Machine learning methods for gene/protein function prediction

Giorgio Valentini
valentini@di.unimi.it

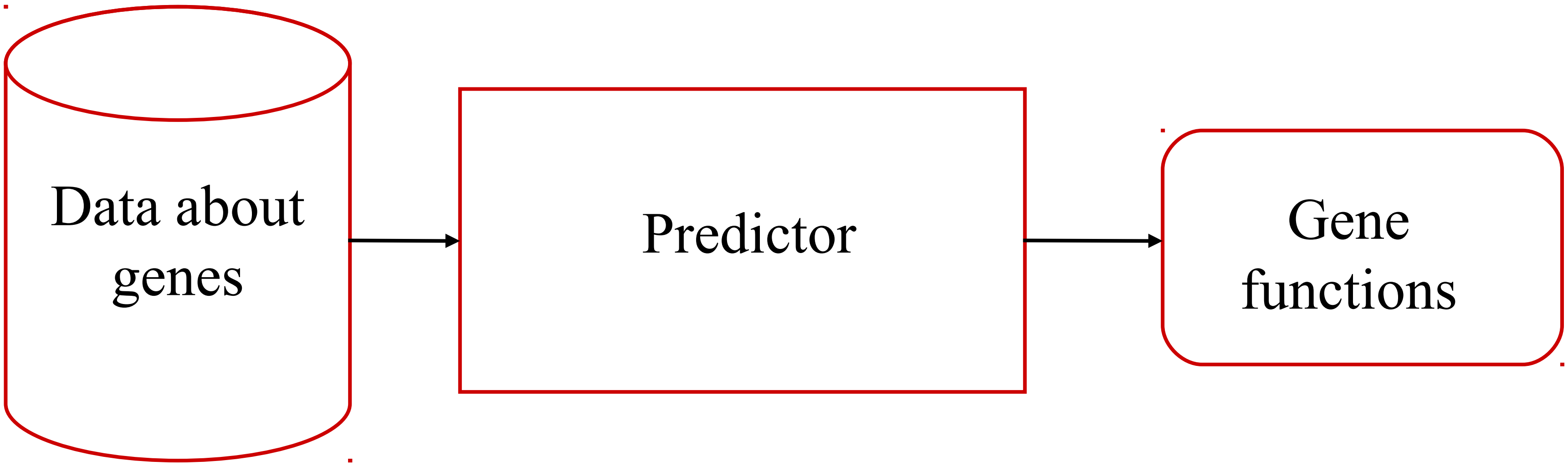
DI - Dipartimento di Informatica

Università degli Studi di Milano

Outline

- Gene Function Prediction (GFP)
- Gene Ontology and FunCat
- Computational approaches to GFP
- Hierarchical Ensemble methods for GFP
- Two examples of Hierarchical ensembles:
  - A Bayesian approach (Barutcouglu et al, 2006)
  - True Path Rule ensembles (Valentini, 2011)

Gene function prediction

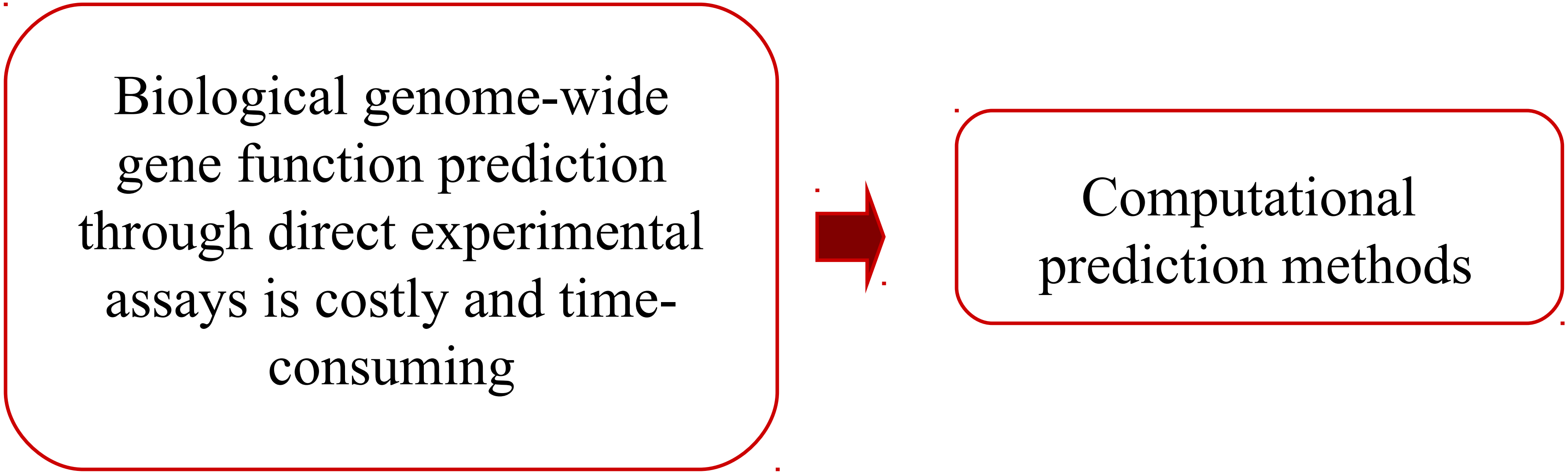


Gene function prediction can be formalized as a supervised machine learning problem

Motivation

- Novel high-throughput biotechnologies accumulated a wealth of data about genes and gene products
- Manual annotation of gene function is time consuming and expensive and becomes infeasible for growing amount of data.
- For most species the functions of several genes are unknown or only partially known: “in silico” methodsrepresent a fundamental tool for gene function prediction at genome-wide and ontology-wide level (*Friedberg*, 2006).
- Computational analysis provide predictions that can be considered hypotheses to drive the biological validation of gene function (*Pena-Castillo et al*. 2008).

Computational prediction supports biological gene function prediction



Biological genome-wide gene function prediction through direct experimental assays is costly and time-consuming

Computational prediction methods

Computational prediction methods assist the biologist to:

- Suggest a restricted set of candidate functions that can be experimentally verified
- Directly generate new hypotheses
- Guide the exploration of promising hypotheses

Characteristics of the Gene Function Prediction (GFP) problem

- Large number of functional classes: hundreds (FunCat) or thousands (Gene Ontology (GO)) : large multi-class classification
- Multiple annotations for each gene: multilabel classification
- Different level of evidence for functional annotations: labels at different level of reliability
- Hierarchical relationships between functional classes (tree forest for FunCat, direct acyclic graph for GO): hierarchical relationships between classes (structured output)
- Class frequencies are unbalanced, with positive examples usually largely lower than negatives: unbalanced classification
- The notion of "negative example" is not univocally determined: different strategies to choose negative examples
- Multiple sources of data available: each type captures specific functional characteristics of genes/gene products: multi-source classification
- Data are usually complex (e.g. high-dimensional) and noisy: classification with complex and noisy data

Taxonomies of gene function

1. *Gene Ontology* (GO)

   http://www.geneontology.org/

   Fine grained: classes structured according to a directed acyclic graph

2. *Functional Catalogue (FunCat)*

   http://www.helmholtz-muenchen.de/en/mips/projects/funcat/

   Coarse grained: classes structured according to a tree

The Gene Ontology

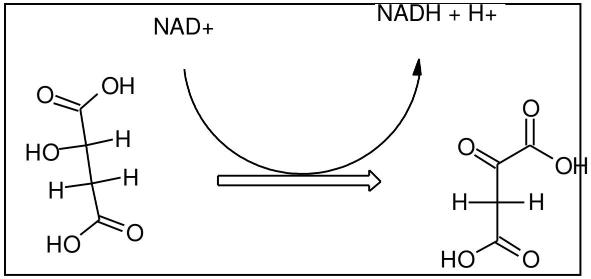
The Gene Ontology (GO) project began as a collaboration between three model organism databases, FlyBase (*Drosophila*), the *Saccharomyces* Genome Database (SGD) and the Mouse Genome Database (MGD), in 1998. Now it includes several of the world's major repositories for plant, animal and microbial genomes.

The GO project has developed three structured controlled vocabularies (ontologies) that describe gene products in terms of their associated biological processes, cellular components and molecular functions in a species-independent manner

The Gene Ontology (GO) is actually three Ontologies

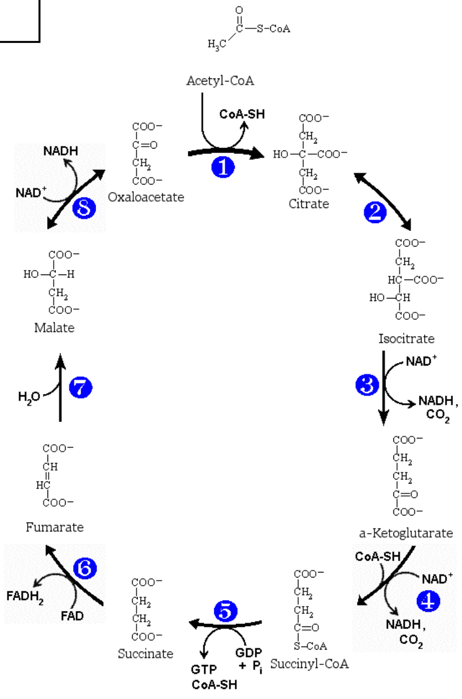
1) Molecular Function

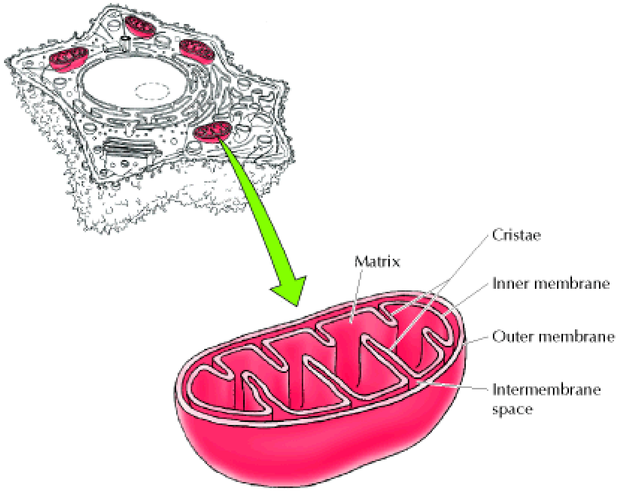
GO term: Malate dehydrogenase activity
GO id: GO:0030060
(S)-malate + NAD(+) = oxaloacetate + NADH.



2) Biological Process

GO term: tricarboxylic acid cycle
Synonym: Krebs cycle
Synonym: citric acid cycle
GO id: GO:0006099





3) Cellular Component

GO term: mitochondrion
GO id: GO:0005739
Definition: A semiautonomous, self replicating organelle that occurs in varying numbers, shapes, and sizes in the cytoplasm of virtually all eukaryotic cells. It is notably the site of tissue respiration.

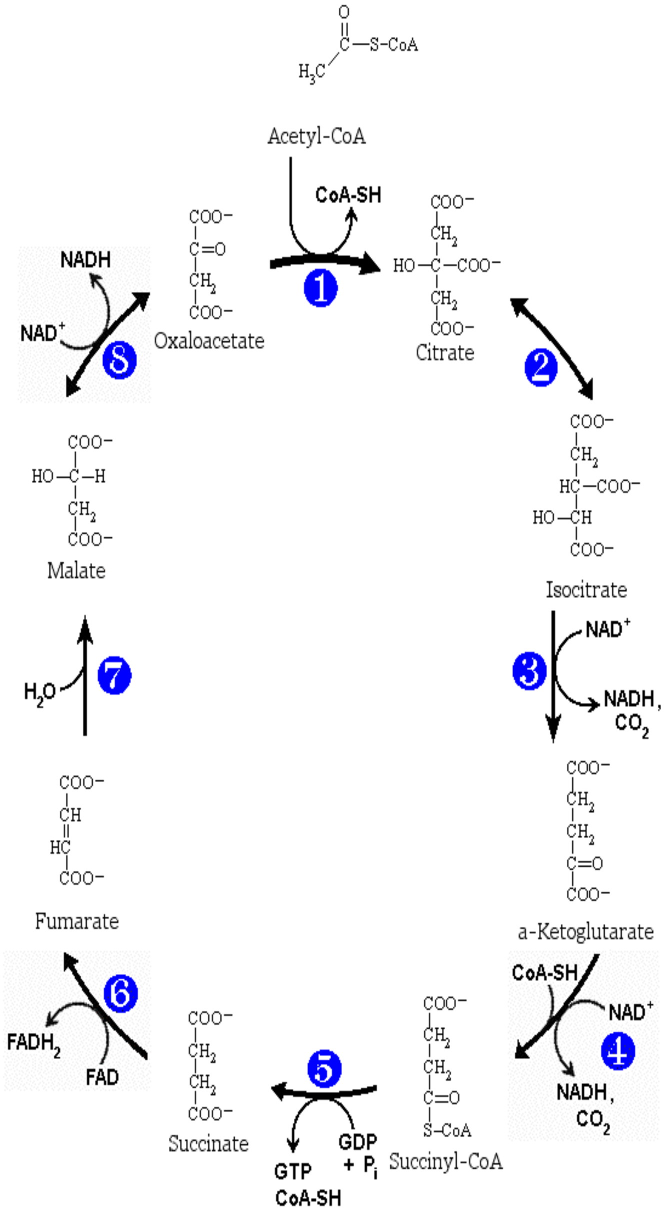
(Slide downloaded from www.geneontology.org)

GO term: tricarboxylic acid cycle

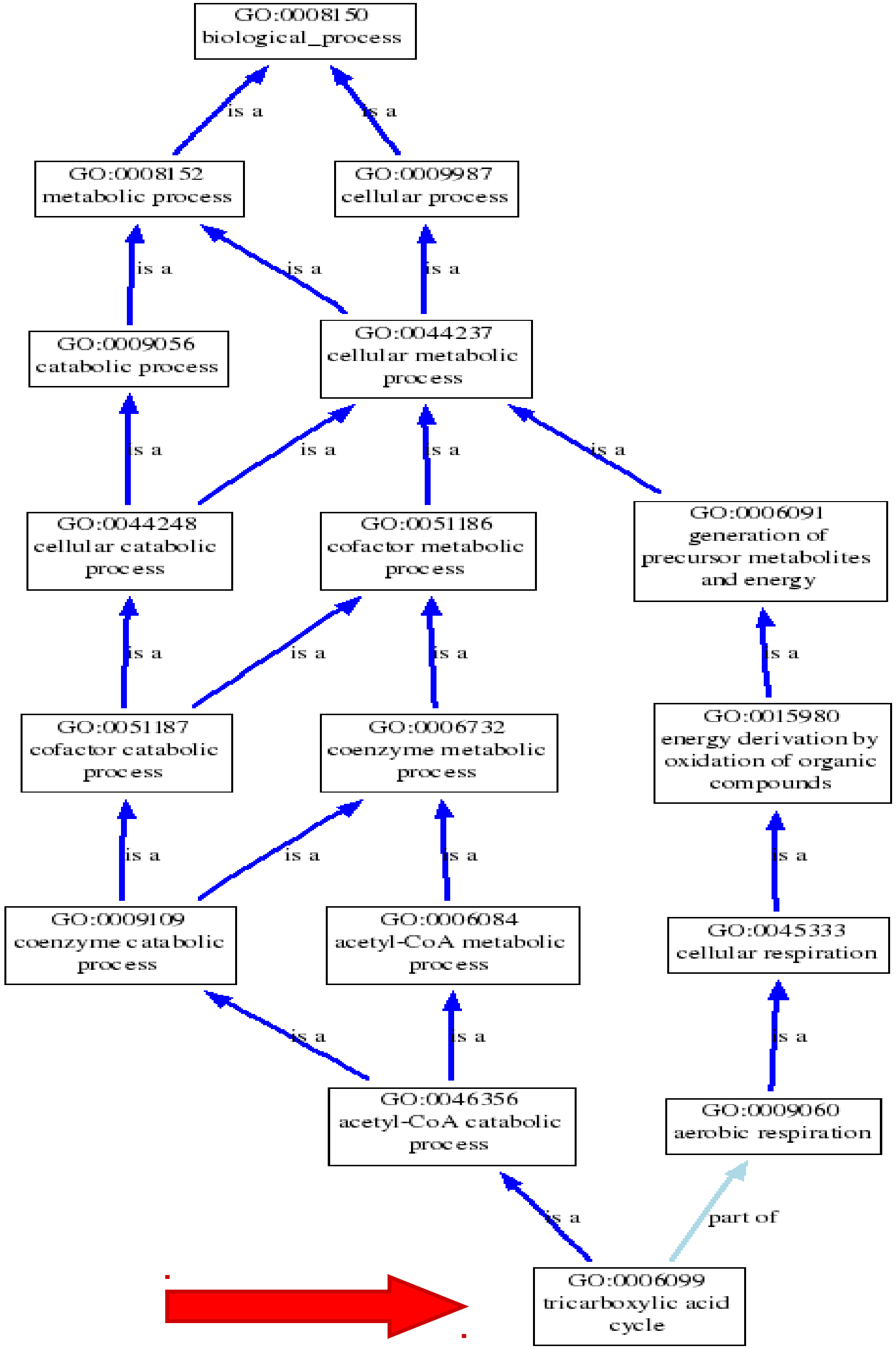
GO Accession : GO:0006099
Ontology : Biological Process

Definition

A nearly universal metabolic pathway in which the acetyl group of acetyl coenzyme A is effectively oxidized to two CO2 and four pairs of electrons are transferred to coenzymes. The acetyl group combines with oxaloacetate to form citrate, which undergoes successive transformations to isocitrate, 2-oxoglutarate, succinyl-CoA, succinate, fumarate, malate, and oxaloacetate again, thus completing the cycle. In eukaryotes the tricarboxylic acid is confined to the mitochondria.



998 annotated gene products



Relationships between GO terms are structured according to a DAG

Relationships between terms in the GO

The ontologies of GO are structured as a directed acyclic graph *(DAG)* $G=<V,E>$

$V = \{t \mid \text{terms of the GO}\}$ $E= \{(t, u) \mid t \, \varepsilon \, V \text{ and } t \, \varepsilon \, V\}$

Relations between GO terms are also categorized and defined:

- *is a* (subtype relations)
- *part of* (part-whole relations)
- *regulates* (control relations)

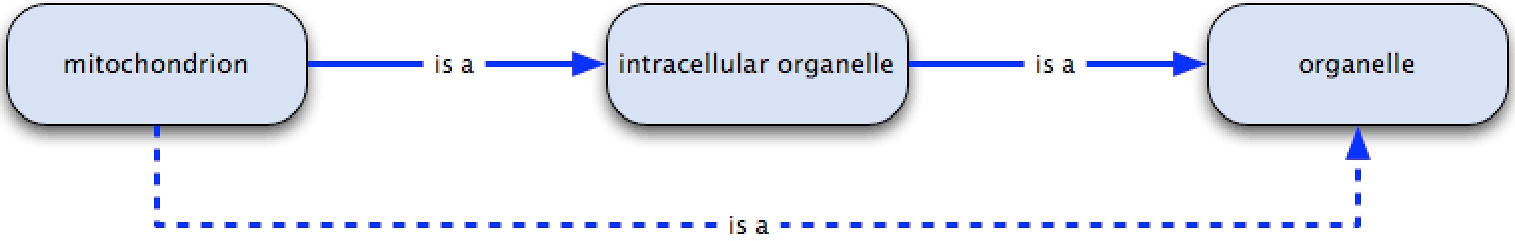
Is a relation

If we say A is a B, we mean that node A is a subtype of node B.

For example, mitotic cell cycle is a cell cycle, or lyase activity is a catalytic activity.

The is a relation is transitive, which means that if A is a B, and B is a C, we can infer that A is a C.

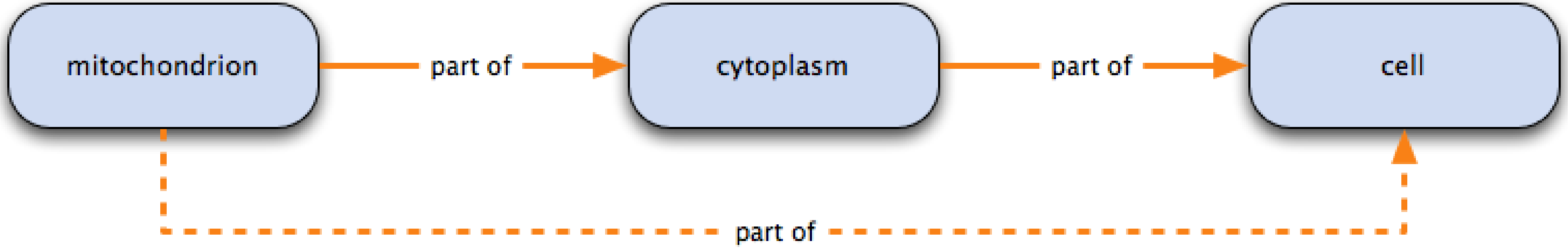
E.g.:



Part of relation

The relation part of represents part-whole relationships in the GO.

The part of relation is transitive:

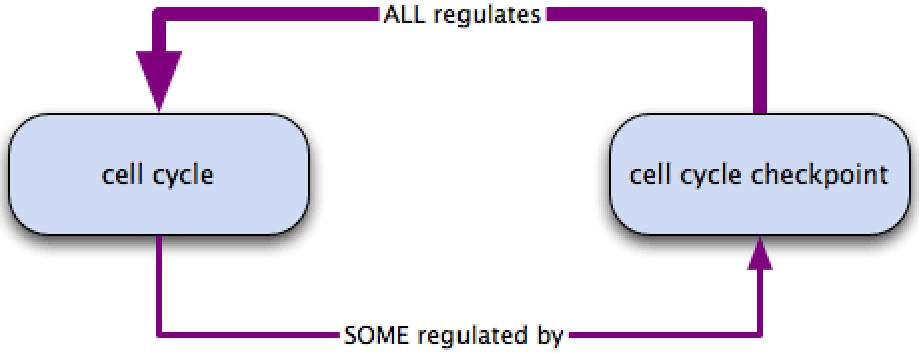


Regulates relation

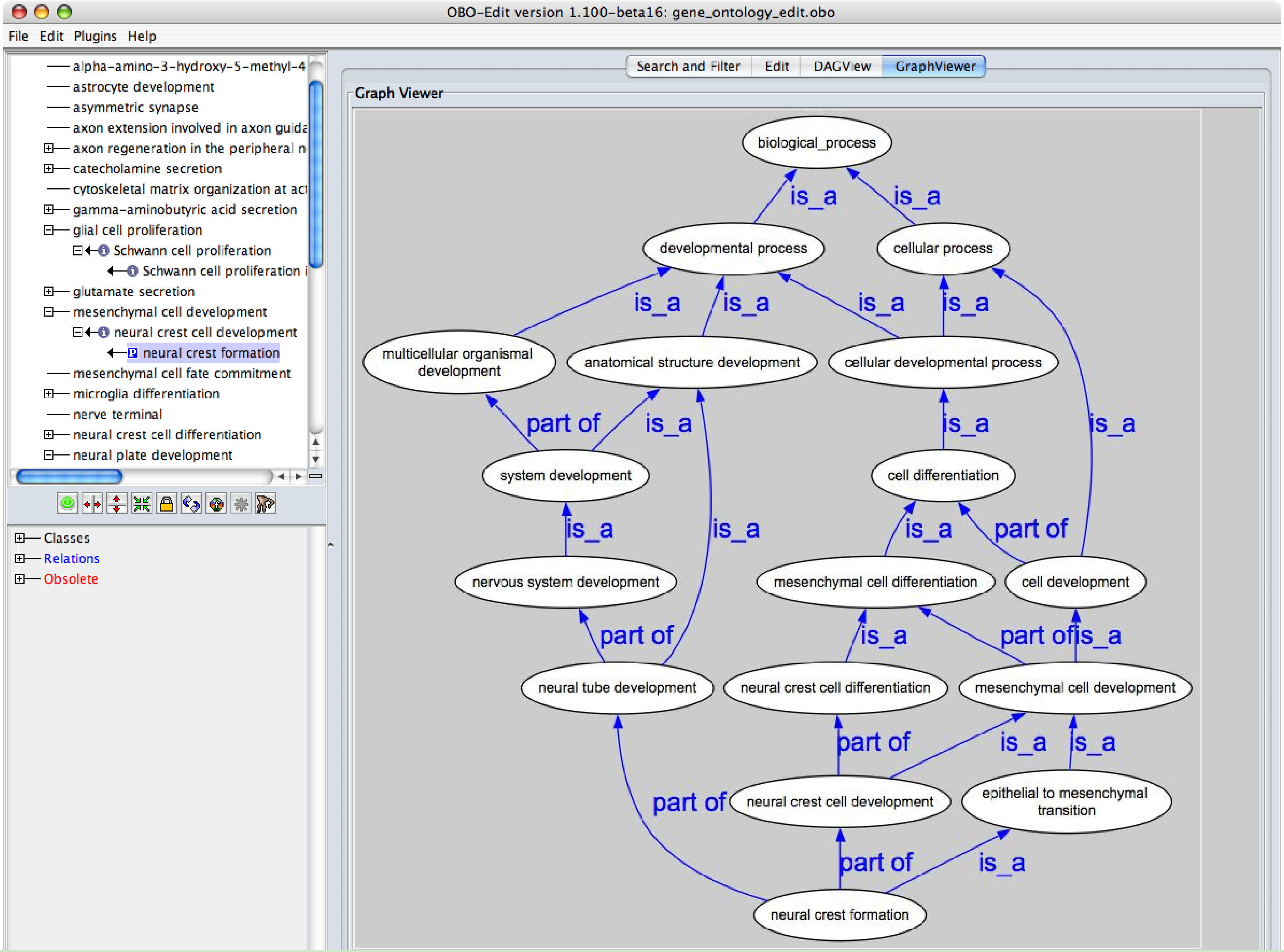
If we say that A regulates B we mean that A directly affects the manifestation of B, i.e. the former regulates the latter.

For example, the target of the regulation may be another process—for example, regulation of a pathway or an enzymatic reaction—or it may be a quality, such as cell size or pH.

Analogously to part of, this relation is used specifically to mean necessarily regulates:

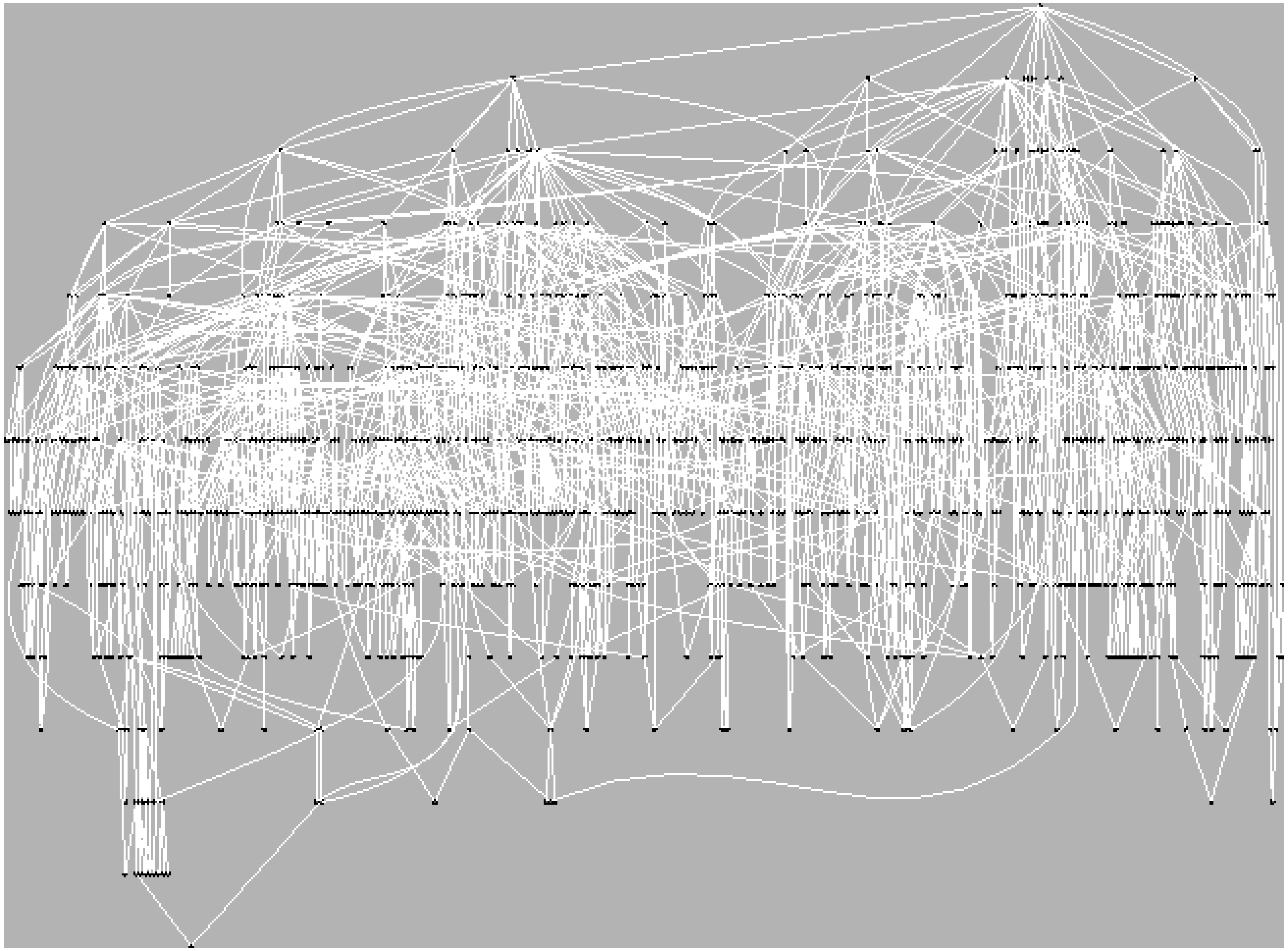


In general regulates is not transitive


OBO-Edit version 1.100-beta16: gene_ontology_edit.obo
File
Edit
Plugins
Help
alpha-amino-3-hydroxy-5-methyl-4
astrocyte development
asymmetric synapse
axon extension involved in axon guida
axon regeneration in the peripheral n
catecholamine secretion
cytoskeletal matrix organization at act
gamma-aminobutyric acid secretion
glial cell proliferation
Schwann cell proliferation
Schwann cell proliferation i
glutamate secretion
mesenchymal cell development
neural crest cell development
neural crest formation
mesenchymal cell fate commitment
microglia differentiation
nerve terminal
neural crest cell differentiation
neural plate development
Classes
Relations
Obsolete
Search and Filter
Edit
DAGView
GraphViewer
Graph Viewer
biological_process
is_a
is_a
developmental process
cellular process
is_a
is_a
is_a
is_a
multicellular organismal development
anatomical structure development
cellular developmental process
part of
is_a
is_a
is_a
system development
cell differentiation
is_a
is_a
is_a
part of
nervous system development
mesenchymal cell differentiation
cell development
part of
is_a
part ofis_a
neural tube development
neural crest cell differentiation
mesenchymal cell development
part of
is_a
is_a
part of
neural crest cell development
epithelial to mesenchymal transition
part of
is_a
neural crest formation

A visualization of the GO DAG trough OBO-Edit

GO DAG of the BP ontology *(S. cerevisiae)*



1074 GO classes (nodes) connected by 1804 edges

Graph realized through *HCGene* (Valentini, Cesa-Bianchi, *Bioinformatics* 24(5), 2008)

Evidence codes

Evidence codes indicate how the annotation to a particular term is supported:

Experimental Evidence Codes:

an experimental assay has been used for the annotation

Author statement codes:

indicate that the annotation was made on the basis of a statement made by the author(s) in the reference cited.

Curatorial evidence codes:

annotations inferred by a curator from other GO annotations

Computational analysis evidence codes:

based on an *in silico* analyses manually reviewed

Automatically-assigned Evidence Codes :

based on an *in silico* analyses not manually reviewed

Groups of evidence codes

Experimental Evidence Codes

EXP: Inferred from Experiment

IDA: Inferred from Direct Assay

IPI: Inferred from Physical Interaction

IMP: Inferred from Mutant Phenotype

IGI: Inferred from Genetic Interaction

IEP: Inferred from Expression Pattern

Author Statement Evidence Codes

TAS: Traceable Author Statement

NAS: Non-traceable Author Statement

Curator Statement Evidence Codes

IC: Inferred by Curator

ND: No biological Data available

Computational Analysis Evidence Codes

ISS: Inferred from Sequence or Structural Similarity

ISO: Inferred from Sequence Orthology

ISA: Inferred from Sequence Alignment

ISM: Inferred from Sequence Model

IGC: Inferred from Genomic Context

RCA: inferred from Reviewed Computational Analysis

Automatically-assigned Evidence Codes

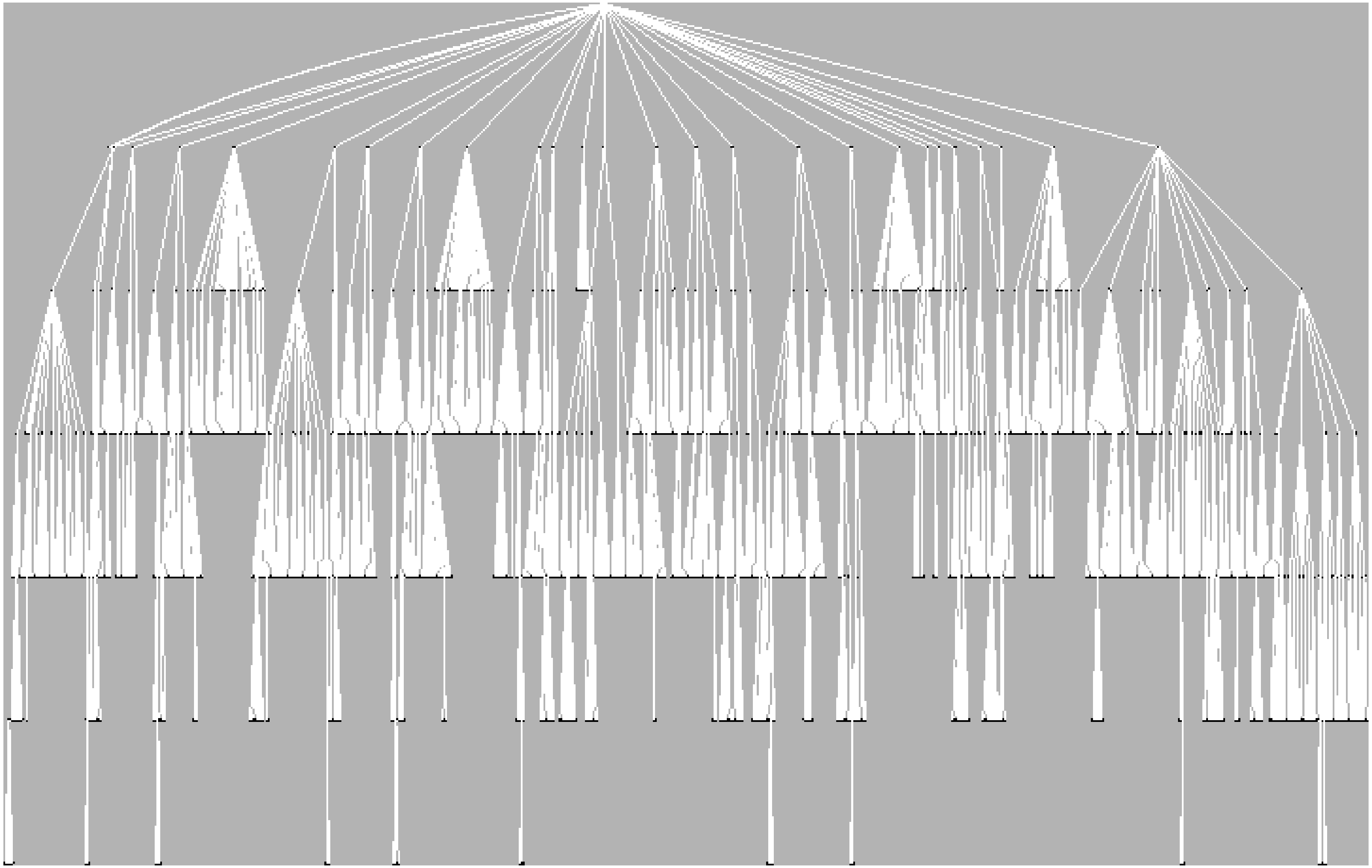
IEA: Inferred from Electronic Annotation

Obsolete Evidence Codes

NR: Not Recorded

The Functional Catalogue (FunCat)

http://www.helmholtz-muenchen.de/en/mips/projects/funcat



The Functional Catalogue (FunCat)

http://www.helmholtz-muenchen.de/en/mips/projects/funcat

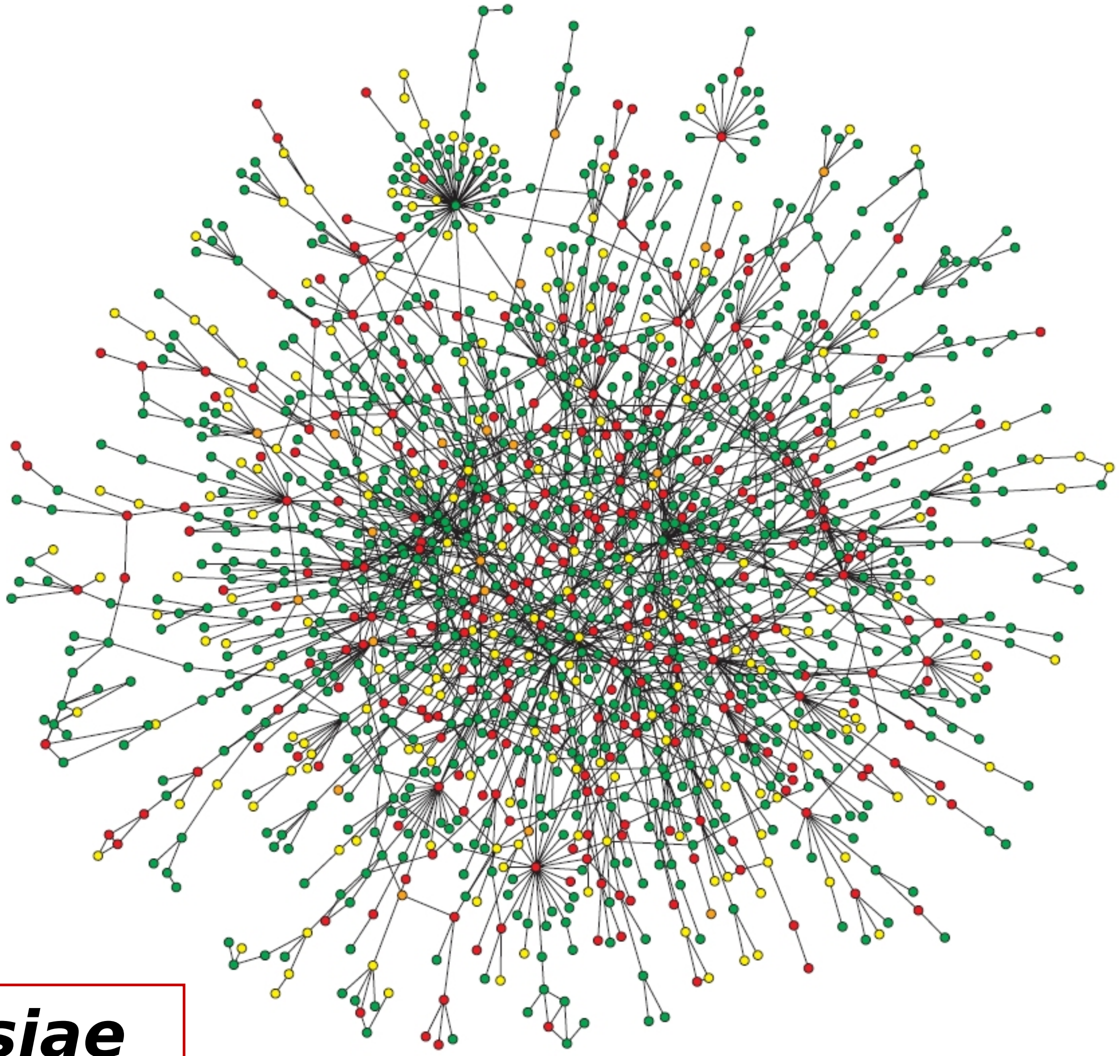
- The *Functional Catalogue* is an annotation scheme for the functional description of proteins of prokaryotic and eukaryotic origin
- Hierarchical tree like structure.
- Up to six levels of increasing specificity. FunCat version 2.1 includes 1362 functional categories.
- FunCat descriptive, but compact: classifies protein functions not down to the most specific level.
- Comparable to parts of the 'Molecular Function' and 'Biological Process' terms of the GO system.
- More compact and stable than GO, focuses on the functional process not describing the molecular function on the atomic level

Computational approaches to GFP

A very schematic taxonomy of computational GFP methods:

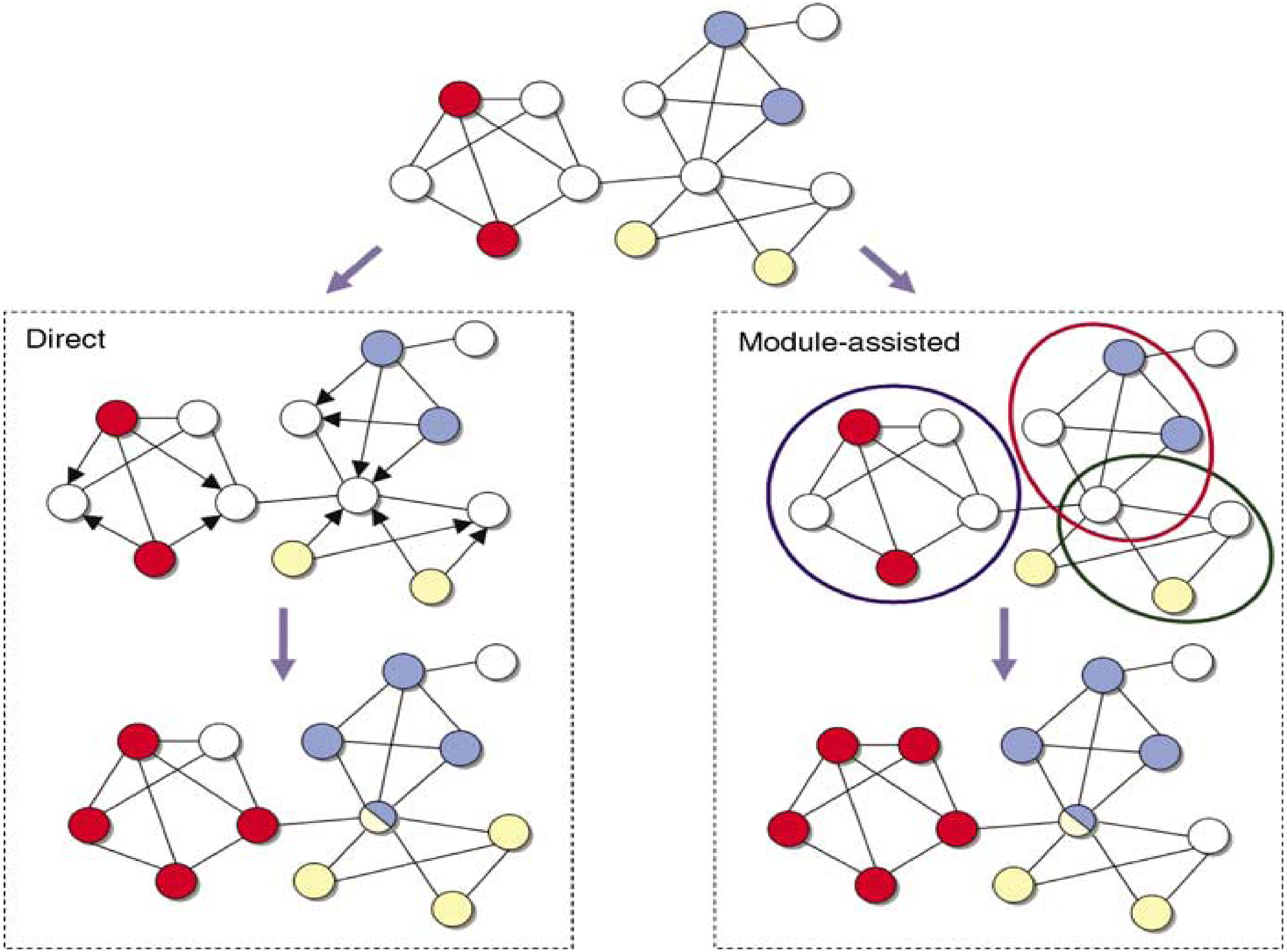
- Inference and *annotation transfer through sequence similarity* (BLAST)
- *Network-based* methods
- *Kernel methods* for structured output spaces
- *Hierarchical ensemble methods*

Biological networks



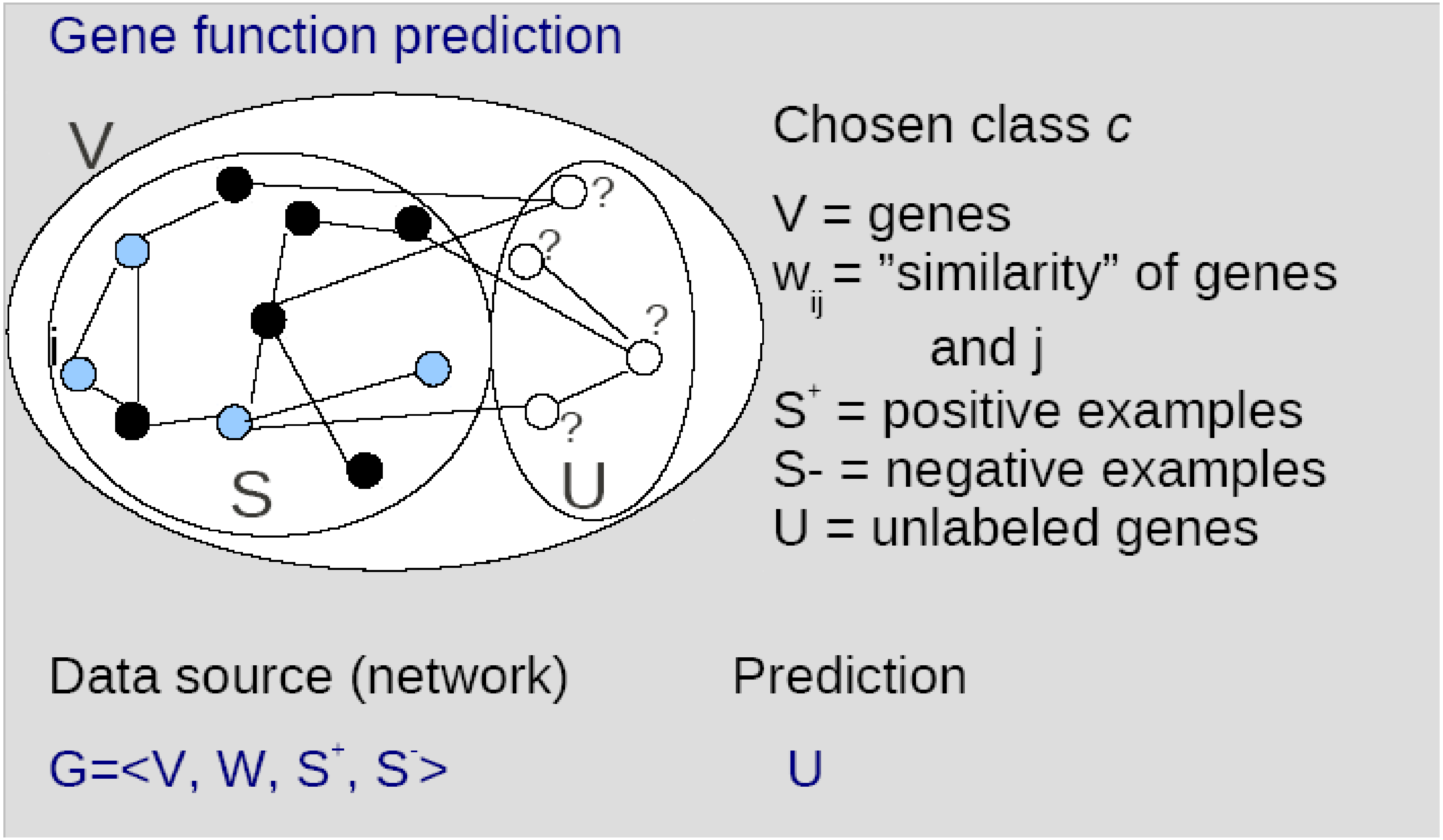
S. Cerevisiae
4389 proteins
14319 interactions

A network-based approach



From: Sharan et al. Mol. Sys. Biol. 2007

Network based methods: predicting a specific functional term



Gene function prediction

Chosen class *c*

V = genes

w_{ij} = "similarity" of genes

and j

S^+ = positive examples

S- = negative examples

U = unlabeled genes

Data source (network)

Prediction

$G=<V, W, S^+, S^->$

U

Network-based methods

Several available methods:

- *Guilt by association* (Marcotte *et al.* 1999, Oliver et al. 2000)
- *Label propagation* (Zhu and Ghahramani, 2003, Zhou et al. 2004)
- *Markov random walks* (Szummer and Jaakkola, 2002, Azran et al 2007)
- *Markov random fields* (Deng et al. 2004)
- *Graph regularization techniques* (Belkin et al. 2004, Dellaleu et al 2005)
- *Gaussian random fields* (Tsuda et al. 2005, Mostafavi et al. 2010)
- *Hopfield networks* (Karaoz et al. 2004, Bertoni et al. 2011, Frasca et al. 2015)

These different approaches *minimize a similar quadratic criterion* to improve:

a) Consistency of the initial labeling

b) Topological consistency of the data

They exploit different types of relational data: physical and genetic interactions, similarities between protein domains or motifs, structural and sequence homologies, correlations between expression profiels, …

-→ need for **network integration algorithms**

Kernel methods

Kernel methods are largely applied to classification problems:

1. Obtaining a non-linear classifier, through a non-linear mapping into the feature space, using an algorithm designed for linear discrimination :

$$f(x) = w^T \phi(x)$$

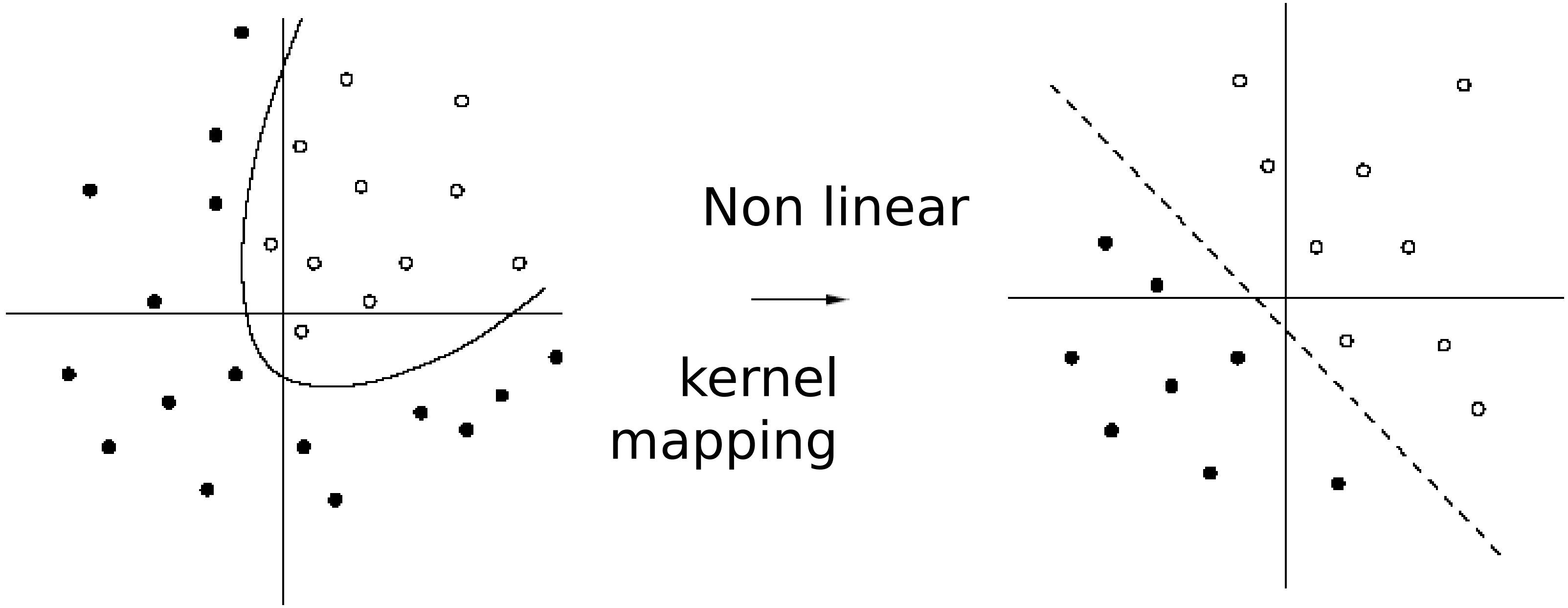
2. Whenever **w** can be expressed as a weighted sum over the images of the input examples:

$$w = \sum_i \alpha_i \phi(x_i) \Rightarrow f(x) = \sum_i \alpha_i \phi(x_i)^T \phi(x)$$

3. The discriminant function can be expressed through a suitable kernel function:

$$f(x) = \sum_i \alpha_i K(x_i, x)$$

Kernel metods for binary classification problems



Non linear kernel mapping

Original input space $\xrightarrow{\phi}$ Transformed feature space

Kernel methods for structured output spaces

A binary classier can predict whether a protein performs a certain function:

$$f : X \rightarrow Y_i \qquad Y_i = \{0,1\} \qquad 1 \leq i \leq k$$

How to predict the full hierarchical annotation $y = \{y_1, y_2, \ldots, y_k\}$?

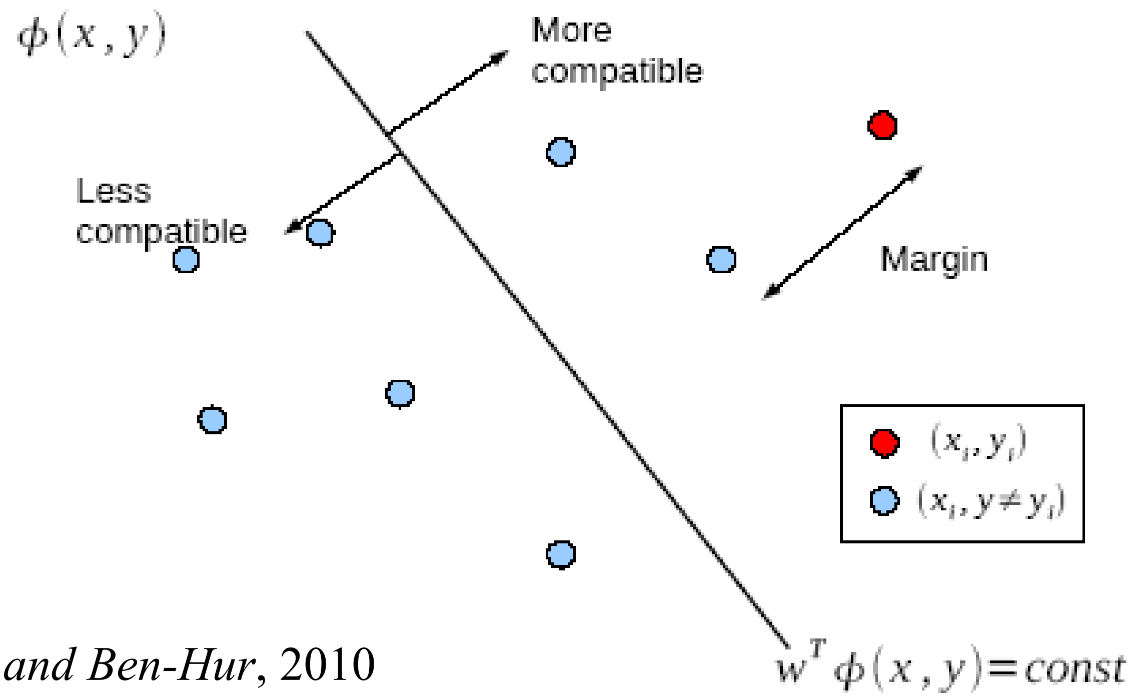
The main idea: using a kernel for structured output, that is a function:

$$f : X \times Y \rightarrow \Re$$

This classification rule chooses the label **y** that is most compatible with an input *x*.

Whereas in two-class classification problems the kernel depends *only on the input* (proteins), in the structured-output setting it is a *joint function of inputs and outputs* (set of the labels)

Kernel methods for structured output spaces: a geometric view



From: *Sokolov and Ben-Hur*, 2010

The classifier is assumed to be linear in the joint input-output feature space:

$$f(x,y|w) = w^T \phi(x,y)$$

Structured output kernel methods for gene function prediction

- *Sokolov and Ben-Hur* (2010): a structured Perceptron,

and a variant of the structured support vector machine (*Tsochantaridis et al*. 2005), applied to the the prediction of GO terms in mouse and other model organisms

- *Astikainen et al. (*2008) and *Rousu et al. (*2006): Structured output maximum-margin algorithms applied to the tree-structured prediction of enzyme functions

Hierarchical ensemble methods: the next lecture ...