary Differential Equations
Stochastic Differential Equations

Predicting edges (only)
unsupervised
Gaussian Graphical models

Supervised
Decision trees, SVM,ILP
Metric and kernel learning

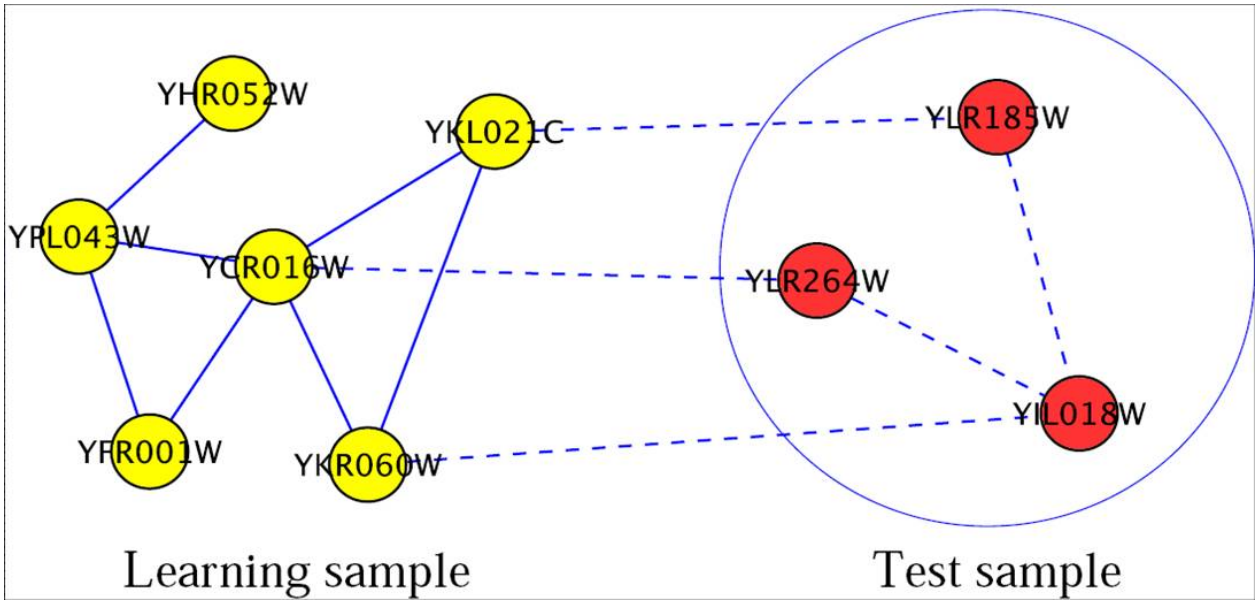
Outline

1. Introduction
2. Supervised Predictive approaches
3. Modeling approaches
4. Conclusion

Supervised learning of the concept of regulation

- **Instance Problem 1 (transcriptional regulatory networks)**:Training sample $S = \{(w_i = (v_i, v_i'), y_i), i = 1...n\}$ where w_i are **pairs** of components v_i and v_i' (think transcription factor and potential regulee) and $y_i \in Y$ indicates if there is v_i is a transcription factor for v_i'. We wish to be able to predict new regulations.
- Reference : Quian et al. 2003, Bioinformatics.
- In symbolic machine learning, this corresponds to the framework of **relational learning** classically associated with inductive logic programming (ILP) and more recently to statistical ILP :
- The predicate **interaction(X,Y)** can be learned from labeled examples

Supervised learning of interactions



- From a known network where each vertex is described by some input feature vector x, predict the edges involving new vertices described by their input feature vector

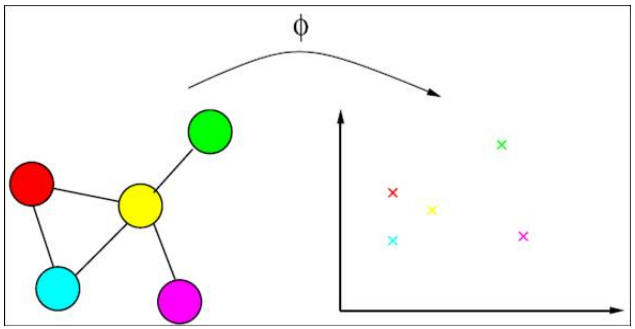
Supervised prediction of protein-protein interaction network

- **Instance Problem 2 (protein-protein interaction networks)** : Training sample $S = \{(w_i = (v_i, v_i'), y_i), i = 1...n\}$ where w_i are couples of components v_i and v_i' (think proteins) and $y_i \in Y$ indicates if there is an edge or not between v_i and v_i'. We wish to predict interactions for test and training input data
- Noble et al. in 2005 (SVM) with kernel combination
- Further studied by Biau and Bleakley 2006, Bleakley et al. 2007

Similarity or kernel learning

- In the case of non oriented graphs, a similarity between components can be learnt instead of a classification function
- Yamanishi and Vert's work (2005) first introduced this kind of approach
- We proposed a new way of formulating the problem as regression in output space endowed with a kernel(Geurts et al. 2006,2007)

A solution based on a kernelized output space 1



- For objects $v_1, ..., v_N$, let us assume we have : feature vectors $x(v_i), i = 1...N$ and a Gram matrix K defined as $K_{i,j} = k(v_i, v_j)$. The kernel k reflects the proximity between objects v, as vertices in the known graph.
- Reminder: kernel k is a positive definite (similarity) function. For such function, there exists a function ϕ called a feature map :$\mathcal{V} \rightarrow \mathcal{F}$ such that $k(v, v') = \langle \phi(v), \phi(v') \rangle$.

A solution based on a kernelized output space 2

- Use a machine learning method that can infer a function $h : \mathcal{X} \rightarrow \mathcal{F}$ to get for a given $x(v)$, an approximation of $\phi(v)$ and get an approximation $g(x(v), x(v')) = \langle h(x(v)), h(x(v')) \rangle$ of the kernel value between v and v' described by their input feature vectors $x(v)$ and $x(v')$
- Connect these two vertices if $g(x(v), x(v')) > \theta$

(by varying θ we get different tradeoffs between true positive and false positive rates)

A kernel on graph nodes

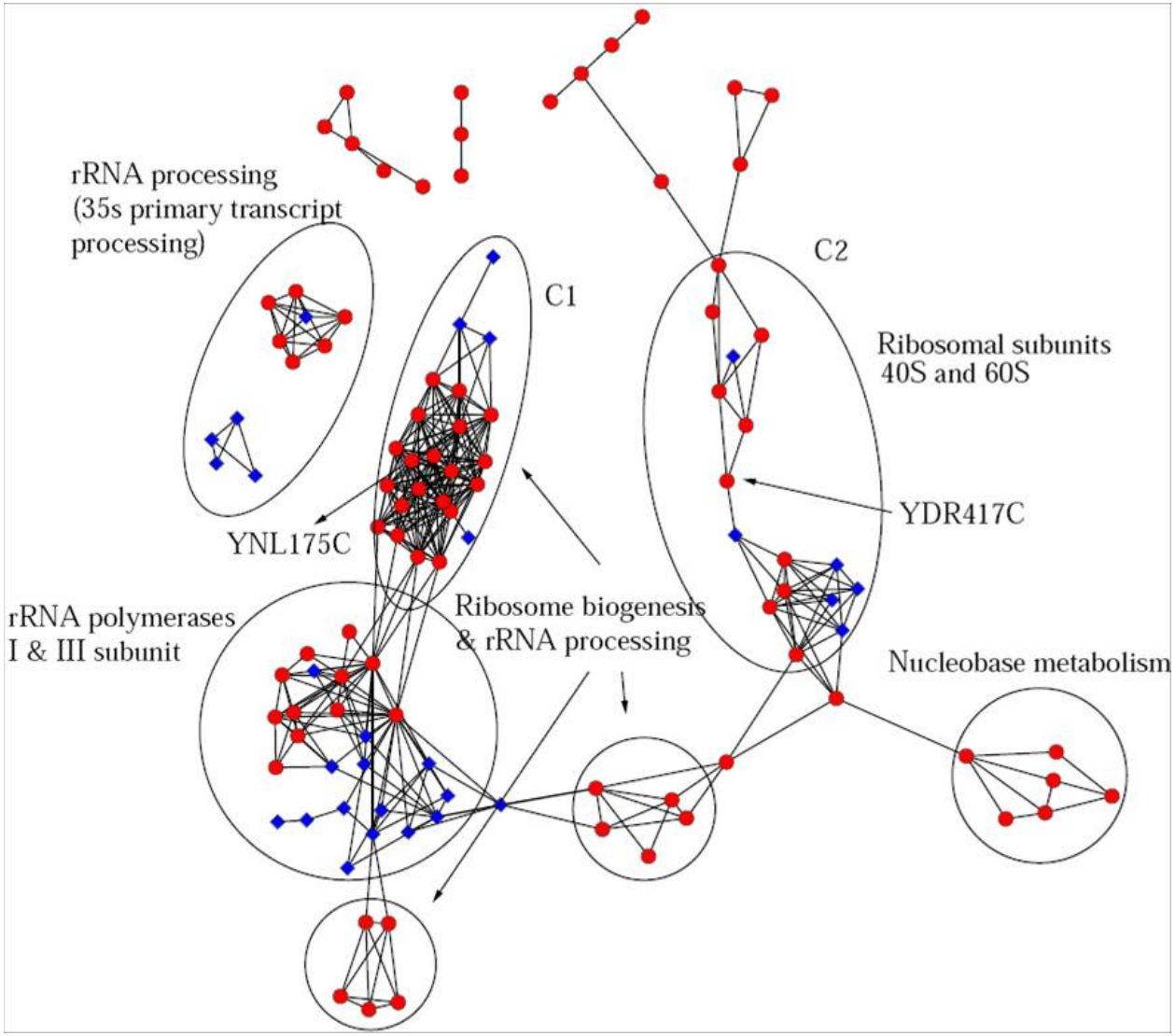
- Diffusion kernel (Kondor and Lafferty, 2002):
  The Gram matrix K with $K_{i,j} = k(v_i, v_j)$ is given by:

$$K = \exp(-\beta L)$$

  where the graph Laplacian L is defined by:

$$L_{i,j} = \begin{cases} d_i & \text{the degree of node } v_i \text{ if } i = j; \\ -1 & \text{if } y(v_i) \text{ and } y(v_j) \text{ are connected;} \\ 0 & \text{otherwise.} \end{cases}$$

Network completion and function prediction for yeast data



Challenges and limitations in supervised predictive approaches

- Semi-supervised learning or even transductive learning
- Issue : very large number of negative examples, few positive examples
- local approach (the graph is not seen as one variable)
- data (labeled examples) are not i.i.d. : regulations are interdependent

Outline

1. Introduction
2. Supervised Predictive approaches
3. Modeling approaches
4. Conclusion

Graphical models : from simple interactions models to complex ones

- Graphical Gaussian Model model estimation: estimating partial correlation as a measure of conditional independency (classified as graph prediction in my terminology)
- Bayesian networks estimation: modeling directed interactions
- Dynamic Bayesian Networks estimation: modeling directed interactions through time
- **State-space models estimation**: modeling observed and hidden dynamical processes as well

Focus on state-space models

- Goal:
  - Quantitative models (easier to learn, encompass mechanistic models : biological relevance)
  - Taking into account time
  - Some variables are not measured: assumption of an hidden process
  - Linear Gaussian models: parameters encapsulate network structure (Perrin et al. 03, Rangel et al. 04)
  - Nonlinear models (more biologically relevant): the structure is encapsulated in the form of the transition function (Nachman 04, Rogers et al. 06, Quach et al. 07)

$$\begin{aligned} \mathbf{x}(t_{k+1}) &= \mathbf{F}(\mathbf{x}(t_k), \mathbf{u}; \theta) + \epsilon_{\boldsymbol{h}}(t_k) \\ \mathbf{y}(t_k) &= \mathbf{H}(\mathbf{x}(t_k), \mathbf{u}(t_k); \theta) + \epsilon(t_k) \end{aligned}$$

System of Ordinary Differential Equations (ODE)

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t); \theta)$$

- Let us focus on gene regulatory networks
- $\mathbf{x}(t)$: state variables at time t
  - protein concentrations
  - mRNA concentrations
- $\mathbf{f}$: the form of f encodes the nature of interactions (and their structure)
  - linear/nonlinear models
  - Michaelis-Menten kinetics
  - Mass action kinetics
  - ...
- θ: parameter set (kinetic parameters, rate constants,...)
- $\mathbf{u}(t)$: input variables at time t

Reverse Engineering of Biological Networks

- An ODE model :

$$\frac{d\mathbf{x}(\mathbf{t})}{dt} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t); \theta)$$

- A partially and noisy observation model:

$$\mathbf{y}(t) = \mathbf{H}(\mathbf{x}(t), \mathbf{u}(t); \theta) + \epsilon(t)$$

  where **H** is a nonlinear observation function, $\epsilon(t)$ is a i.i.d noise
- A sequence of observed data : $\mathbf{y}_{1:K} = \{\mathbf{y}_1, ..., \mathbf{y}_K\}$ at time $t_1, t_2, ..., t_k$

Reverse Engineering of Biological Networks

- An ODE model :

$$\frac{d\mathbf{x}(\mathbf{t})}{dt} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t); \theta)$$

- A partially and noisy observation model:

$$\mathbf{y}(t) = \mathbf{H}(\mathbf{x}(t), \mathbf{u}(t); \theta) + \epsilon(t)$$

  where **H** is a nonlinear observation function, $\epsilon(t)$ is a i.i.d noise
- A sequence of observed data : $\mathbf{y}_{1:K} = \{\mathbf{y}_1, ..., \mathbf{y}_K\}$ at time $t_1, t_2, ..., t_k$

- Structure estimation
- Parameters estimation θ
- States estimation $\mathbf{x}(t)$

Structure learning

- **Case 1**: a very few variables involved, then a combinatorial search for structure can be processed. For each potential structure, estimation of parameters has to be carried on
- **Case 2**: more than a tens of variables are involved, then it is worth using an algorithm dedicated to structure learning. Structure learning in nonlinear dynamical models as well as in static Bayesian networks can be solved by stochastic exploration of the candidates (huge) set using an appropriate criterion that take into account data and parameters estimation, given the candidate structure. MCMC methods, evolutionary approaches are used.
- In the following, we assume that the network structure is given

An example of Nonlinear State-Space Model

$$\begin{aligned} \frac{d\mathbf{x}(\mathbf{t})}{dt} &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t); \theta) \\ \mathbf{y}(t) &= \mathbf{H}(\mathbf{x}(t), \mathbf{u}(t); \theta) + \epsilon(t) \end{aligned}$$

$$\begin{aligned} \mathbf{x}(t_{k+1}) &= \mathbf{F}(\mathbf{x}(t_k), \mathbf{u}; \theta) \\ \mathbf{y}(t_k) &= \mathbf{H}(\mathbf{x}(t_k), \mathbf{u}(t_k); \theta) + \epsilon(t_k) \end{aligned}$$

with

$$\mathbf{F}(\mathbf{x}(t_k), \mathbf{u}; \theta) = \mathbf{x}(t_k) + \int_{t_k}^{t_{k+1}} \mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau); \theta) d\tau$$

Bayesian inference

- Prior distribution over the initial state and parameters: $p(\mathbf{x}_1, \theta)$
- A state transition model: $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \theta)$
- An observation model: $p(\mathbf{y}_k|\mathbf{x}_k, \theta)$
- A sequence of observations: $\mathbf{y}_{1:K} = \{\mathbf{y}_1, ..., \mathbf{y}_K\}$

Bayesian inference

- Prior distribution over the initial state and parameters: $p(\mathbf{x}_1, \theta)$
- A state transition model: $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \theta)$
- An observation model: $p(\mathbf{y}_k|\mathbf{x}_k, \theta)$
- A sequence of observations: $\mathbf{y}_{1:K} = \{\mathbf{y}_1, ..., \mathbf{y}_K\}$

- Focus on the filtering distribution: $p(\mathbf{x}_k, \theta|\mathbf{y}_{1:k})$
- Tool: Unscented Kalman Filter to deal with nonlinearities (Quach et al., 2007)

Challenges in (dynamical) modeling approaches

- identifiability
- **coupled systems** : metabolic and regulatory networks, protein-protein interactions and regulatory network
- scale (large networks)
- non stationnarity
- incorporate other components : space, cellular compartments ...
- MORE DATA, make repository benchmarks, challenges

General conclusion and perspective

- Different views of the learning problem, different scales, different prior knowledge
- Some of these methods could be linked to participate to the same discovery process
- Need for building data repository and demand for biological validation

C. Auliac, V. Frouin, X. Gidrol, F. d'Alché-Buc, Evolutionary Approaches for the Reverse-Engineering of Gene Regulatory Networks: A Study on a Realistic Biological Dataset, accepté à BMC Bioinformatics, à paraître en 2008.
P. Geurts, N. Touleimat, M. Dutreix, F. d'Alché-Buc, Inferring biological networks with output kernel trees, BMC Bioinformatics, to appear, May 3, 2007.
Kato, K. Tsuda, EM based algorithm for kernel matrix completion, Bioinformatics, vol. 21, 2005.
B.-E. Perrin, L. Ralaivola,A. Mazurie, S. Bottani, J. Mallet, F. d'Alché-Buc, Inference of gene regulatory network with Dynamic Bayesian Network, Bioinformatics (Oxford Press), vol. 19, pi38-49,2003.
M. Quach, N.Brunel, F. d'Alché-Buc, Estimating parameters and hidden variables in nonlinear state-space models based on ODEs for biological networks inference, November, 23:3209-3216, 2007.
Y. Yamanishi, Y., J.-P. Vert and Kanehisa, M. Supervised enzyme network inference from the integration of genomic data and chemical information,Bioinformatics, vol. 21,2005.