Hierarchical Ensemble Methods for Ontology-based Prediction in Computational Biology

Computer Science Department

UNIVERSITÀ DEGLI STUDI DI MILANO

AnacletoLAB

Computational Biology and Bioinformatics

Marco Notaro

https://marconotaro.github.io

Bioinformatics vs Computational Biology

• **Computational Biology**: is the study of Biology using computational techniques. The main goal of a computational biologist is to make new insights about Biology and living system. Then **Computational Biology** is about **Science**.

• **Bioinformatics**: is about the creation of new algorithms able to solve problems. The main goal of a bioinformatician is to build tools that can work on biological, medical and pharmaceutical data. Then **Bioinformatics** is about **Computer Science**.

How byte is the human genome?

Things to know:

- DNA is composed of 4 different bases: Adenine (A); Thymine (T), Cytosine (C), Guanine (G)
- DNA has a twisted-ladder double helix shape: A=>T and C=>G
- human genome (haploid): 3e+09 base pair

Solution (in a perfect word): ≈ 715 megabyte

- 2 bits foreach base pair
  - 4 different base pair possibilities: AT; TA; CG; GC;
  - 4 different bits possibilities: 00; 11; 10; 01;

1 bytes (8 bits) represents 4 DNA base pairs;

- 3e+09 base pair / 4 DNA base pairs * 1 bytes = 7.5e+08 bytes

7.5e+08 bytes / 2^20 megabyte ≈ 715 megabyte

Solution (in a real word): ≈ 200 gigabyte

- generation of short “reads” and “align” them => coverage;
- for example, a whole genome sequenced at 30x coverage means that, on average, each base on the genome was covered by 30 sequencing reads;
- output file stores not only letters, but also a lot of other info (eg quality);

Prediction of:

- **Protein Function (applications in Molecular Biology);**
- **Human gene-abnormal phenotype associations (applications in Medicine);**

Complex Classification or Ranking Problem

Issues:

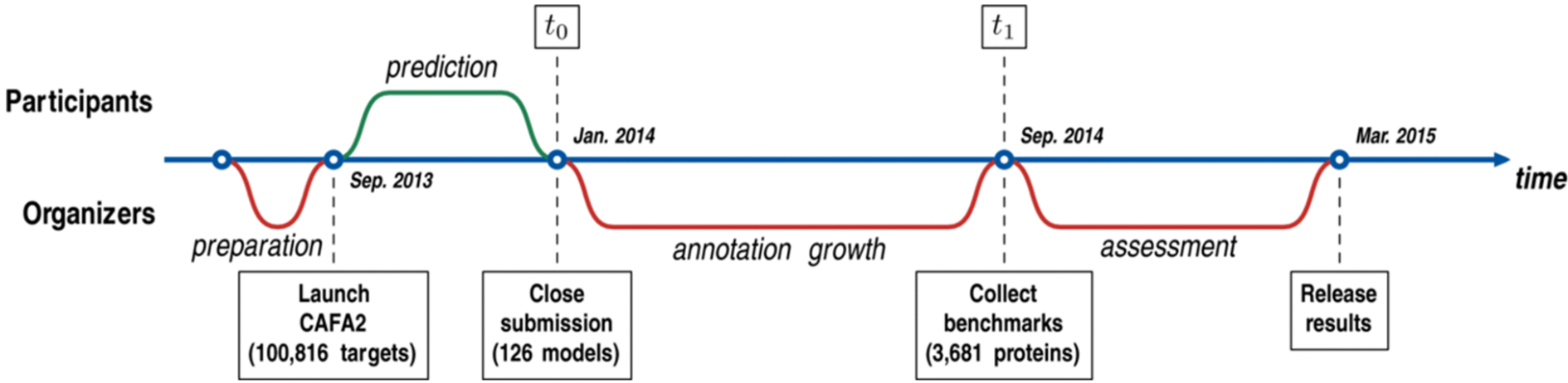
- **multi-class**: hundreds of thousands of functional classes to predict;
- **multi-label**: an instance (i.e. gene/protein) may be annotated to more than one class at the same time;
- **classes are unbalanced**: small number of 'positives' annotations and a large number of 'negatives' annotations;
- **dependencies among labels***:* functional classes are hierarchically related;
- **different level of reliability**: each annotation is labeled with an *evidence code that* indicates how the annotation to a particular term is supported;
  - IPI/IGI: Inferred from Physical/Genetic Interaction (Experimental Evidence);
  - ISS: Inferred from Sequence Similarity (Computational Analysis Evidence)
  - TAS: Traceable Author Statement (annotation made on the basis of a statement made by the authors in the reference cited)
  - … and much more. Full set of available evidence codes at GO website;

Problems of great interest in the Scientific Community

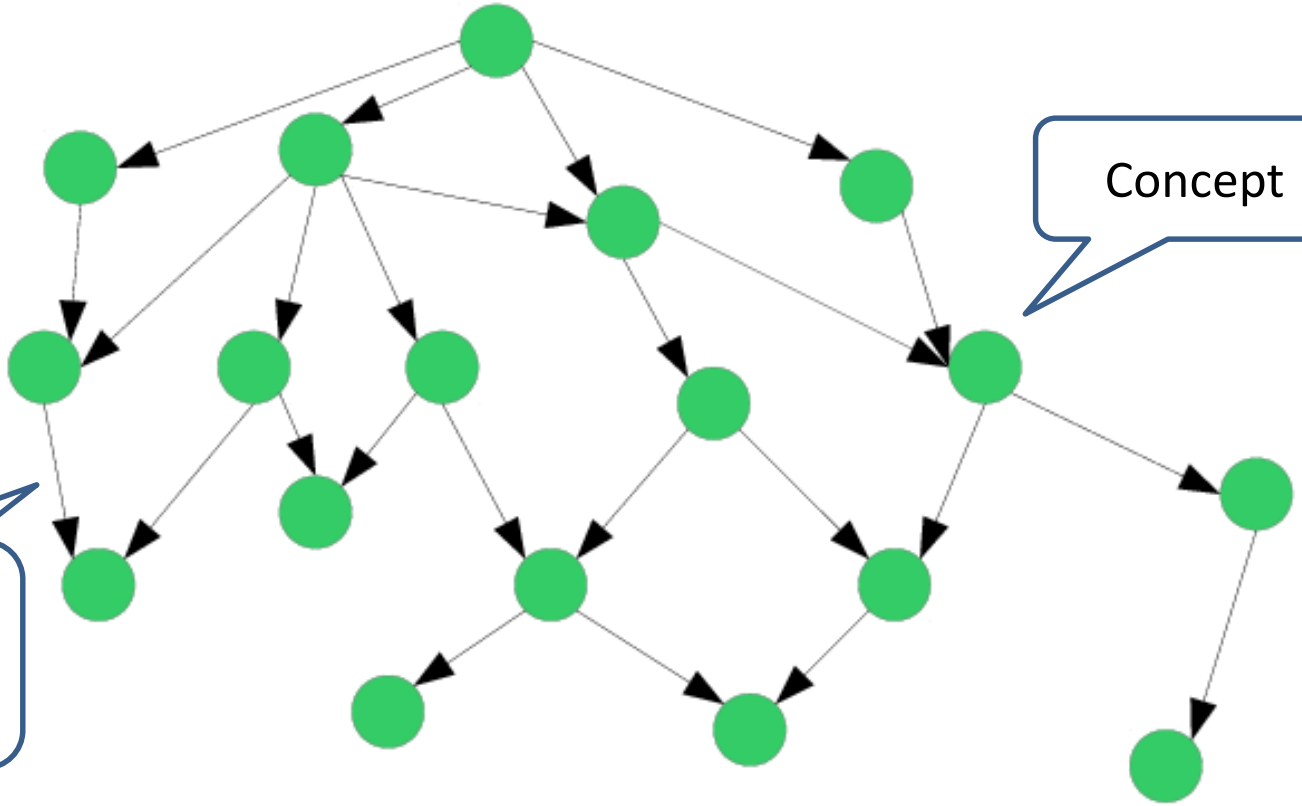
CAFA3

Critical Assessment of Function Annotation (CAFA3) gathering the main international research groups interested:

1. Protein Function Prediction;
2. Prediction of Gene-Abnormal Phenotype Association;



An ontology is an high-level representation of a domain of knowledge that describes concepts and semantic relationships between them in a form of Directed Acyclic Graph (DAG).



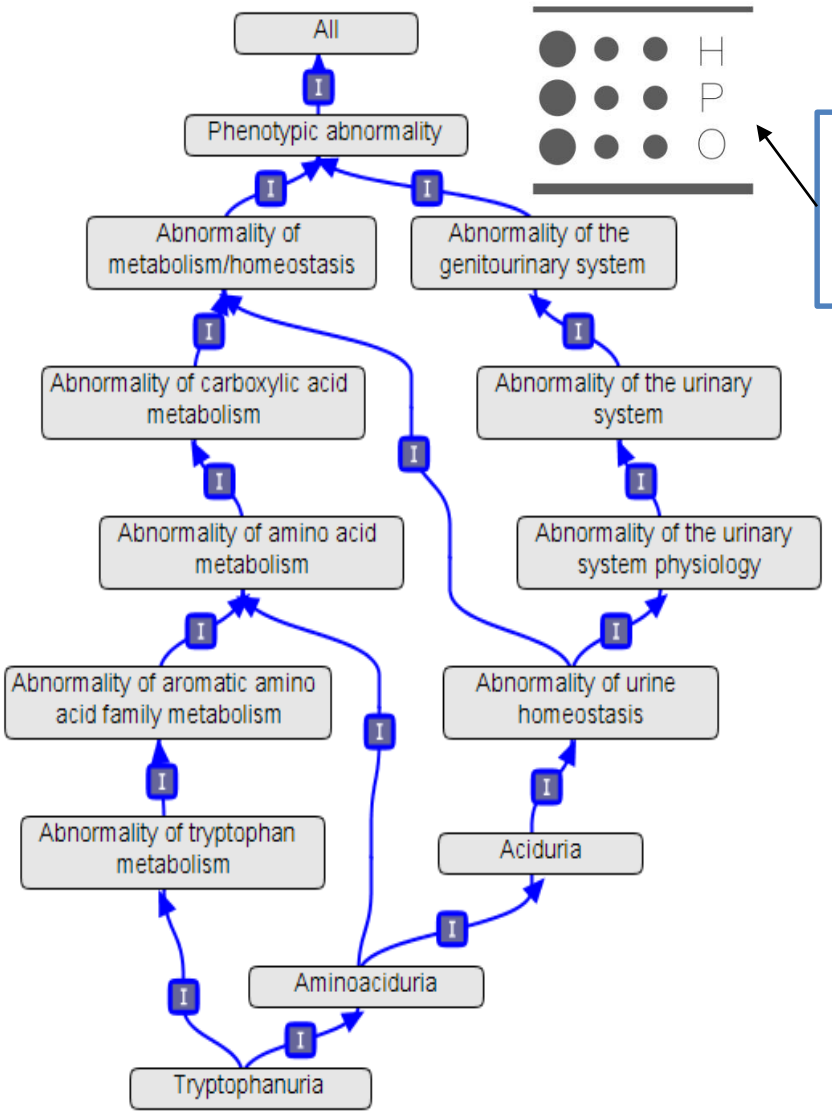
- **Human Phenotype Ontology** (HPO): provides a standardized categorization of the abnormalities associated to human diseases;
- **Gene Ontology** (GO): describes the function of genes and gene products;
- **Disease Ontology** (DO): describes the classification of human diseases organized by etiology;
- **Chemical Entities of Biological Interest** (ChEBI): structured dictionary of molecular entities focused on 'samall' chemical compound;
- **MErged Disease voCabulary** (MEDIC): map the flat list of OMIM disease terms into the hierarchical nature of the MeSH vocabulary;
- **Anatomical Ontologies** : structured controlled vocabulary of the anatomy and development of the Zebrafish (ZFO), Xenopus (XAO), Mouse (MA);

More at OBO Foundry (Open Biological and Biomedical Ontologies): http://www.obofoundry.org/

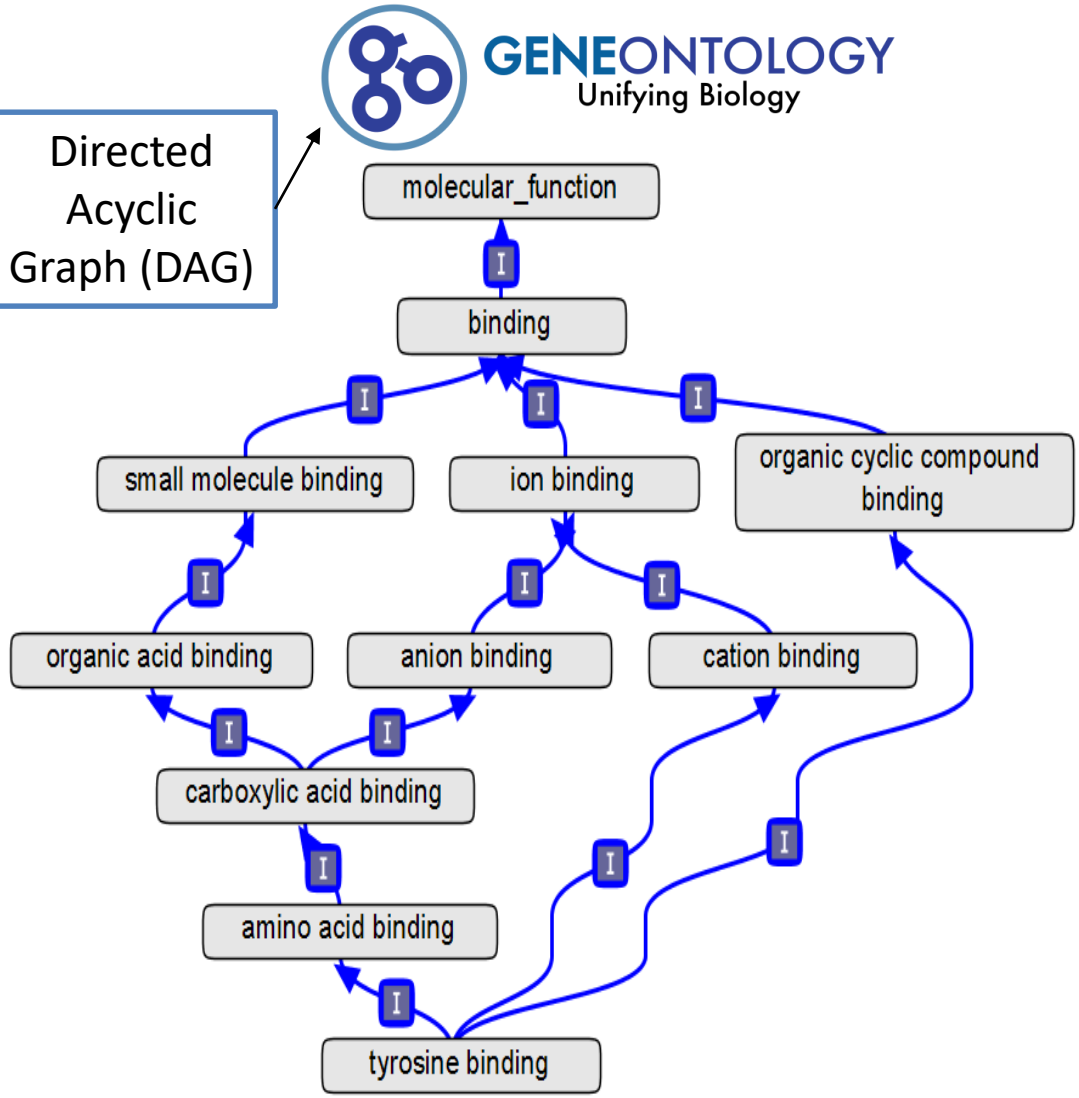
OBO-EDIT (http://oboedit.org/): open source ontology editor

Problem: Hierarchical prediction of Abnormal Phenotype associated to human diseases

Problem: Hierarchical Prediction of Protein Functions



Directed Acyclic Graph (DAG)



Human Phenotype Ontology

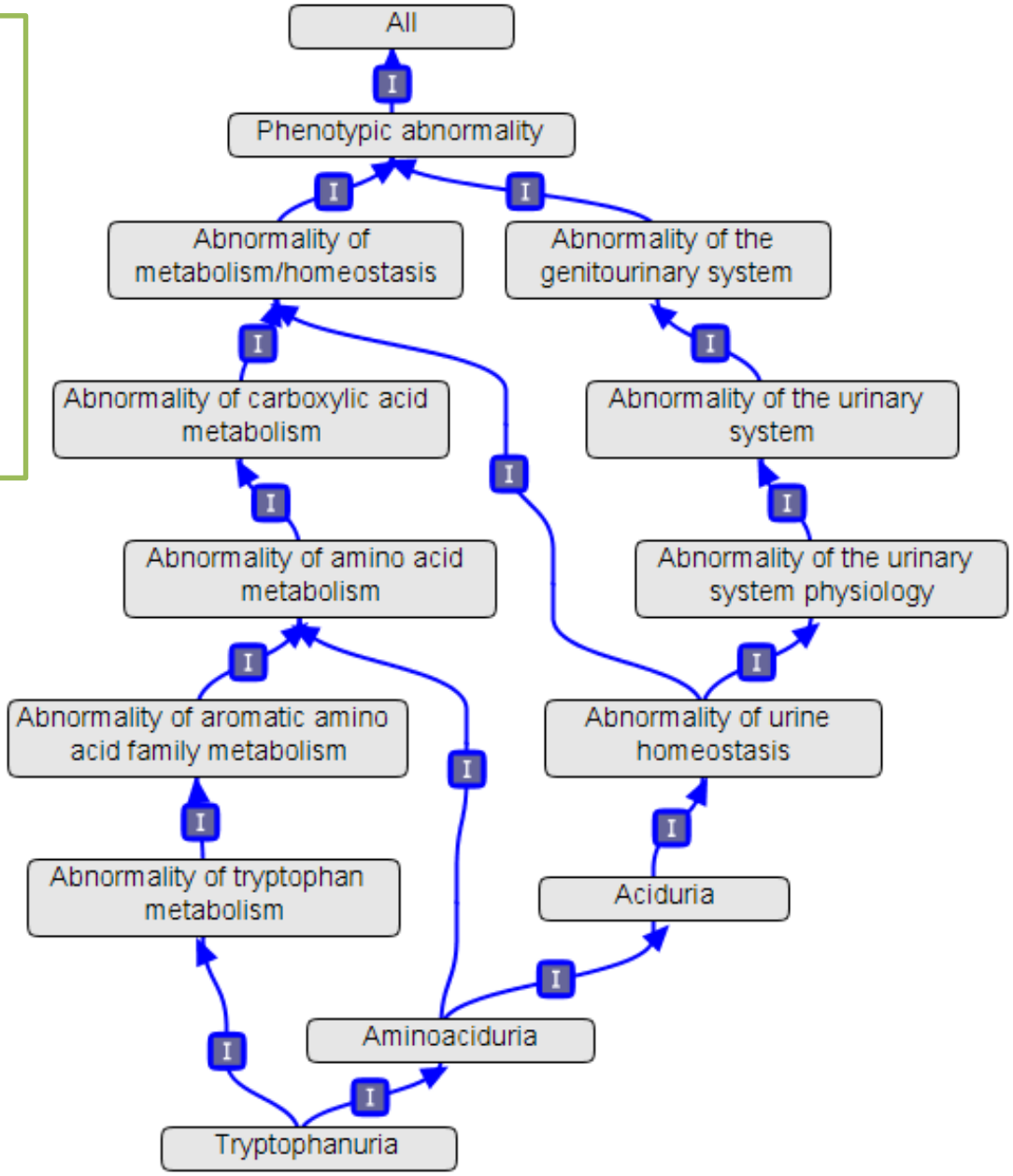
(HPO) (Köhler et al., 2017)

Link: http://human-phenotype-ontology.github.io/

What is: standardized categorization of the phenotypic abnormalities associated to human diseases

I : all relationships in the HPO are **is-a** relationships, i.e. simple class-subclass relationships

HPO (release: 2019-04-15)
Tot. Number of Nodes: 14,407
Tot. Number of Edges: 18,249

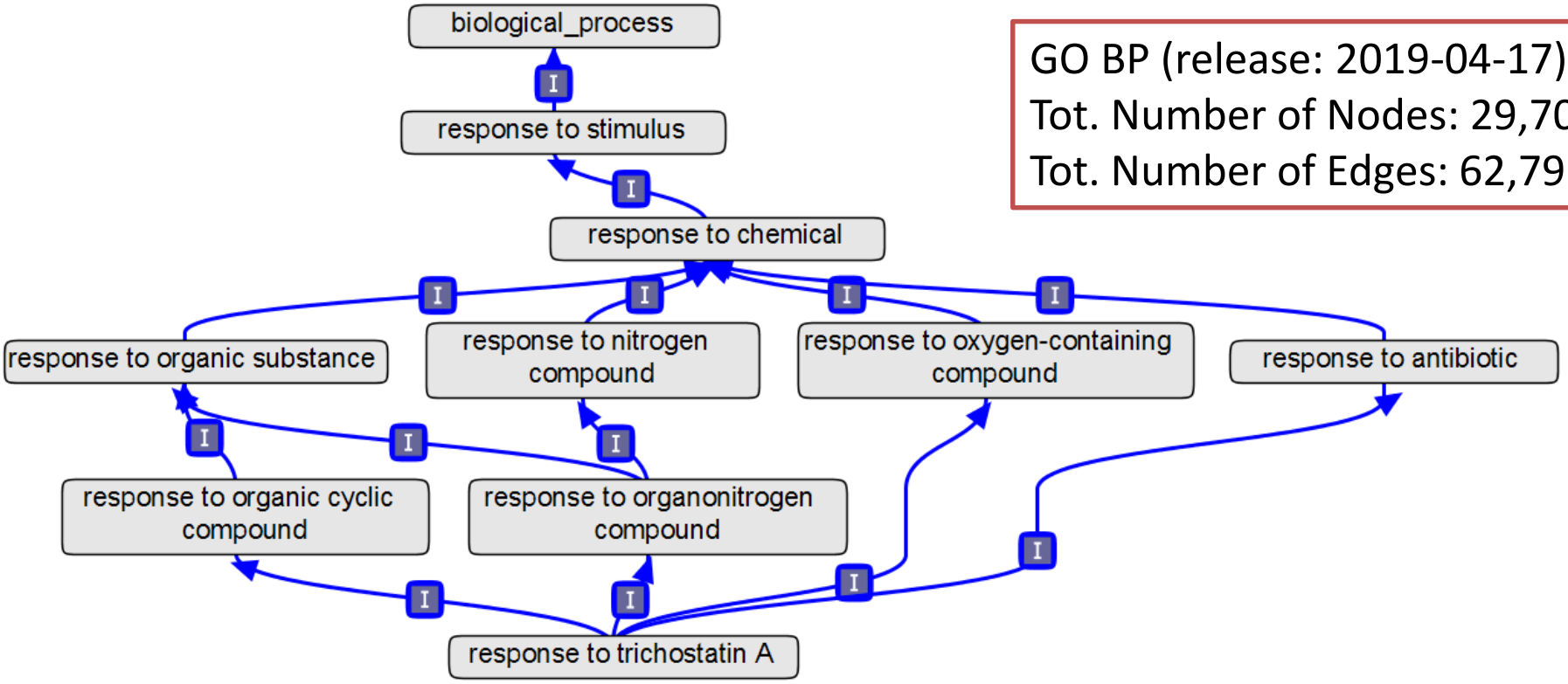


Gene Ontology (GO) (Ashburner et al., 2000)

Link: http://www.geneontology.org/

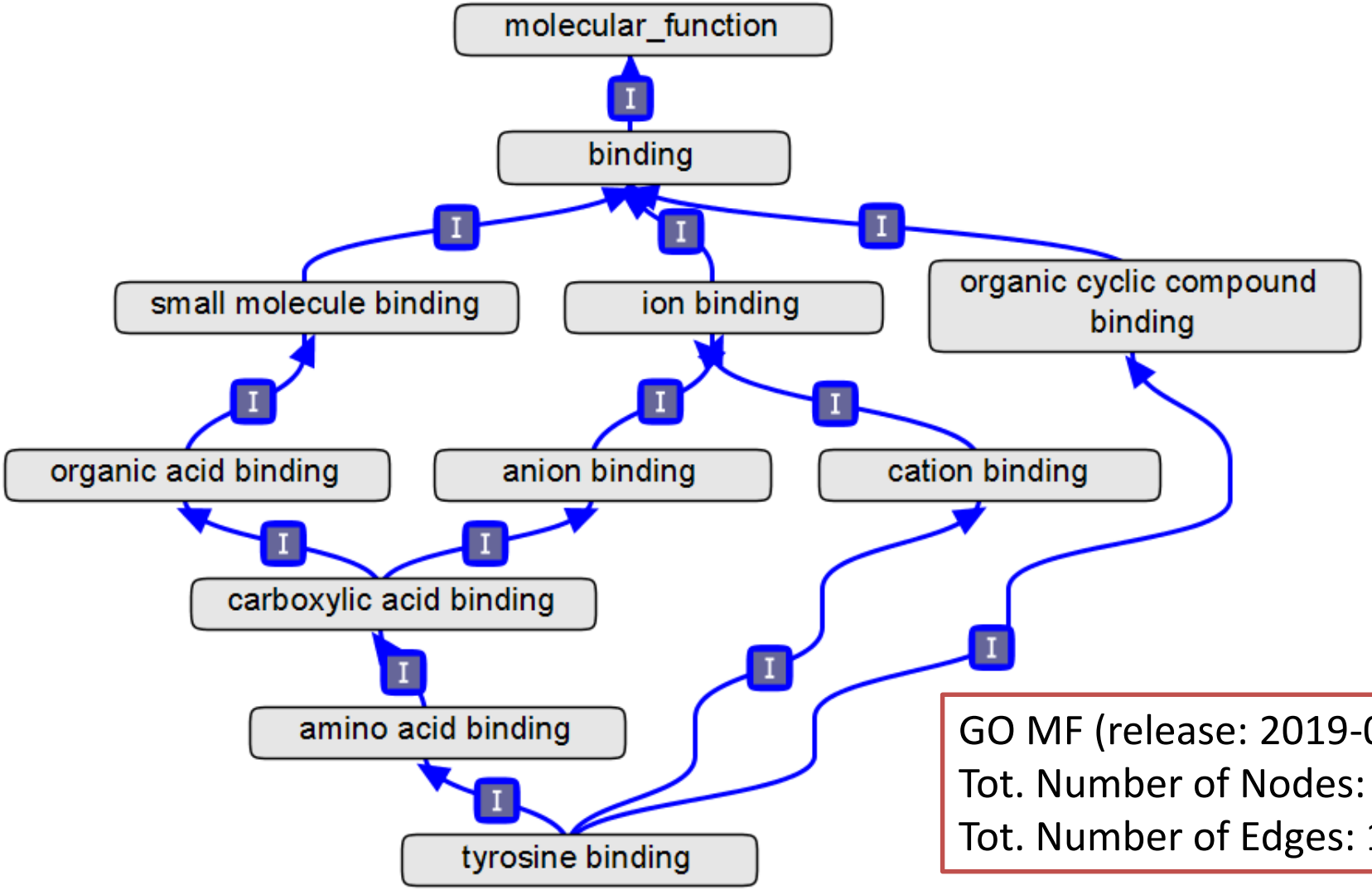
What is: three ***disjoint*** structured ontologies that describe gene products in terms of their association with BP, MF and CC in a species-independent manner.

Biological Process (BP) describes a collection of events carried out by one or more molecular functions (lipid metabolic process, Krebs acid cycle, antibiotic response).



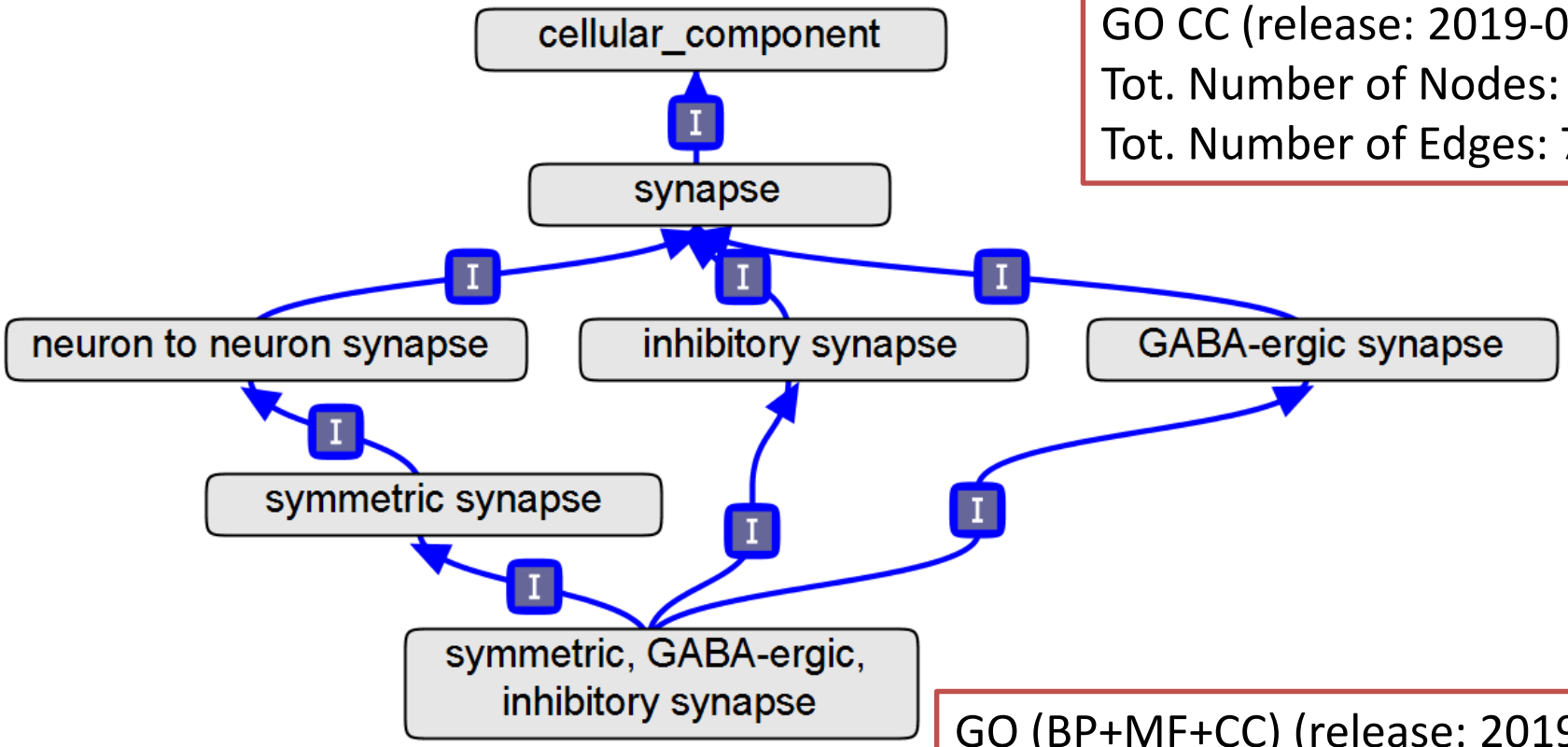
GO BP (release: 2019-04-17)
Tot. Number of Nodes: 29,704
Tot. Number of Edges: 62,791

Molecular function (MF) describes activities that occur at molecular level, such as catalytic or binding activities.



GO MF (release: 2019-04-17)
Tot. Number of Nodes: 11,113
Tot. Number of Edges: 13,584

Cellular component (CC) ontology describes locations, at the levels of subcellular structures or macromolecular complexes, in which a specific gene product is located (e.g. nucleus, nuclear inner membrane, ribosome, synapse).



GO CC (release: 2019-04-17)
Tot. Number of Nodes: 4,200
Tot. Number of Edges: 7,533

GO (BP+MF+CC) (release: 2019-04-17)
Tot. Number of Nodes: 45,017
Tot. Number of Edges: 83,908

Hierarchy-unaware approaches proposed in literature

- **sequence based methods**: follow "transfer-of-annotation" paradigm (BLAST (Altschul et al. 1990), PANNZER (Holm et al. 2018))
- **network based methods**: transfer annotations by exploiting the "proximity relationships" between connected nodes (GBA (Oliver et al. 2000), RANKS (Valentini et al. 2018))

Drawback:

fail to exploit the inherent hierarchical structure of the annotation space

Flat Classifier: predict each class separately

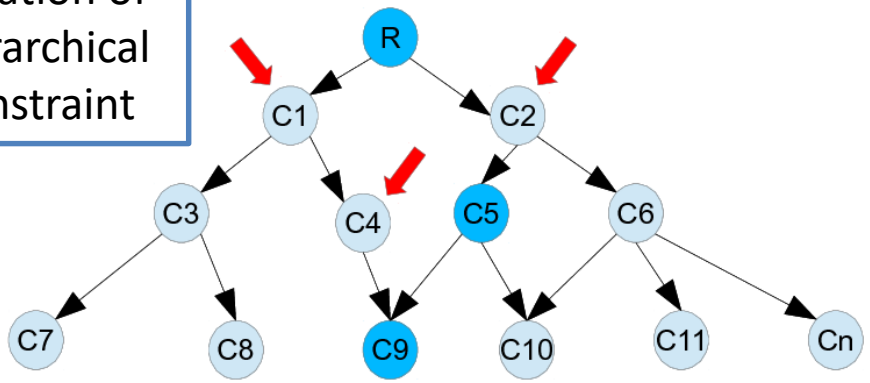
Advantage: simplicity → makes prediction just for one class/term

Drawbacks:
- a priori loss of information
- neglects the hierarchical structure

A Toy Example: Flat Classification

Hierarchical Constraint:
positive instance "P" for a class **implies** positive instance for all ancestors of that class

violation of hierarchical constraint

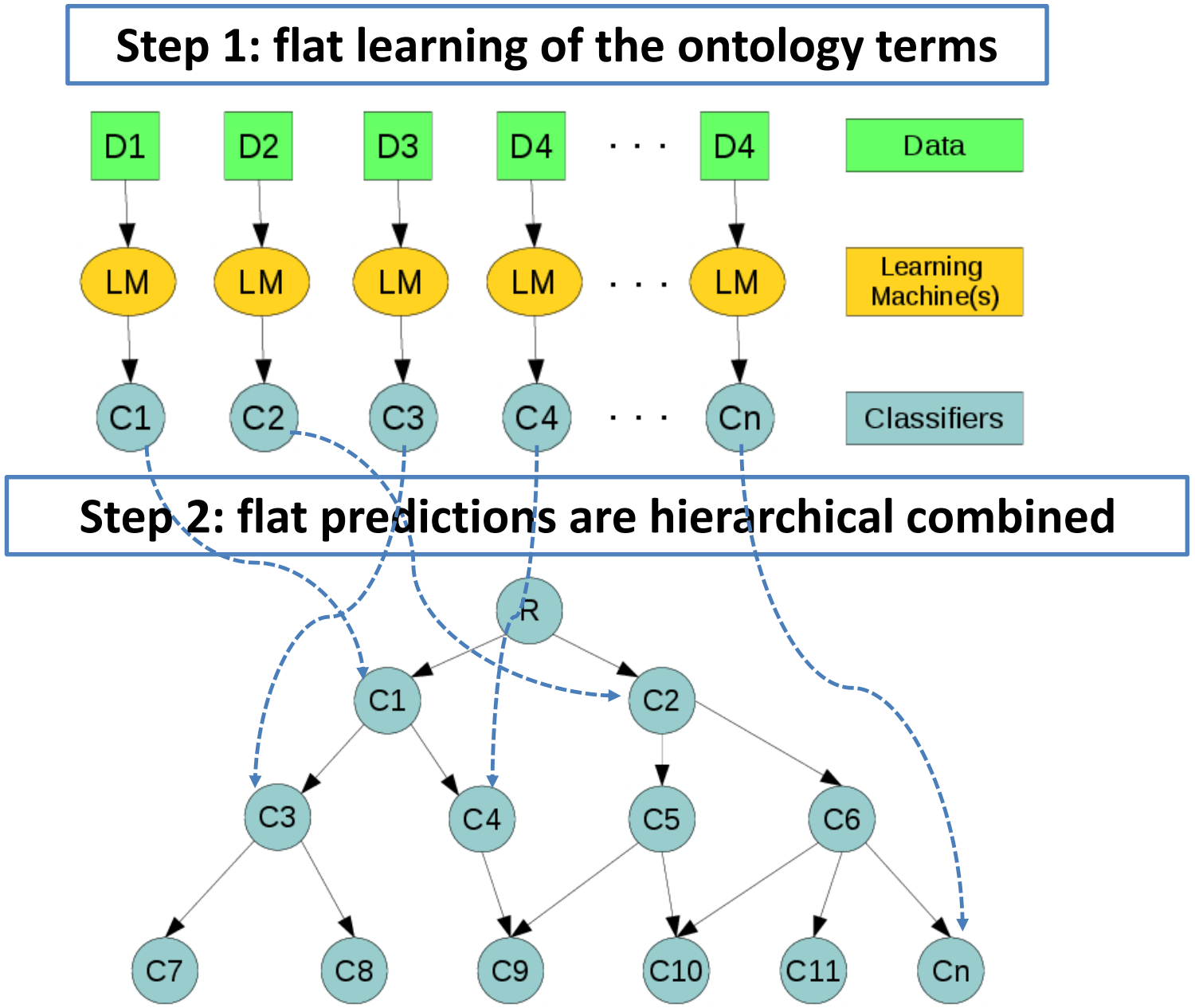


Hierarchy-aware approaches proposed in literature:

- Kernel-based structured output methods: GOstruct (Sokolov and Benhur 2010) PHENOstruct (Kahanda et al. 2015);
- Hierarchical Ensemble Methods (Guan et al. 2008, *Valentini 2014*);

Advantage

- improve classification performance by explicitly taking into account the hierarchical relationships between labels

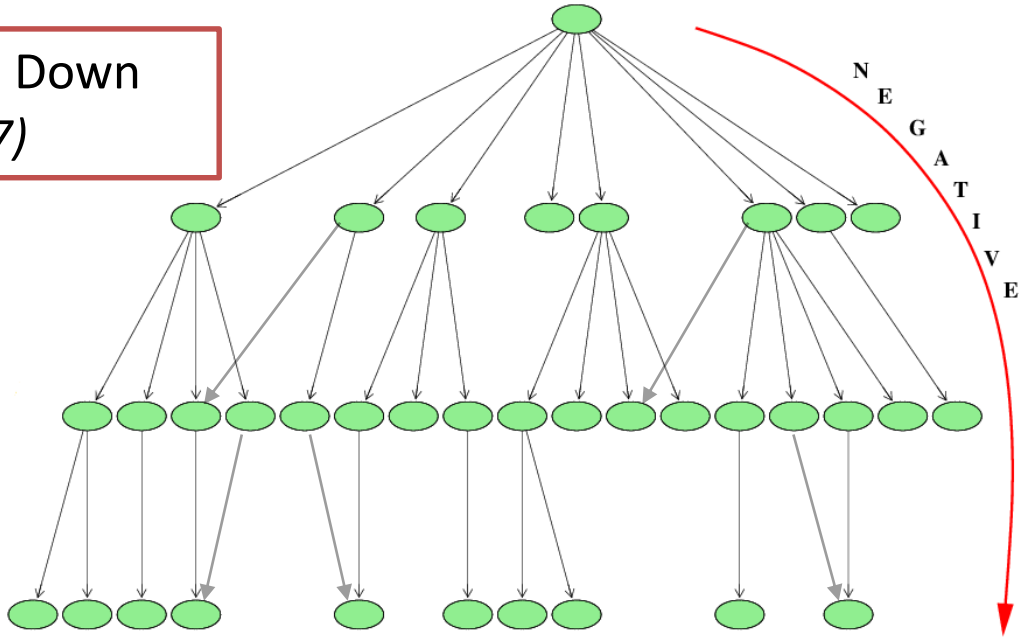

Step 1: flat learning of the ontology terms
D1
D2
D3
D4
D4
Data
LM
LM
LM
LM
LM
Learning Machine(s)
C1
C2
C3
C4
Cn
Classifiers
Step 2: flat predictions are hierarchical combined
R
C1
C2
C3
C4
C5
C6
C7
C8
C9
C10
C11
Cn

State-of-the-art Hierarchical ensemble methods

- Most ensembles are conceived just for tree-structured taxonomies (*Valentini 2011, Cesa-Bianchi et al. 2012, Paes et al. 2012, Hernandez et al. 2013*);
- Only a few for DAG-structured taxonomies (*Obozinski et al. 2008, Schietgat et al. 2010*);
- With DAG-structured taxonomies it is difficult to achieve results comparable with flat methods (*Obozinski et al. 2008*);
- DAGs are more complex than trees:
  - more parents;
  - more edges;
  - multiple paths;
  - nodes may belong to multiple levels;

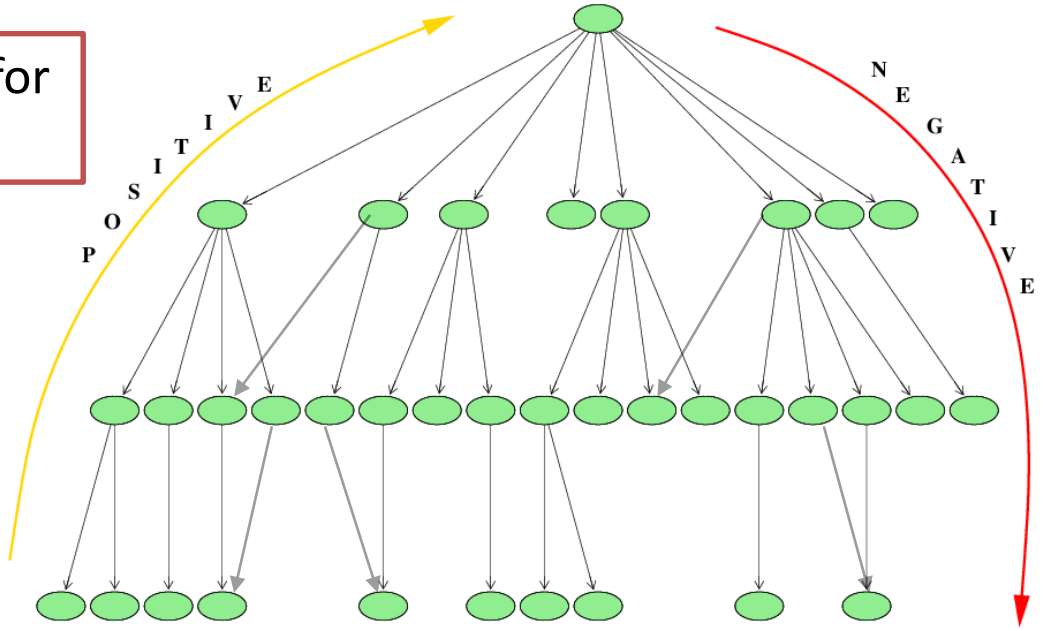
HTD-DAG: Hierarchical Top Down for DAGs *(Notaro et al. 2017)*

Just Top-Down Step



TPR-DAG: True Path Rule for DAGs *(Notaro et al. 2017)*

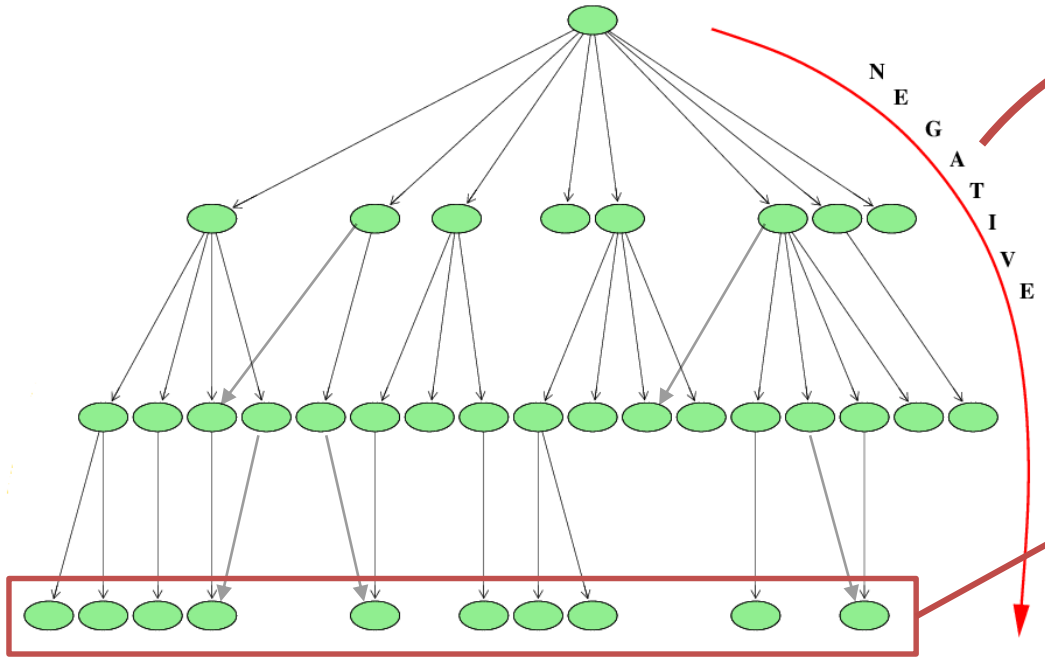
1. Bottom-Up Step
2. Top-Down Step



HTD-DAG:

Flat scores $\hat{y}_i$ are hierarchically corrected to $\bar{y}_i$ according to this simple rule:

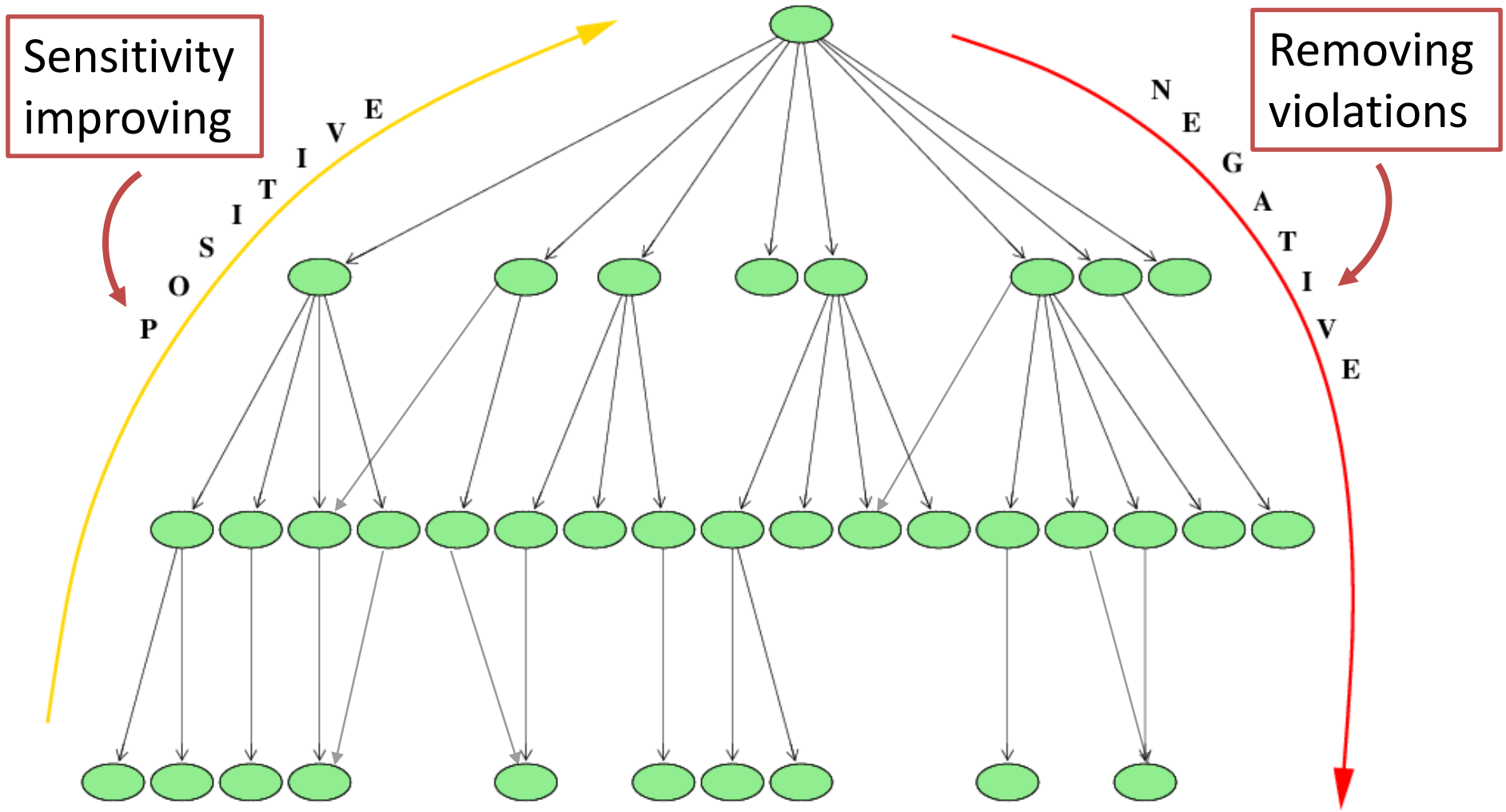
$$\bar{y}_i := \begin{cases} \hat{y}_i & \text{if} \quad i \in root(G) \\ \min_{j \in par(i)} \bar{y}_j & \text{if} \quad \min_{j \in par(i)} \bar{y}_j < \hat{y}_i \\ \hat{y}_i & \text{otherwise} \end{cases}$$



Remove constraint violations

Worst scenario: predictions at leaves nodes are negatives

TPR ensemble for DAGs: double flow of information



In the bottom-up Step the ensemble decision is modified by averaging the local prediction of a node *i* with that of its positive children ϕ_i :

1)

$$\bar{y}_i := \frac{1}{1 + |\phi_i|}(\hat{y}_i + \sum_{j \in \phi_i} \bar{y}_j)$$

Different strategies can be used to define the positive ϕ_i of class *i* :

A. Adaptive Threshold Strategy: maximize $\mathcal{M}$ on training data by internal CV

$$\phi_i := \{j \in child(i) | \bar{y}_j > t_j^*, t_j^* = \arg\max_t \mathcal{M}(j,t)\}$$

B. Threshold Free Strategy: positive children are those that achieve a score higher than that of their parents

$$\phi_i := \{j \in child(i) | \bar{y}_j > \hat{y}_i\}$$

TPR-DAG is a family of algorithms

C. **Weighted TPR**: $w \in [0,1]$ to balance the contribution between node *i* and that of its positive children

$$\bar{y}_i := w\hat{y}_i + \frac{(1-w)}{|\phi_i|}\sum_{j\in\phi_i}\bar{y}_j$$

D. **DEScendant Classifier ENSemble (DESCENS)**: to enhance the contribution of the of the most specific nodes we can consider the descendants instead of children

$$\bar{y}_i := \frac{1}{1+|\Delta_i|}(\hat{y}_i + \sum_{j\in\Delta_i}\bar{y}_j) \qquad \Delta_i = \{j \in desc(i) | \bar{y}_j > t_j\}$$

E. **Descendants-τ** : $\tau \in [0,1]$ to balance the contribution between ϕ_i e δ_i

$$\bar{y}_i := \frac{\tau}{1+|\phi_i|}(\hat{y}_i + \sum_{j\in\phi_i}\bar{y}_j) + \frac{1-\tau}{1+|\delta_i|}(\hat{y}_i + \sum_{j\in\delta_i}\bar{y}_j) \qquad \delta_i = \Delta_i \setminus \phi_i$$

Input:
- $G =< V, E >$
- $V = \{1, 2, \ldots, |V|\}$
- $\hat{\boldsymbol{y}} =< \hat{y}_1, \hat{y}_2, \ldots, \hat{y}_{|V|} >, \quad \hat{y}_i \in [0, 1]$

begin algorithm

01: A. Compute $\forall i \in V$ the max distance from $root(G)$:
02: $E' := \{e' | e \in E, e' = -e\}$
03: $G' :=< V, E' >$
04: $dist :=$ Bellman.Ford$(G', root(G'))$
05: B. Per-level bottom-up visit of G:
06: for each d from $\max(dist)$ to 0 do
07: $N_d := \{i | dist(i) = d\}$
08: for each $i \in N_d$ do
09: Select the set ϕ_i of "positive" children
10: $\bar{y}_i := \frac{1}{1+|\phi_i|}(\hat{y}_i + \sum_{j \in \phi_i} \bar{y}_j)$
11: end for
12: end for
13: C. Per-level top-down visit of G:
14: $\hat{\boldsymbol{y}} := \bar{\boldsymbol{y}}$
15: for each d from 1 to $\max(dist)$ do
16: $N_d := \{i | dist(i) = d\}$
17: for each $i \in N_d$ do
18: $x := \min_{j \in par(i)} \bar{y}_j$
19: if $(x < \hat{y}_i)$
20: $\bar{y}_i := x$
21: else
22: $\bar{y}_i := \hat{y}_i$
23: end for
24: end for

end algorithm

Output:
- $\bar{\boldsymbol{y}} =< \bar{y}_1, \bar{y}_2, \ldots, \bar{y}_{|V|} >$

Block A. Maximum Distance of each node from the root:
- Bellman-Ford algorithm;
- Topological Sort algorithm.

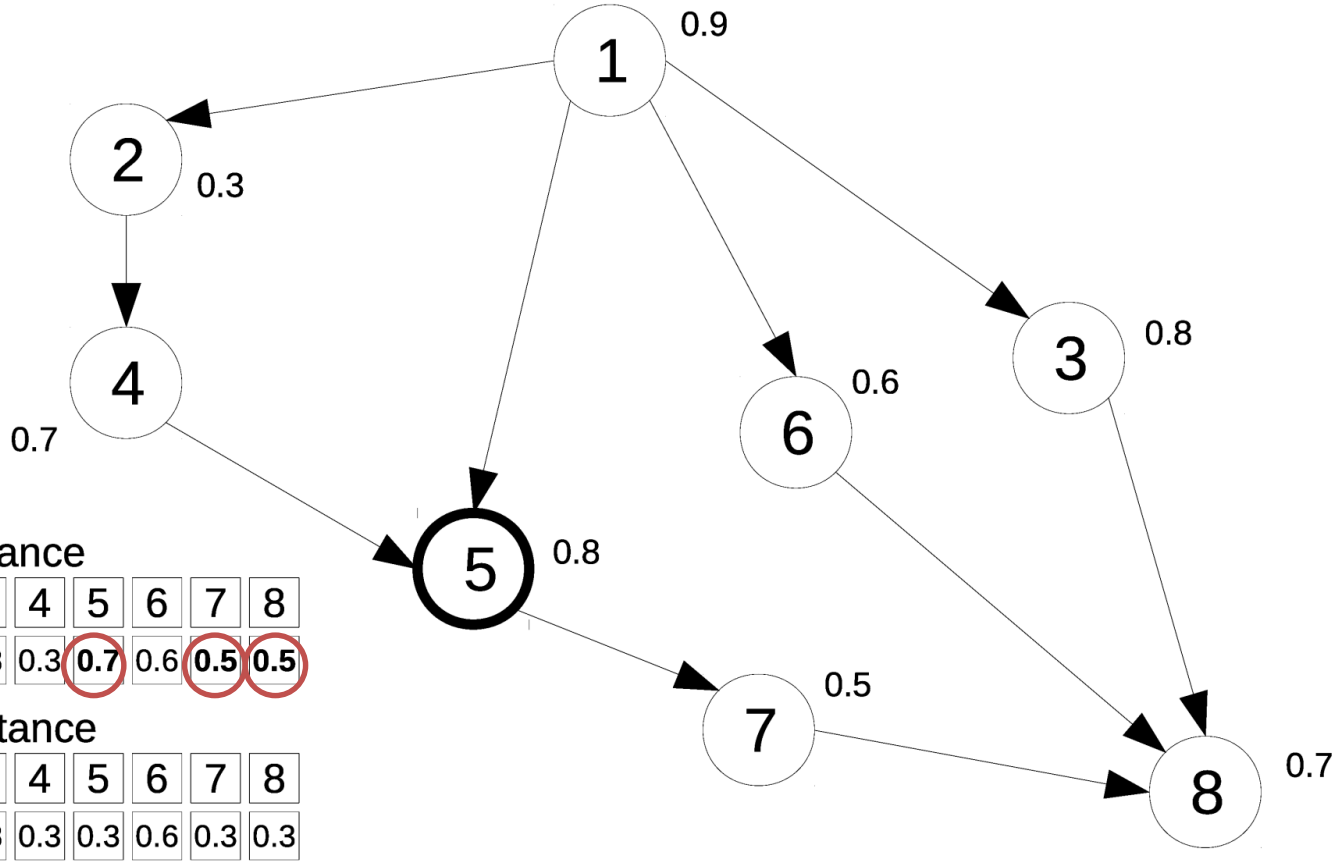
Block B. Performs a per-level bottom-up visit of the graph and updates the flat predictions according to one of the aforementioned strategies.
This step *does not assure* the consistency of the predictions.

Block C. Nodes are processed by level from the least to the most specific terms and the bottom-up scores are corrected according to HTD-DAG rule.

Overall TPR-DAG Computational Complexity: $O(|V|)$

To preserve the consistency of the predictions the levels must be defined according to the maximum distance from the root:

$$\boldsymbol{y} \text{ is consistent} \iff \forall i \in V, j \in parents(i) \Rightarrow y_j \geq y_i$$



inconsistent predictions

Min. distance

1	2	3	4	5	6	7	8
0.9	0.3	0.8	0.3	**0.7**	0.6	**0.5**	**0.5**

consistent predictions

Max. distance

1	2	3	4	5	6	7	8
0.9	0.3	0.8	0.3	0.3	0.6	0.3	0.3

Partial Order Isotonic Regression (IR) (Barlow and Brunk, 1972)

Input:
- $G =< V, E >$
- $V = \{1, 2, \ldots, |V|\}$
- $\hat{\boldsymbol{y}} =< \hat{y}_1, \hat{y}_2, \ldots, \hat{y}_{|V|} >, \quad \hat{y}_i \in [0, 1]$

begin algorithm

01: A. Isotonic correction:

02: $$\bar{\boldsymbol{y}} = \begin{cases} \min_{\bar{\boldsymbol{y}}} \sum_{i \in V} (\hat{y}_i - \bar{y}_i)^2 \\ \forall i, \quad j \in par(i) \Rightarrow \bar{y}_j \geq \bar{y}_i \end{cases}$$

end algorithm

Output:
- $\bar{\boldsymbol{y}} =< \bar{y}_1, \bar{y}_2, \ldots, \bar{y}_{|V|} >$

- IR selects the closest solution (in the sense of the least squared error) to the flat predictions that obeys to the true path rule

IR computational complexity is: $\mathcal{O}(|V|^4)$ (Maxwell et al. 1985)

Generalized Pool-Adjacent-Violators (GPAV) (Burdakov et al., 2006):
- accurate solution to IR problem
- computational complexity is: $\mathcal{O}(|V|^2)$

Input:
- $G =< V, E >$
- $V = \{1, 2, \ldots, |V|\}$
- $\hat{\boldsymbol{y}} =< \hat{y}_1, \hat{y}_2, \ldots, \hat{y}_{|V|} >, \quad \hat{y}_i \in [0, 1]$
- $\boldsymbol{w} =< w_1, w_2, \ldots, w_{|V|} >, \quad w_i \in [0, 1]$

begin algorithm

01: A. $dist := \forall i \in V$ ComputeMaxDist $(G, root(G))$
02: B. Per-level bottom-up visit of G:
03: for each d from $\max(dist)$ to 0 do
04: $N_d := \{i | dist(i) = d\}$
05: for each $i \in N_d$ do
06: Select the set ϕ_i of "positive" children
07: $\bar{y}_i := \frac{1}{1+|\phi_i|}(\hat{y}_i + \sum_{j \in \phi_i} \bar{y}_j)$
08: end for
09: end for
10: C. GPAV algorithm
12: $\hat{\boldsymbol{y}} := \bar{\boldsymbol{y}}$
14: $V = \{1, 2, \ldots, |V|\}$ *topologically ordered*;
14: $H := V$
15: $\forall i \in V$ set $B_i = \{i\}$; $B_i^- = i^-$; $U_i = \hat{y}_i$; $W_i = w_i$;
16: for each k from 1 to $|V|$ do
17: while exists $i \in B_k^-$ such that $U_i > U_k$ do
18: find $j \in B_k^-$ such that $U_j := max\{U_i : i \in B_k^-\}$
19: $H := H \setminus \{j\}$
20: $B_k^- := B_j^- \cup B_k^- \setminus \{j\}$
21: $U_k := (W_k U_K + W_j U_K)/(W_k + W_j)$
22: $B_k := B_k \cup B_j$
23: $W_k := W_k + W_j$
24: $\forall i \in B_k$ and $\forall k \in H$ set $\bar{y} := U_k$
25: end while
26: $\bar{y} := U_k \quad \forall i \in B_k$ and $\forall k \in H$
27: end for

end algorithm

Output:
- $\bar{\boldsymbol{y}} =< \bar{y}_1, \bar{y}_2, \ldots, \bar{y}_{|V|} >$

Block A-B: same of *TPR-DAG*

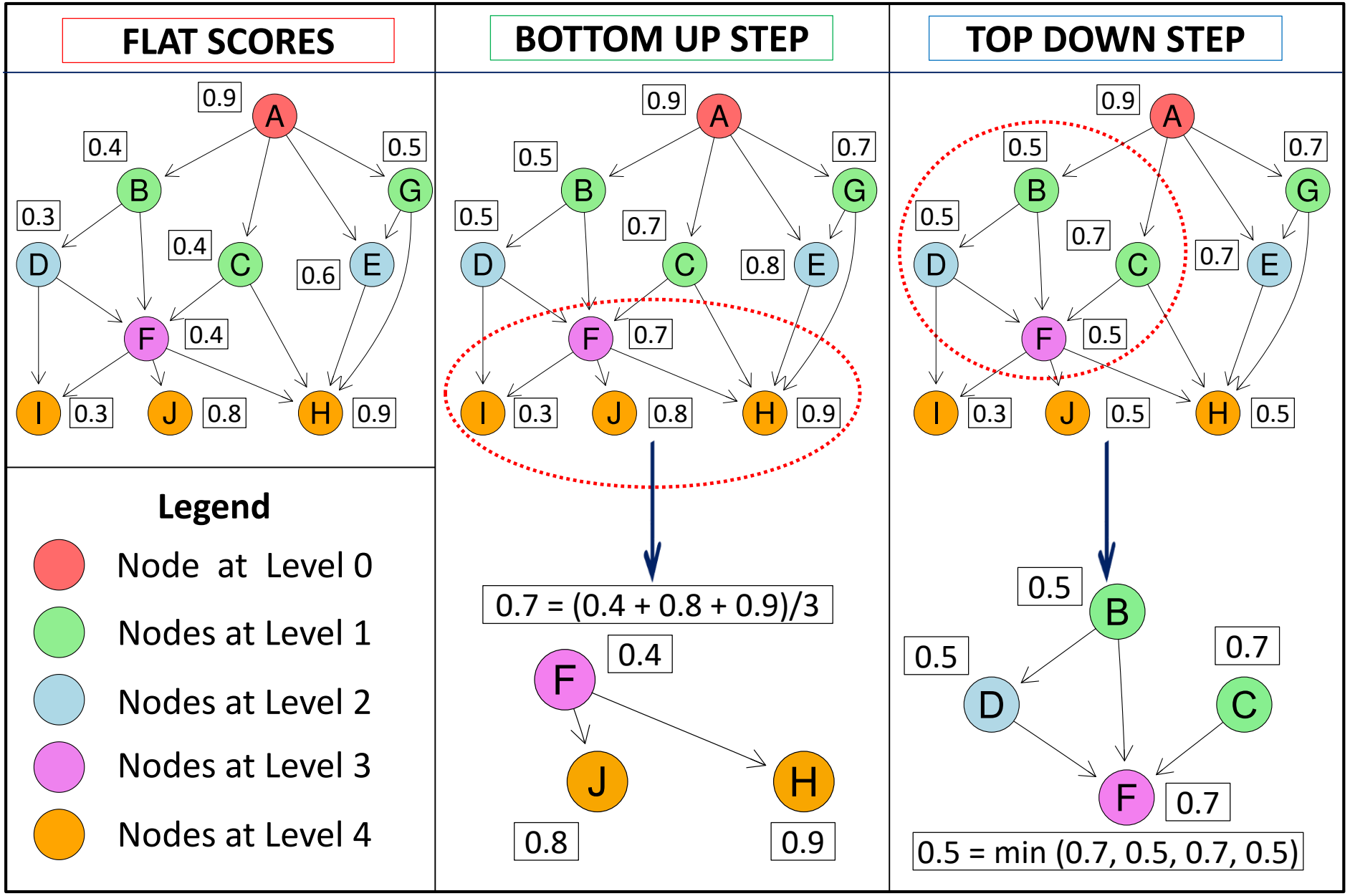
↓

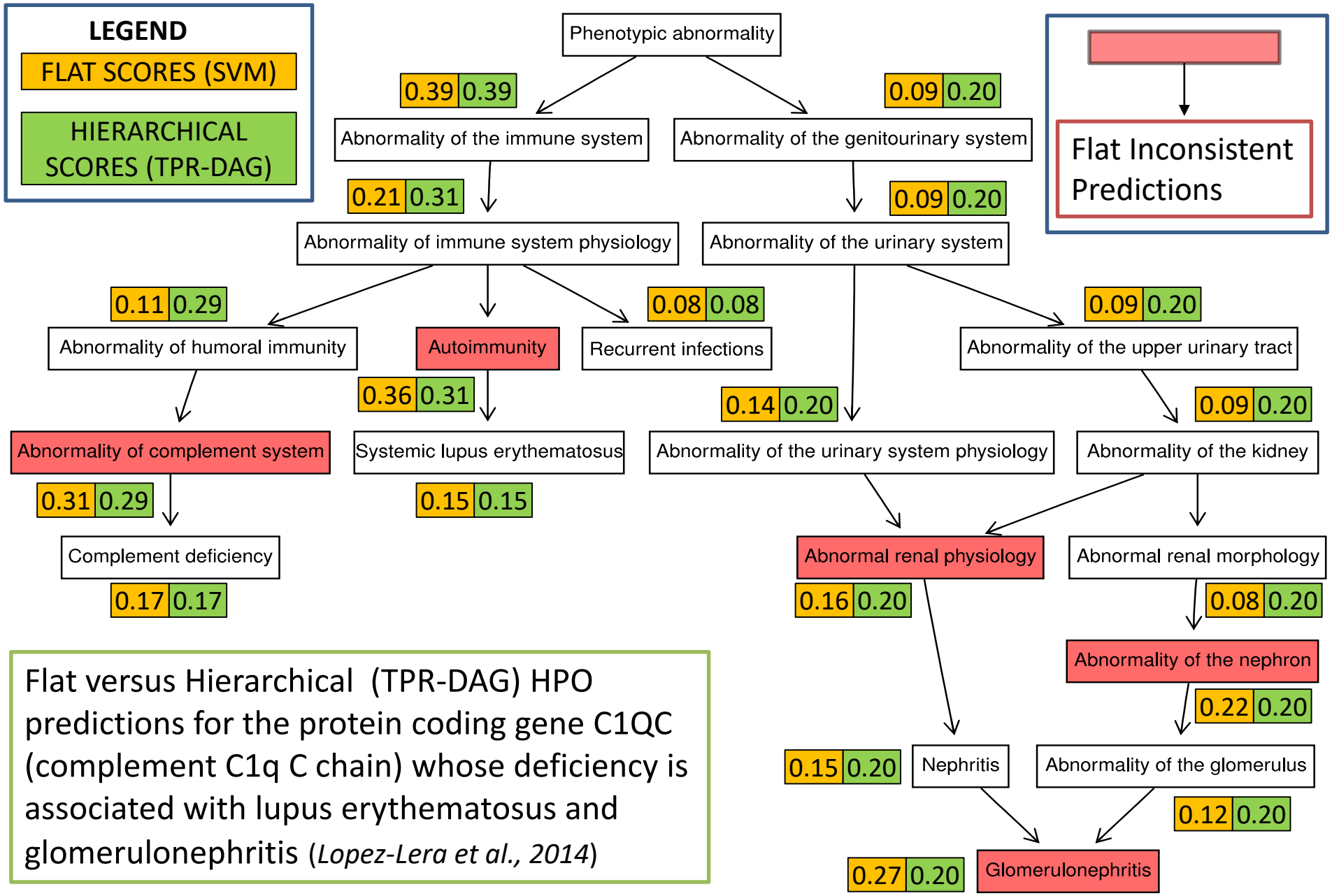
Consistency of prediction violated

Block C: *GPAV* instead of *HTD-DAG*

↓

Consistency of prediction guaranteed


FLAT SCORES
BOTTOM UP STEP
TOP DOWN STEP
Legend
Node at Level 0
Nodes at Level 1
Nodes at Level 2
Nodes at Level 3
Nodes at Level 4
0.7 = (0.4 + 0.8 + 0.9)/3
0.5 = min (0.7, 0.5, 0.7, 0.5)



Flat versus Hierarchical (TPR-DAG) HPO predictions for the protein coding gene C1QC (complement C1q C chain) whose deficiency is associated with lupus erythematosus and glomerulonephritis (*Lopez-Lera et al., 2014*)

HTD-DAG provides consistency predictions:

Given a DAG $G = < V, E >$ a level function ψ that assigns to each node its maximum path length from the root and the set of HTD-DAG flat predictions $\hat{y} = < \widehat{y_1}, \widehat{y_2}, \ldots, \widehat{y_{|V|}} >$ the top-down hierarchical correction of the HTD-DAG algorithm assures that the set of ensemble predictions $\bar{y} = < \overline{y_1}, \overline{y_2}, \ldots, \overline{y_{|V|}} >$ satisfies the following property:

$$\forall i \in V,\ j \in par(i) \Rightarrow \bar{y}_j \geq \bar{y}_i$$

TPR-DAG provides consistency predictions:

Given a DAG $G = < V, E >$, a level function ψ that assigns to each node its maximum path length from the root, a set of predictions $\tilde{y} = < \widetilde{y_1}, \widetilde{y_2}, \ldots, \widetilde{y_{|V|}} >$ generated by the bottom-up step of the TPR-DAG algorithm for each class associated to each node $i \in \{1, \ldots, |V|\}$, the top-down step of the TPR-DAG algorithm assures that for the set of ensemble predictions $\bar{y} = < \overline{y_1}, \overline{y_2}, \ldots, \overline{y_{|V|}} >$ the following property holds:

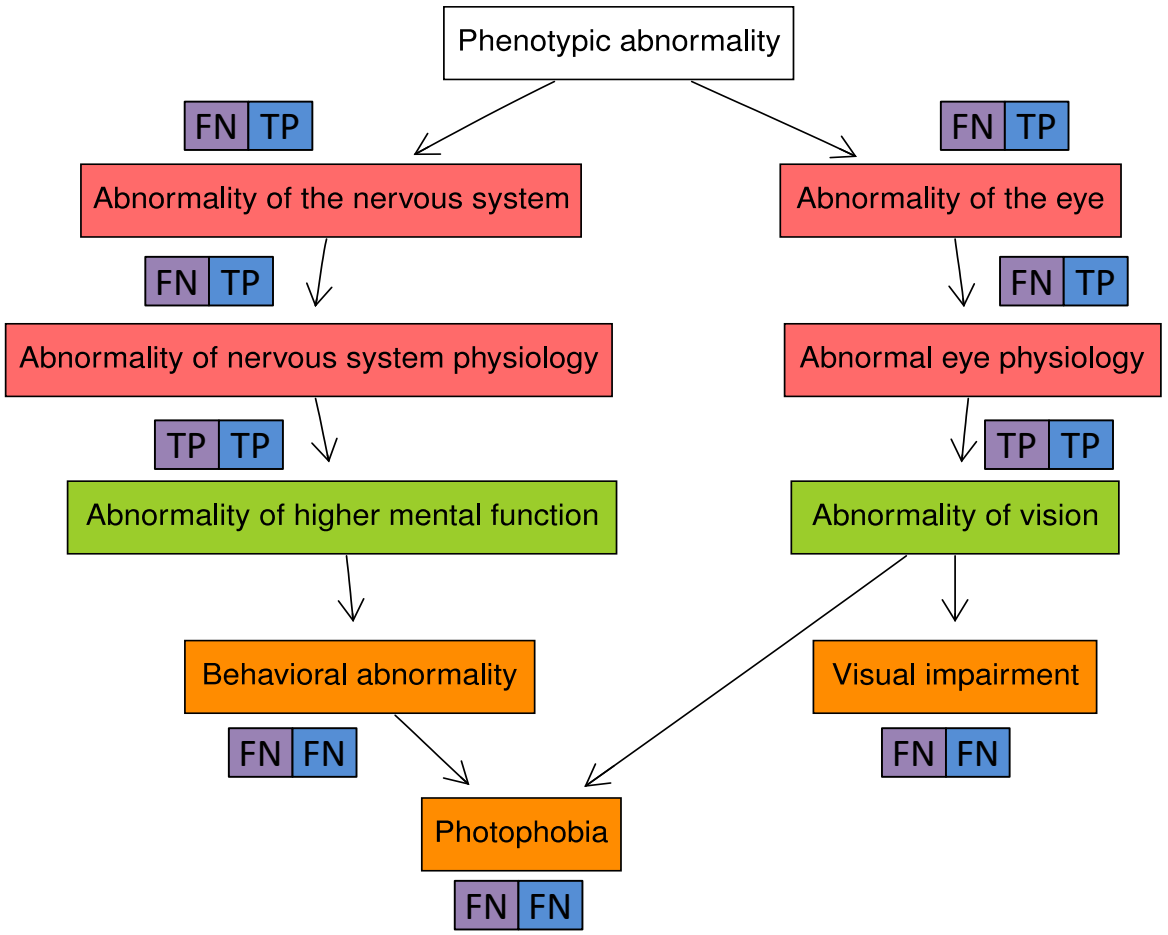
$$\forall i \in V,\ j \in par(i) \Rightarrow \bar{y}_j \geq y_i$$

For an arbitrary node $i \in V$ when it is processed by the top-down step of HTD-DAG algorithm, we may have two basic cases:

1. $i \in root(G)$. By applying the HTD-DAG rule we set $\bar{y}_i := \hat{y}_i$ and the property $j \in par(i) \Rightarrow \bar{y}_j \geq \bar{y}_i$ trivially holds, since $par(i) = \emptyset$
2. $i \notin root(G)$. We may have two cases:
   1. $\hat{y}_i \leq \min_{j \in par(i)} \hat{y}_j$: In this case the HTD-DAG rule sets $\bar{y}_i := \hat{y}_i$ and hence it holds that $j \in par(i) \Rightarrow \bar{y}_j \geq \bar{y}_i$
   2. $\hat{y}_i > \min_{j \in par(i)} \bar{y}_j$: In this case by applying the HTD-DAG rule we have $\bar{y}_i := min_{j \in par(i)} \bar{y}_j$ and hence also in this case the property $j \in par(i) \Rightarrow \bar{y}_j \geq \bar{y}_i$ holds.

The top-down step of the algorithm visits each node exactly one time, at the end of this step the property $j \in par(i) \Rightarrow \bar{y}_j \geq \bar{y}_i$ holds for each node $i \in V$

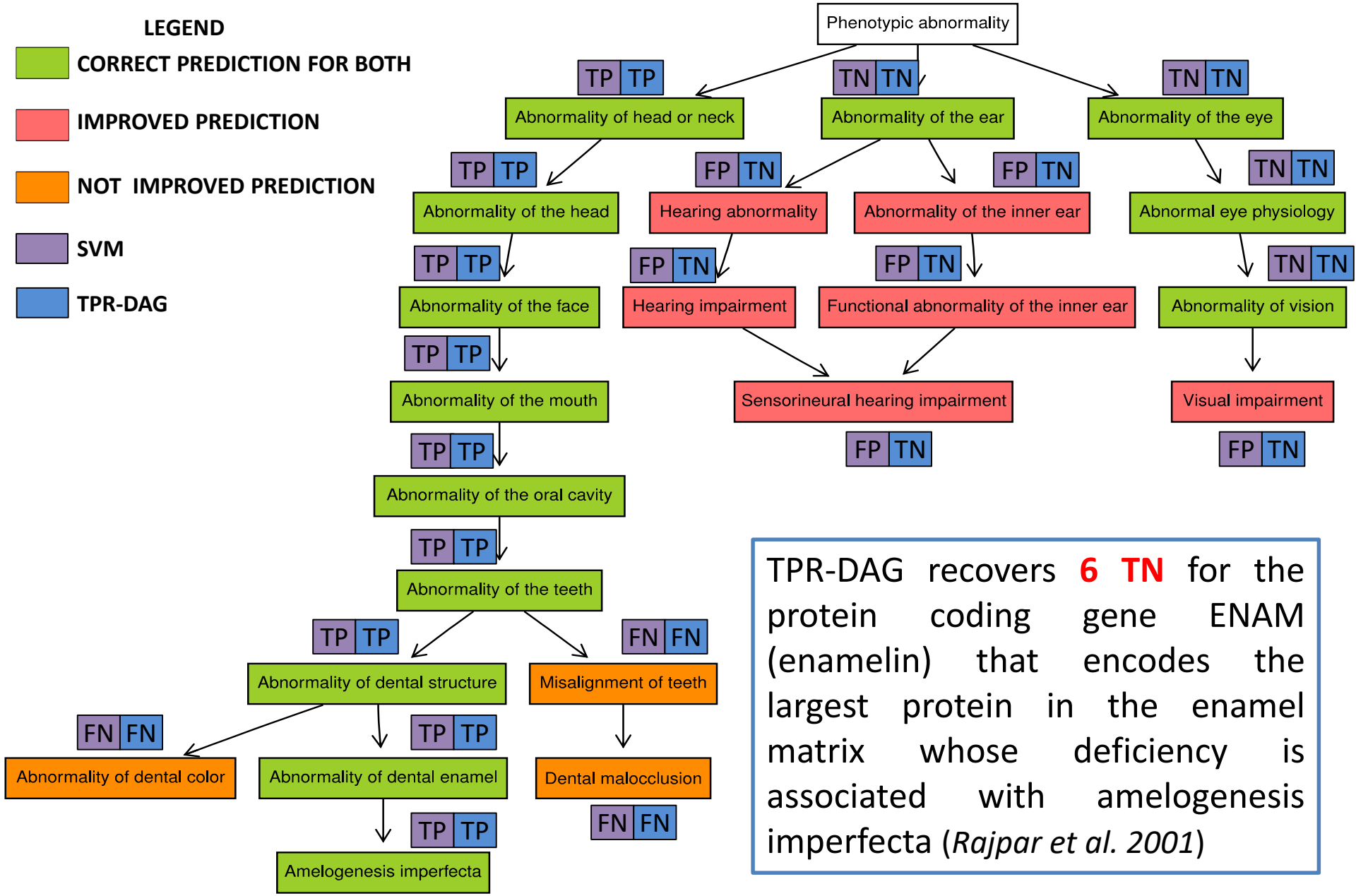
Hierarchical Ensemble Methods (HEMs) improve upon flat predictions by reducing the number of FN and FP.



TPR-DAG recovers **4 TP** for the protein coding gene RGS9 (regulator of G-protein signalling 9) whose mutations cause bradyopsia (*Michaelides et al. 2010*)

LEGEND

- **CORRECT PREDICTION FOR BOTH**
- **IMPROVED PREDICTION**
- **NOT IMPROVED PREDICTION**
- **SVM**
- **TPR-DAG**



TPR-DAG recovers **6 TN** for the protein coding gene ENAM (enamelin) that encodes the largest protein in the enamel matrix whose deficiency is associated with amelogenesis imperfecta (*Rajpar et al. 2001*)

Software Implementation

- HEMs are packaged in the R library **HEMDAG**, which is publicly available both under CRAN and BIOCONDA repository under the GNU General Public License, version 3 (GPL-3.0) and it is available for *Unix*, *Windows* and *Mac* operating system;

- HEMDAG tutorial (created by using SPHINX) explains step-by-step how to use HEDMAG;

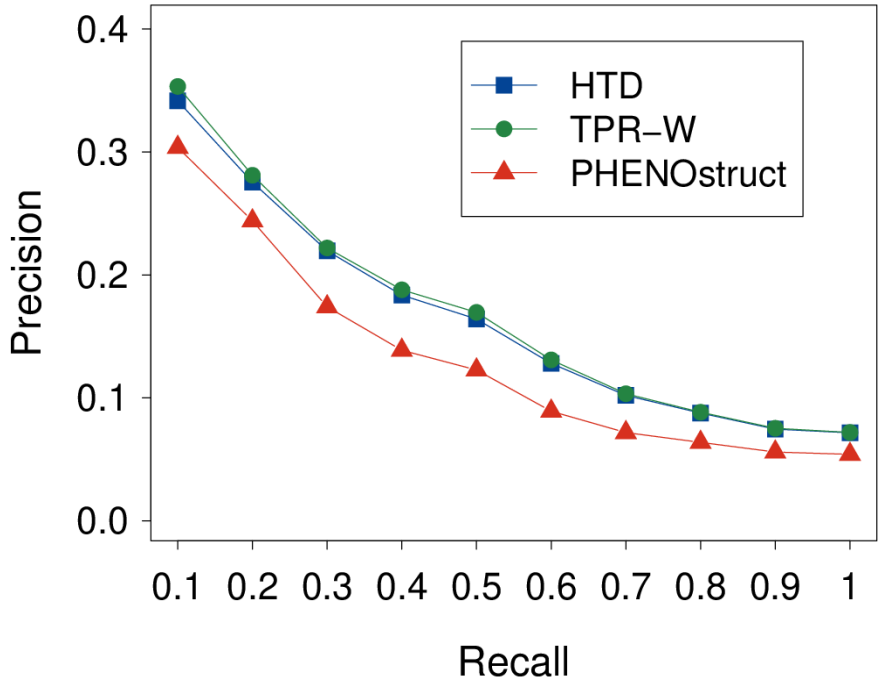
- **HEMDAG** can be safely applied both to DAG and tree-structure taxonomies;

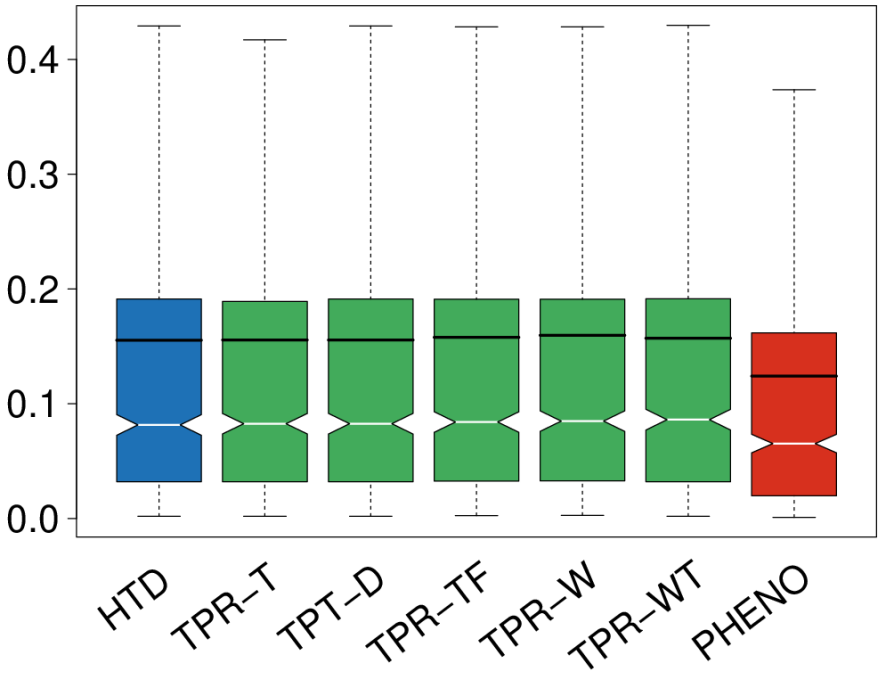
- **OBO::parser** Perl Module to handle GO and HPO obo file (under development);

HEMs vs. **PHENOstruct**, state-of-the-art joint-kernel structured output approach (*Kahanda et al. 2015*)

Precision-Recall curves and AUPRC box-blot across **2444 HPO terms**: HEMs significantly improve PHENOstruct in according to Wilcoxon Sum Rank test ($\alpha = 10^{-9}$) *(Notaro et. al 2017)*

HTD: **12 min**
TPR-W: **3 hours** (tuning of *w* parameter by 5cv)
PHENOstruct: **18 hours**





List of possible "candidate" genes for novel annotations: unannotated genes but predicted to be annotated by our HEMs

Gene Symbol	HPO Term	AUROC	Depth	Distance from Leaves	Evidence
XRCC2	Clubbing of Toes	1.000	9	0	HPO March 2017 Release
LIPE	Insulin-Resistant Diabetes Mellitus	0.9934	6	0	HPO March 2017 Release
IGF2	Neoplasm of the Adrenal Gland	0.9781	5	0	HPO March 2017 Release
ECHS1	Abnormality of Fatty-Acid Metabolism	0.9753	4	0	Chika et al. 2015
CFB	Systemic Lupus Erythematosus	0.9967	5	0	Grossman et al. 2016
TGFBR3	Emphysema	0.9785	5	0	Hersh et al. 2009
BARD1	Nephroblastoma aka Wilms Tumor	0.9615	8	0	Fu et al. 2017
MSH3	Breast Carcinoma	0.9723	5	0	Miao et al. 2015
CAD	Abnormality of Pyrimidine Metabolism	0.9951	4	0	Bobby et al. 2015
COX10	Abnormal Mitochondria in Muscle Tissue	0.9967	6	0	Pitceathly et al. 2013

Inclusion of the novel annotations in the next HPO release

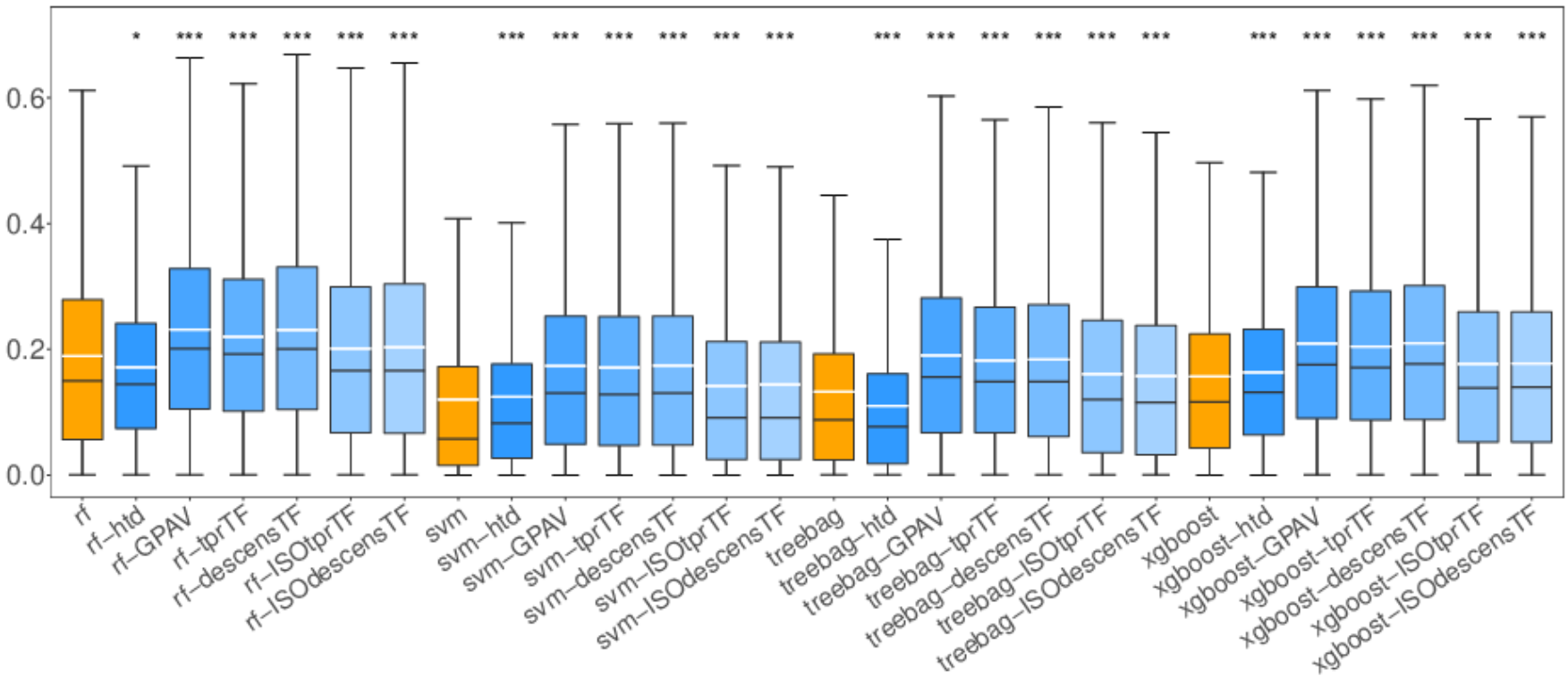
Goal:

- HEM provide consistent predictions with respect the underlying GO ontology
- show that proposed HEM can improve upon flat predictions independently of the choice of the base learner.
  - we chose a range as broad as possible of flat classifier, ranging from linear classifiers (svm), to neural networks (mlp), to ensemble of learning machines (random forest) and to gradient boosting algorithms

Experiments:

- predict the protein function of 6 different model organisms (*D. melanogaster, C.elegans, G.gallus, D.rerio, M. musculus, H. sapiens*) by using the Gene Ontology (GO);
- intensive task: overall we considered over than **100 thousands** of **proteins** and more than **15 thousands** of functional **GO terms**

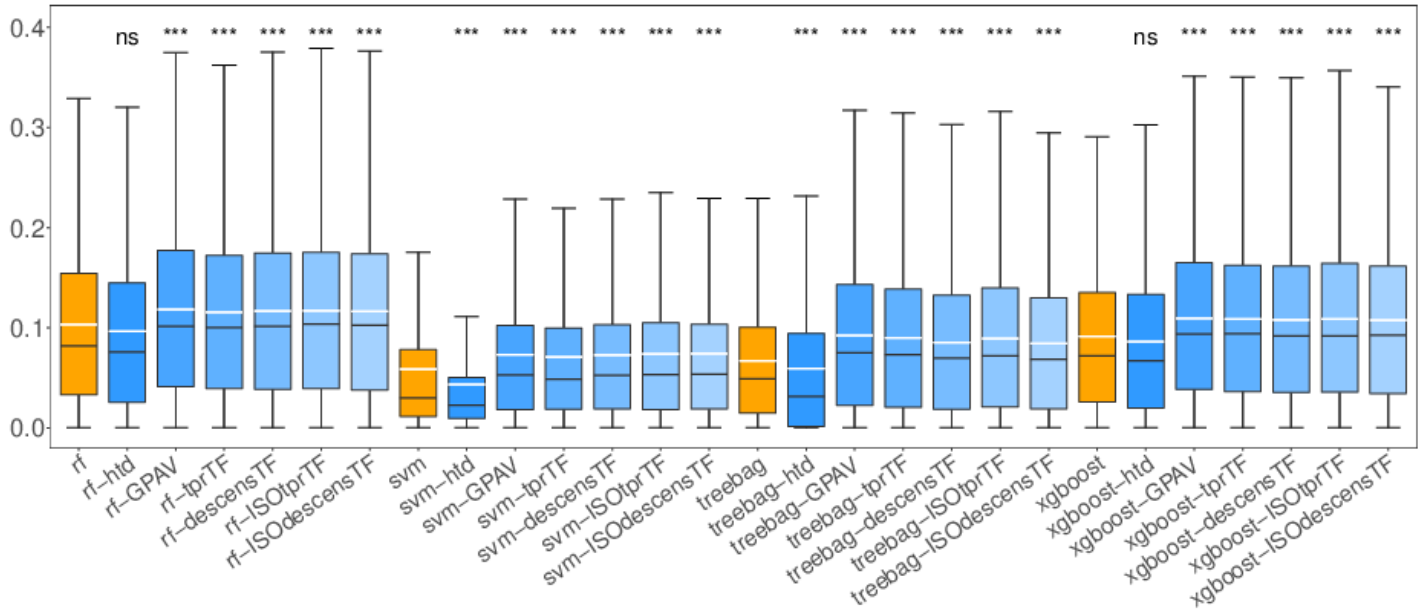
AUPRC boxplot across 760 GO (MF) terms – **Homo Sapiens**



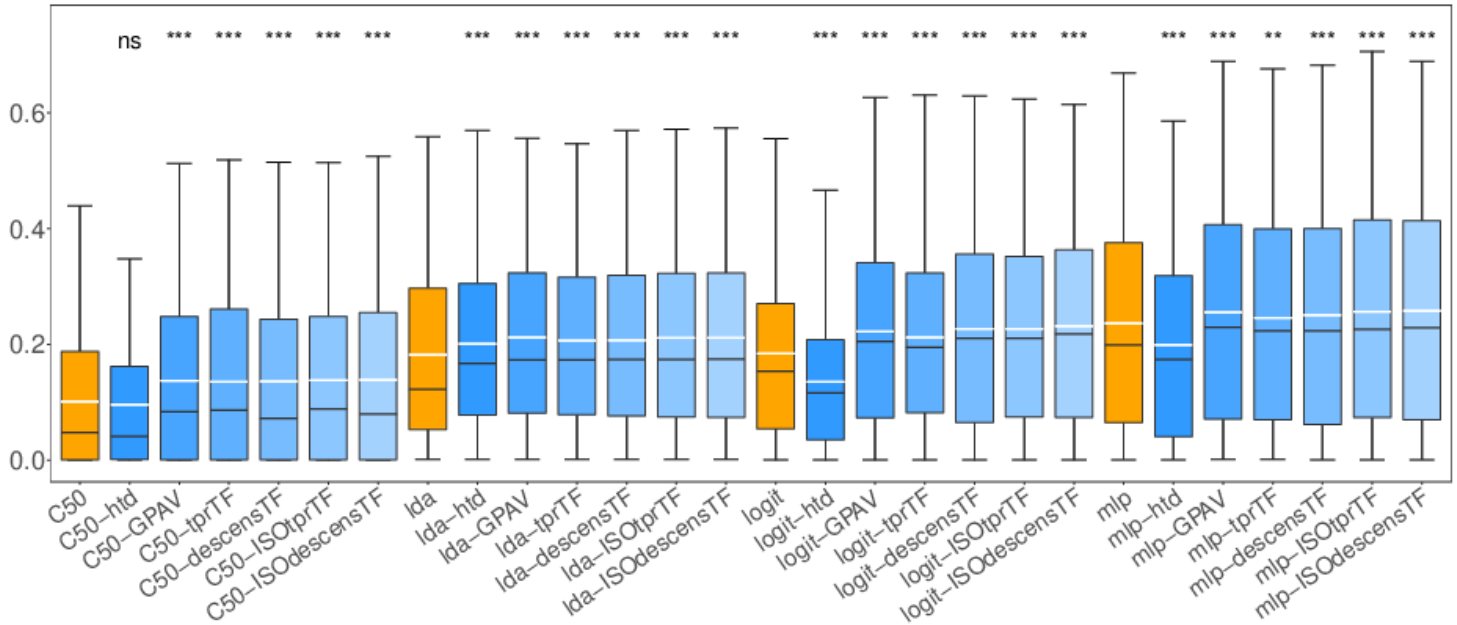
- $\text{pvalue} < 10^{-6} \rightarrow \star\star\star$;
- $\text{pvalue} < 10^{-3} \rightarrow \star\star$;
- $\text{pvalue} < 10^{-2} \rightarrow \star$;
- $\text{pvalue} \geq 10^{-2} \rightarrow$ the difference is not statistically significant (ns);

Paired Wilcoxon Sum Rank Test: Flat vs HEMs

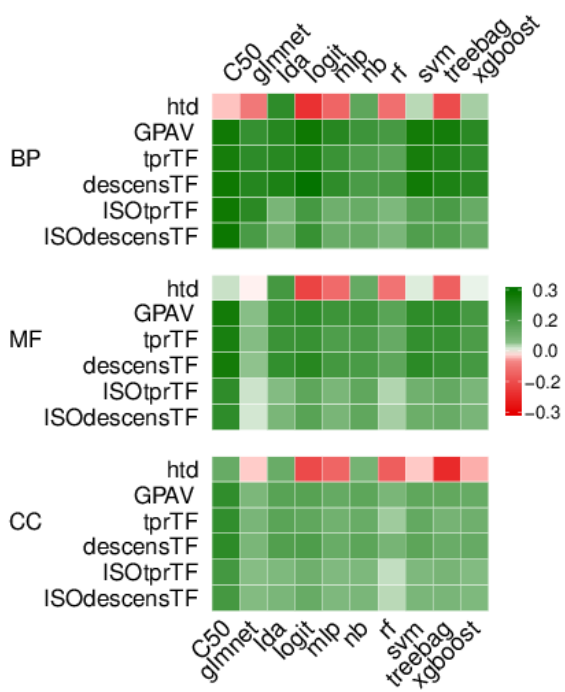
The improvement introduced by HEMs strongly depends on the predictions made by the underlying flat classifier;



D. rerio
GO-BP: 1182



C. elegans
GO-CC: 221

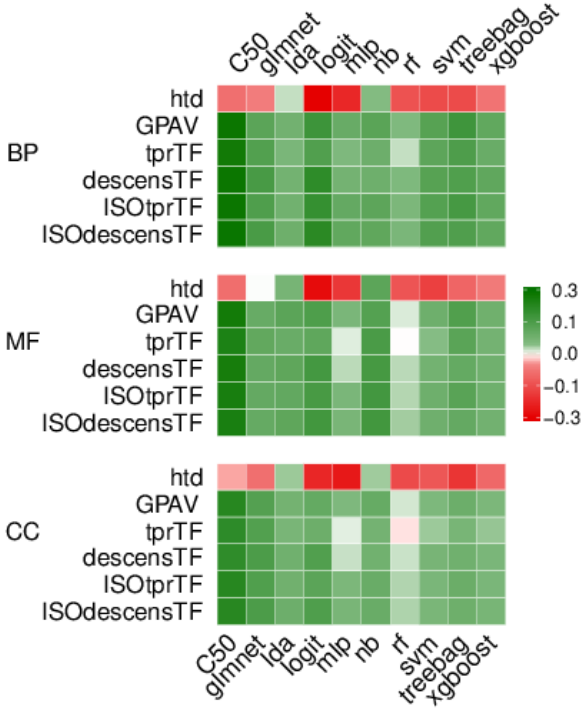


Homo Sapiens

BP terms: 3460
MF terms: 760
CC terms: 541

Mus Musculus

BP terms: 3899
MF terms: 511
CC terms: 445

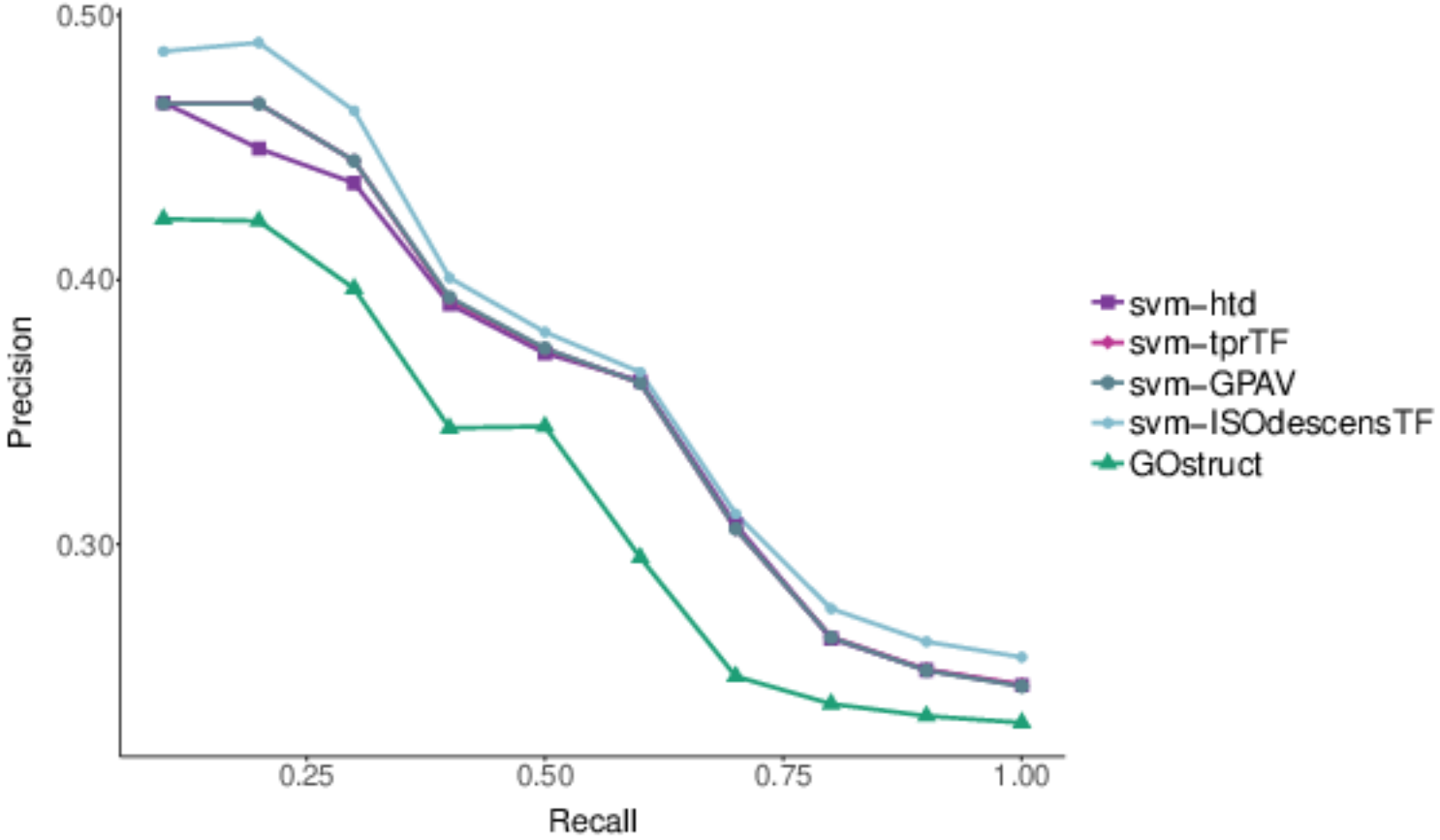


Drosphila Melanogaster

BP terms: 2244
MF terms: 327
CC terms: 348

$$HeatmapCell[i,j] = \frac{\overline{AUPRC}_{hier_j} - \overline{AUPRC}_{flat_i}}{max\left(\overline{AUPRC}_{hier_j}, \overline{AUPRC}_{flat_i}\right)}$$

Organism: Danio Rerio
Compared precision at different recall levels averaged across 89 GO-CC terms



Svm (parallel) + HEMs: 45 seconds
GOstruct: 9 hours

Methodological Results

- **HEMs** are "highly modular" in the sense that they adopt a "two-step" learning strategy: flat predictions + hierarchical correction;
- **HEMs** are characterized either by a single or a double step:
  1. **Bottom-Up step**:
     A. Improve sensitivity of the predictions;
     B. Bottom-up predictions are inconsistent with the hierarchy of the classes;
  2. **Top-Down step**:
     A. Improve precision of the predictions;
     B. Remove hierarchical violations;
- **HEMs** predictions always respect the *True Path Rule* (i.e. consistent with hierarchy of classes)
- **HEMs**: improves flat scores but it cannot of course guarantee the correctness of all the predictions (when e.g. the flat predictions are too bad HEMDAG fails in recovering FP or FN)
- **HEMDAG** is specifically designed for DAG-structured taxonomies, but can be safely applied to tree-structured taxonomies, since trees are DAGs;

Experimental Results

1. **Prediction of HPO terms**

2. **Prediction of GO terms**

- competitive with state-of-the-art results and at lower computational complexity cost;
- predictions of novel gene-abnormal phenotype associations;
- HEMs algorithms systematically improve flat methods;

flexible tool that can be used to virtually improve any flat learning method

References (1)

International Peer Reviewed Journal

1. **M. Notaro**, M. Schubach, P. Robinson, and G. Valentini, *Prediction of Human Phenotype Ontology terms by means of Hierarchical Ensemble methods*, BMC Bioinformatics, 18(1):449, 2017. **Note: *awarded by the International Medical Informatics Association (IMIA) as one of the five best "Knowledge Representation and Management" papers of 2017 in the field of Medical Informatics***
2. **M.Notaro**, M. Frasca, A. Petrini, G. Valentini, HEMDAG: a scalable and flexible state-of-the-art tool outperforming flat learning predictions (note: manuscript in preparation)

Poster Presentation at International Conference

2. **M. Notaro**, M. Schubach, P. Robinson, and G. Valentini, *Predicting new relationships between genes and Human Phenotype Ontology terms*, ISMB 2018: 26th International conference on intelligent systems for molecular biology, 6-10 July, Chicago, United States, 2018

References (2)

Proceedings of International Conferences and Peer-Reviewed Book Chapters

3. **M. Notaro**, M. Schubach, P.N. Robinson, G. Valentini, *Ensembling Descendant Term Classifiers to Improve Gene - Abnormal Phenotype Predictions*, In Massimo Bartoletti, Annalisa Barla, Andrea Bracciali, Gunnar W. Klau, Leif Peterson, Alberto Policriti, and Roberto Tagliaferri, editors, *Computational Intelligence Methods for Bioinformatics and Biostatistics*, pages 70–80, Cham, 2019. Springer International Publishing
4. P.N. Robinson, M.Frasca, S. Köhler, **M. Notaro**, M. Re, G. Valentini, *A Hierarchical Ensemble Method for DAG-Structured Taxonomies*, Lecture Notes in Computer Science, vol. 9132, pp. 15–26. Berlin: Springer, 2015
5. G. Valentini, S. Köhler, M. Re, **M. Notaro**, P.N. Robinson*, Prediction of Human Gene-Phenotype Associations by Exploiting the Hierarchical Structure of the Human Phenotype Ontology*, Lecture Notes in Computer Science, vol. 9043, pp. 66–77. Cham: Springer, 2015.