Decision Trees and Random Forests Modeling by using P Systems

José M. Sempere
Department of Information Systems and Computation
Universitat Politècnica de València

jsempere@dsic.upv.es

http://personales.upv.es/jsempere

Goals of this work: (1) Provide algorithms to define P systems from decision trees and random forests
(2) Introduce machine learning techniques to obtain decision trees and random forests from given data through membrane and object rules based on P systems working in entropic/functional manner

Previous Works

- Decision Tree Models Induced by Membrane Systems (2015) J. Wang, J. Hu, M.J. Pérez-Jiménez, A.Riscos-Núñez ROMJIST Vol.18 No.3 pp 228-239

- Self-constructing Recognizer P Systems. D. Díaz-Pernil, F. Peña-Cantillana, M.A. Gutiérrez-Naranjo. In Proceedings of the Thirteenth Brainstorming Week on Membrane Computing , pp 137-154. Fenix Editora. 2014.

Díaz-Pernil *et al*.

Trees defined by the membrane structure
Non-deterministic search of structures

Wang *et al.*

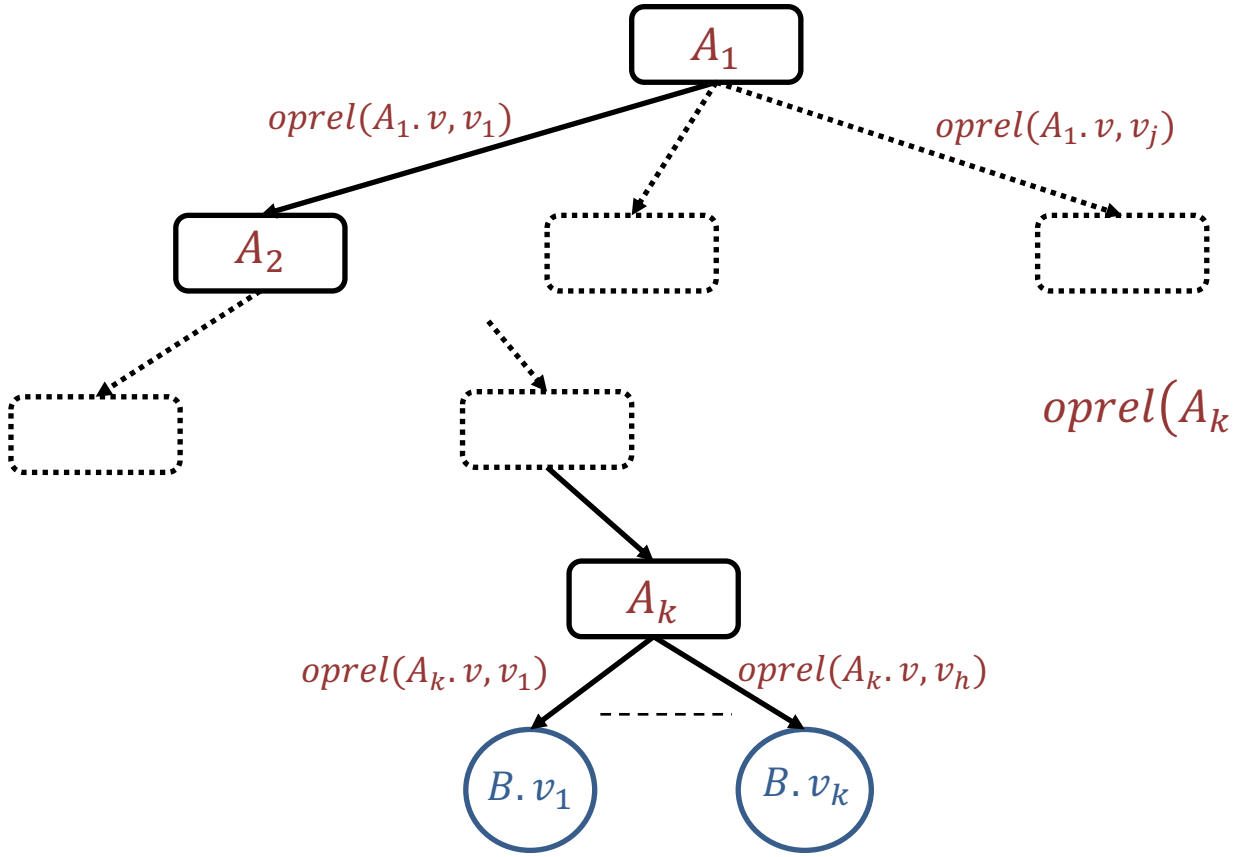
Tree-like objects
External Induction Algorithm

Our approach

Trees defined by the membrane structure
Algorithm runs by P rules within an entropic manner

A Decision Tree is a representation for a discrete-valued function

$$f: A_1 \times A_2 \times \cdots \times A_n \rightarrow B$$



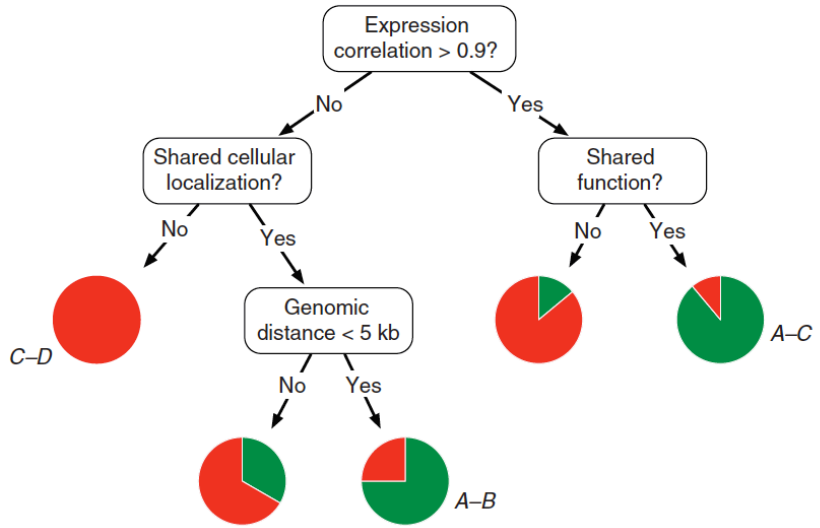
$$oprel(A_k.v, v_p) \in \{A_k.v = v_p, A_k.v \leq v_p, A_k.v \geq v_p, A_k.v \neq v_p, \ldots\}$$

An example: protein-protein interactions prediction

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



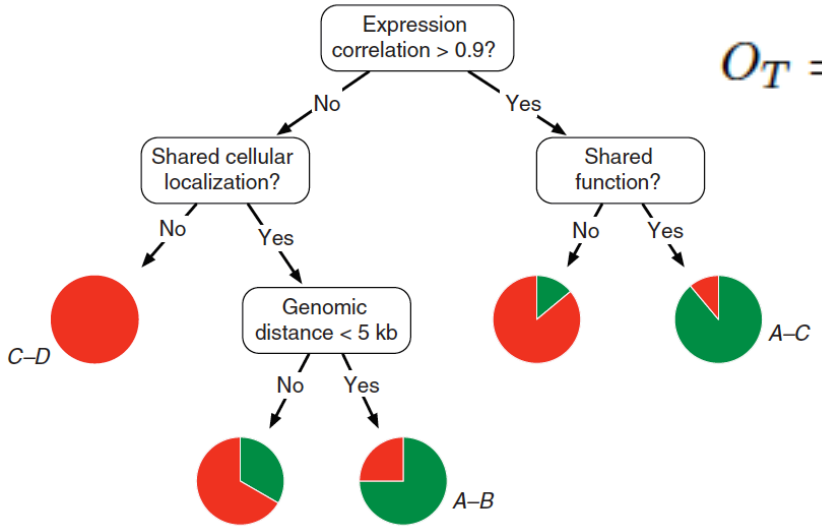
(From **"What are decision trees ?". Carl Kingsford & Steven L. Salzberg. Nature Biotechnology 26 No. 9, 1011 – 1013 (2008)**)

Implementing decision trees by membrane structures (I)

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



Objects alphabet

$$O_T = \{EC_{\leq 0.7}, EC_{>0.7}, SL_Y, SL_N, SF_Y, SF_N, GD_{i_1}, GD_{i_2}, YES, NO\}$$

$$(EC_{\leq 0.7} + EC_{>0.7})(SL_Y + SL_N)(GD_Y + GD_N)(SF_Y + SF_N)$$

Membrane labels alphabet

$$H_T = \{EC, SL, SF, GD\}$$

Implementing decision trees by membrane structures (I)

An algorithm to translate decision trees to P systems

CreateInternalRules($T, \mathcal{A}$)

/* Let $a_i \in \mathcal{A}$ be the (internal) attribute of the root of T */

(1) For every branch with value v_j that goes to (internal) attribute b in *preorder traversal*

(1.1) For every string $x \in (\widehat{\mathcal{A}} - \{a_i\})_{reg}$

(1.1.1) add the rule $[a_{ij}x[\ ]_b]_{a_i} \rightarrow [\ [x]_b]_{a_i}$

/* Let T_b the subtree of T rooted in the attribute b */

(1.2) CreateRules(T_b,$\mathcal{A} - \{a_i\}$)

(2) For every branch with value v_j that goes to class attribute c with value c_k

(2.1) For every string $x \in (\widehat{\mathcal{A}} - \{a_i\})_{reg}$

(2.1.1) add the rule $[a_{ij}x]_{a_i} \rightarrow [\ ]_{a_i}c_k$

CreateClassificationRules($\mathcal{A}$)

---

(1) For every attribute $a \in \mathcal{A}$

(1.1) For every value c_k of the class attribute c

(1.1.1) add the rule $[c_k]_{a_i} \rightarrow [\ ]_{a_i}c_k$

Implementing decision trees by membrane structures (I)

An algorithm to translate decision trees to P systems

1. $[EC_{\leq 0.7} SL_Y GD_Y SF_Y [\,]_{SL}\,]_{EC} \rightarrow [\,[SL_Y GD_Y SF_Y]_{SL}\,]_{EC}$
2. $[EC_{\leq 0.7} SL_Y GD_Y SF_N [\,]_{SL}\,]_{EC} \rightarrow [\,[SL_Y GD_Y SF_N]_{SL}\,]_{EC}$
3. $[EC_{\leq 0.7} SL_Y GD_N SF_Y [\,]_{SL}\,]_{EC} \rightarrow [\,[SL_Y GD_N SF_Y]_{SL}\,]_{EC}$
4. $[EC_{\leq 0.7} SL_Y GD_N SF_N [\,]_{SL}\,]_{EC} \rightarrow [\,[SL_Y GD_N SF_N]_{SL}\,]_{EC}$
5. $[EC_{\leq 0.7} SL_N GD_Y SF_Y [\,]_{SL}\,]_{EC} \rightarrow [\,[SL_N GD_Y SF_Y]_{SL}\,]_{EC}$
6. $[EC_{\leq 0.7} SL_N GD_Y SF_N [\,]_{SL}\,]_{EC} \rightarrow [\,[SL_N GD_Y SF_N]_{SL}\,]_{EC}$
7. $[EC_{\leq 0.7} SL_N GD_N SF_Y [\,]_{SL}\,]_{EC} \rightarrow [\,[SL_N GD_N SF_Y]_{SL}\,]_{EC}$
8. $[EC_{\leq 0.7} SL_N GD_N SF_N [\,]_{SL}\,]_{EC} \rightarrow [\,[SL_N GD_N SF_N]_{SL}\,]_{EC}$

9. $[SL_Y SF_Y GD_Y [\,]_{GD}\,]_{SL} \rightarrow [\,[SF_Y GD_Y]_{GD}\,]_{SL}$
10. $[SL_Y SF_Y GD_N [\,]_{GD}\,]_{SL} \rightarrow [\,[SF_Y GD_N]_{GD}\,]_{SL}$
11. $[SL_Y SF_N GD_Y [\,]_{GD}\,]_{SL} \rightarrow [\,[SF_N GD_Y]_{GD}\,]_{SL}$
12. $[SL_Y SF_N GD_N [\,]_{GD}\,]_{SL} \rightarrow [\,[SF_N GD_N]_{GD}\,]_{SL}$

Implementing decision trees by membrane structures (I)

An algorithm to translate decision trees to P systems

13. $[SL_N SF_N GD_N\ ]_{SF} \rightarrow [\ ]_{SF} NO$
14. $[SL_N SF_N GD_Y\ ]_{SF} \rightarrow [\ ]_{SF} NO$
15. $[SL_N SF_Y GD_N\ ]_{SF} \rightarrow [\ ]_{SF} YES$
16. $[SL_N SF_Y GD_Y\ ]_{SF} \rightarrow [\ ]_{SF} YES$

17. $[EC_{>0.7} SL_Y SF_Y GD_Y [\ ]_{SF}\ ]_{EC} \rightarrow [\ [SL_Y SF_Y GD_Y]_{SF}\ ]_{EC}$
18. $[EC_{>0.7} SL_Y SF_Y GD_N [\ ]_{SF}\ ]_{EC} \rightarrow [\ [SL_Y SF_Y GD_N]_{SF}\ ]_{EC}$
19. $[EC_{>0.7} SL_Y SF_N GD_Y [\ ]_{SF}\ ]_{EC} \rightarrow [\ [SL_Y SF_N GD_Y]_{SF}\ ]_{EC}$

20. $[EC_{>0.7} SL_Y SF_N GD_N [\ ]_{SF}\ ]_{EC} \rightarrow [\ [SL_Y SF_N GD_N]_{SF}\ ]_{EC}$
21. $[EC_{>0.7} SL_N SF_Y GD_Y [\ ]_{SF}\ ]_{EC} \rightarrow [\ [SL_N SF_Y GD_Y]_{SF}\ ]_{EC}$
22. $[EC_{>0.7} SL_N SF_Y GD_N [\ ]_{SF}\ ]_{EC} \rightarrow [\ [SL_N SF_Y GD_N]_{SF}\ ]_{EC}$
23. $[EC_{>0.7} SL_N SF_N GD_Y [\ ]_{SF}\ ]_{EC} \rightarrow [\ [SL_N SF_N GD_Y]_{SF}\ ]_{EC}$
24. $[EC_{>0.7} SL_N SF_N GD_N [\ ]_{SF}\ ]_{EC} \rightarrow [\ [SL_N SF_N GD_N]_{SF}\ ]_{EC}$

Implementing decision trees by membrane structures (I)

An algorithm to translate decision trees to P systems

25. $[SF_Y GD_Y\ ]_{GD} \rightarrow [\ ]_{GD} YES$
26. $[SF_N GD_Y\ ]_{GD} \rightarrow [\ ]_{GD} YES$
27. $[SF_Y GD_N\ ]_{GD} \rightarrow [\ ]_{GD} NO$
28. $[SF_N GD_N\ ]_{GD} \rightarrow [\ ]_{GD} NO$

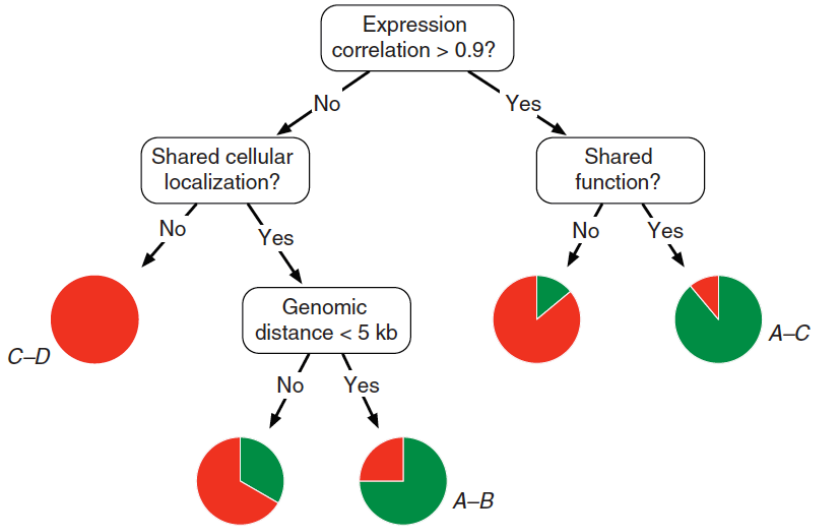
29. $[NO]_{SL} \rightarrow [\ ]_{SL} NO$
30. $[YES]_{SL} \rightarrow [\ ]_{SL} YES$
31. $[NO]_{EC} \rightarrow [\ ]_{EC} NO$
32. $[YES]_{EC} \rightarrow [\ ]_{EC} YES$

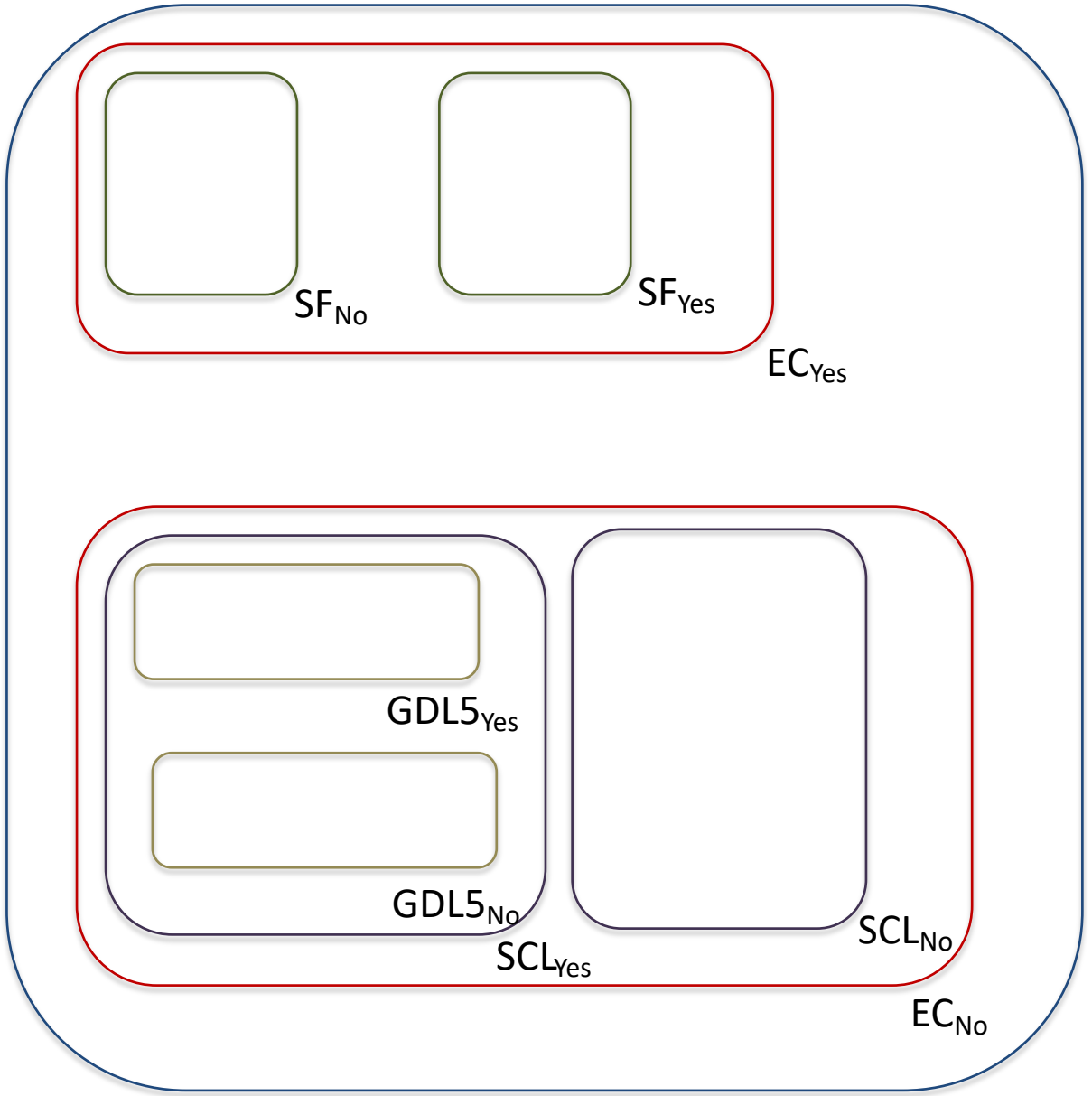
Implementing decision trees by membrane structures (II)

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



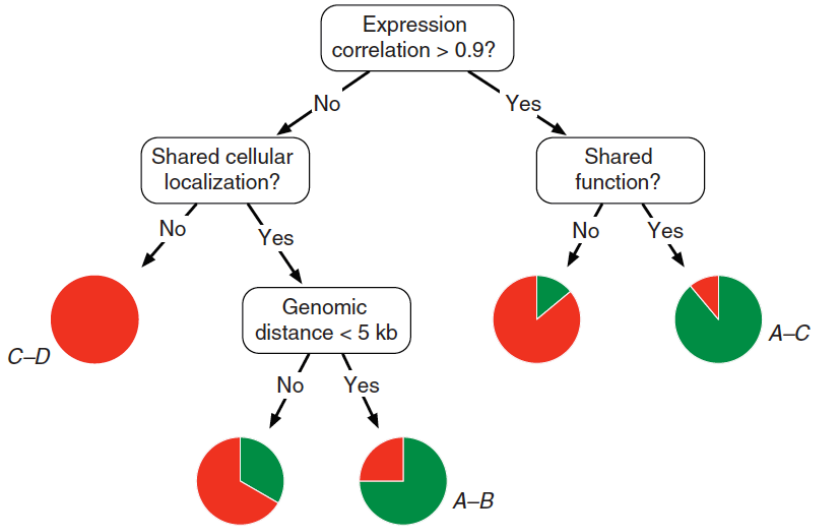


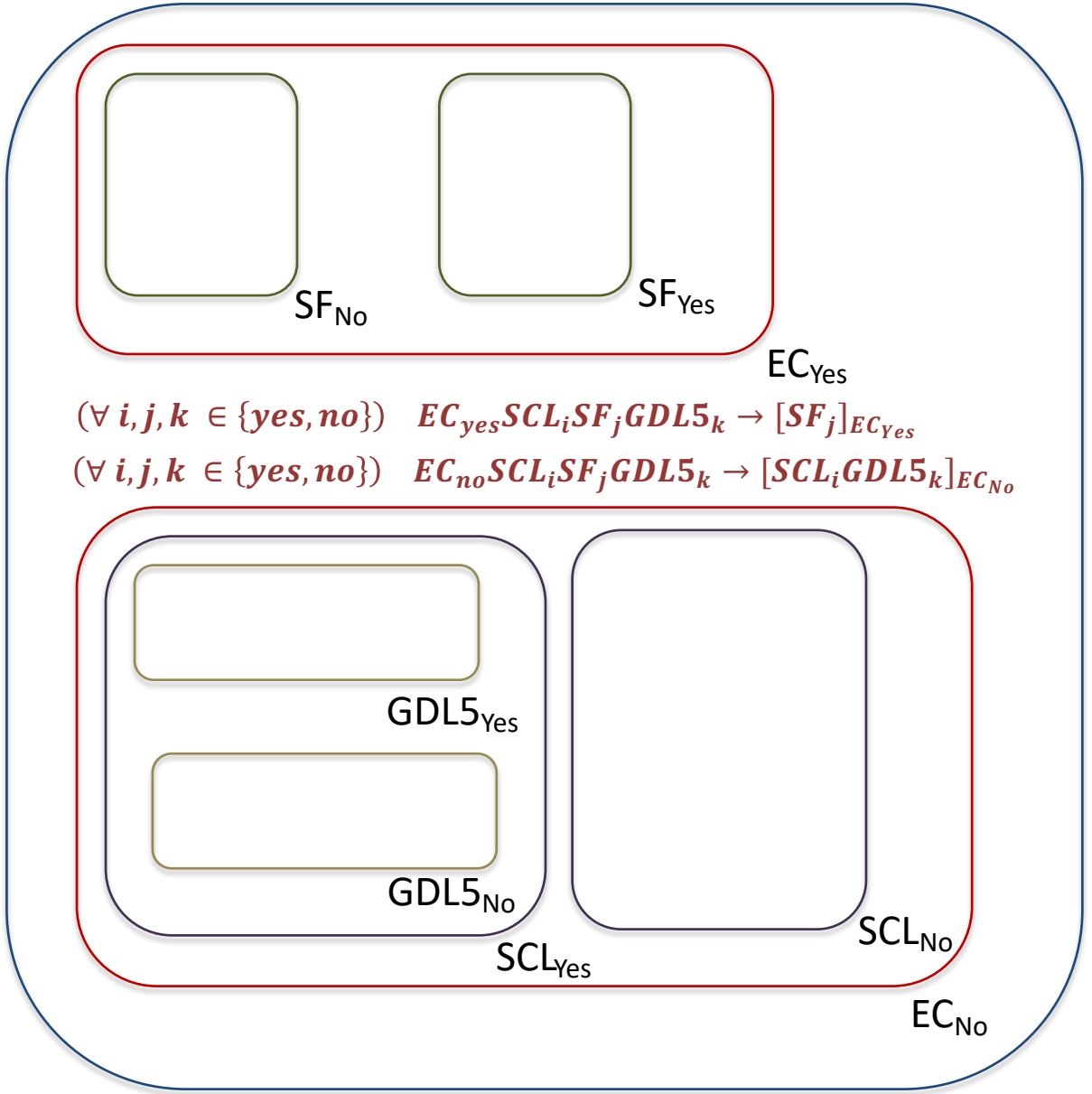
Implementing decision trees by membrane structures (III)

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



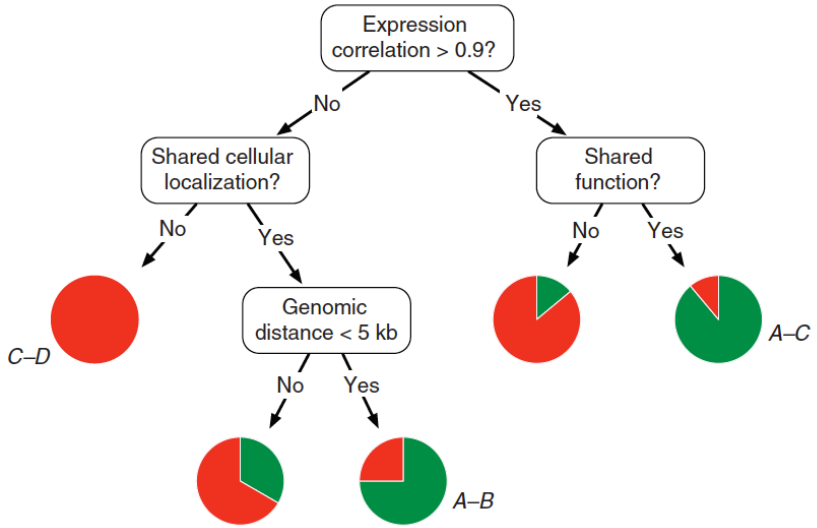


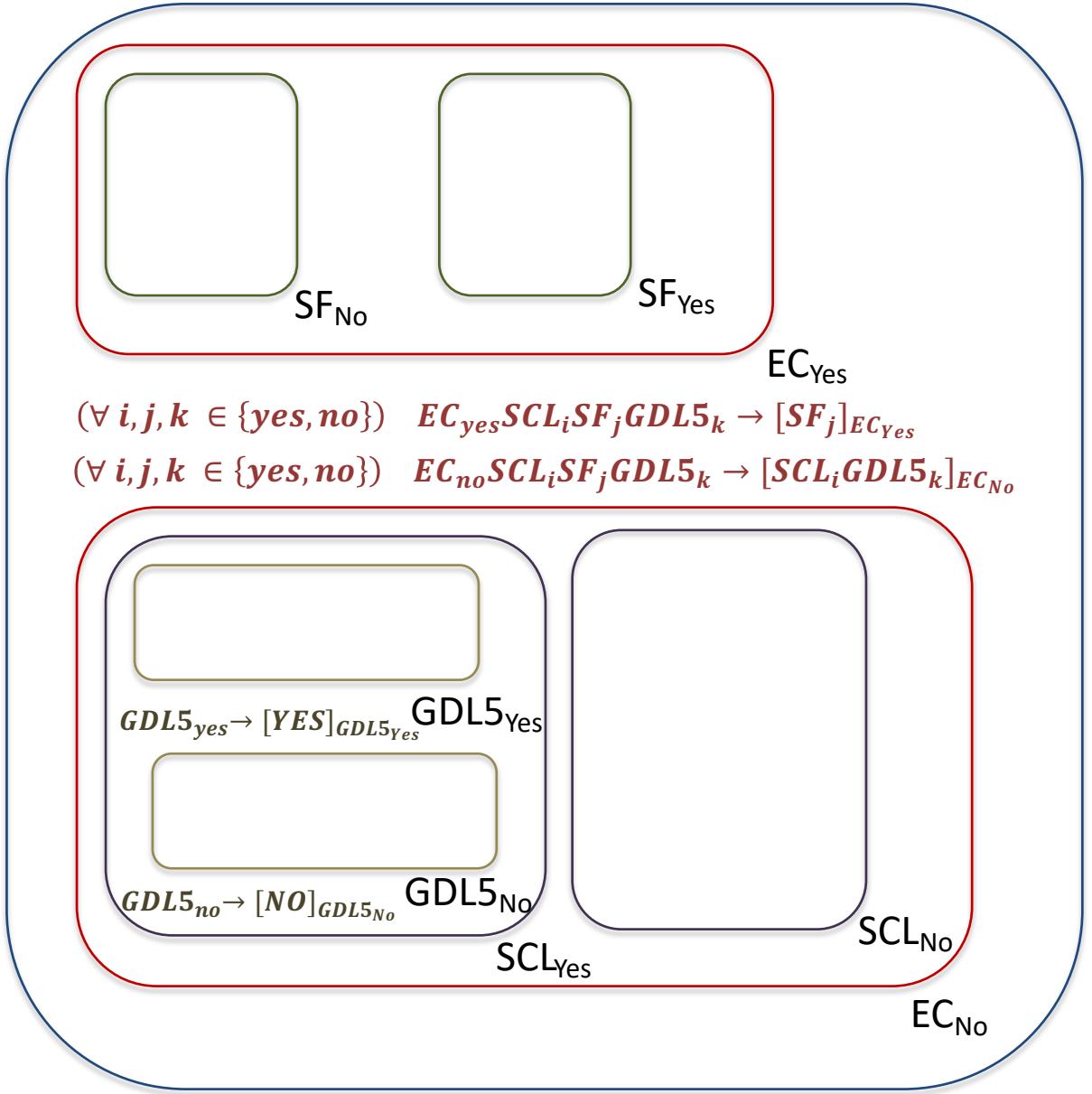
Implementing decision trees by membrane structures (IV)

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



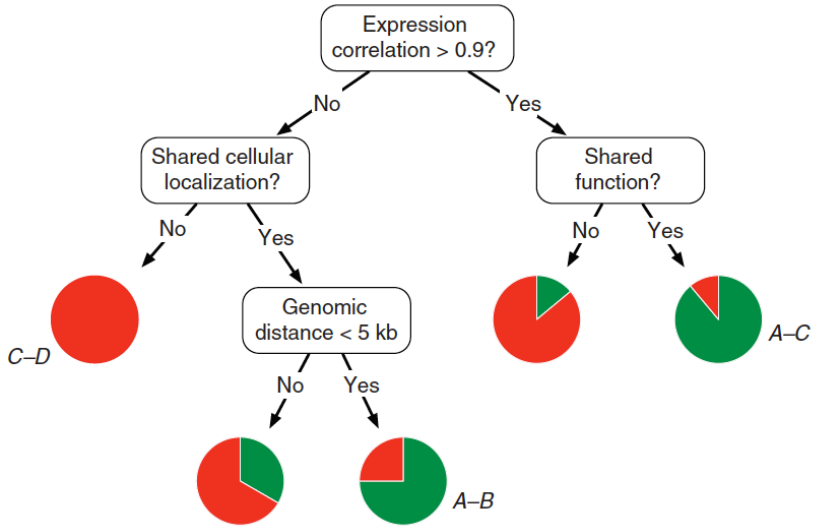


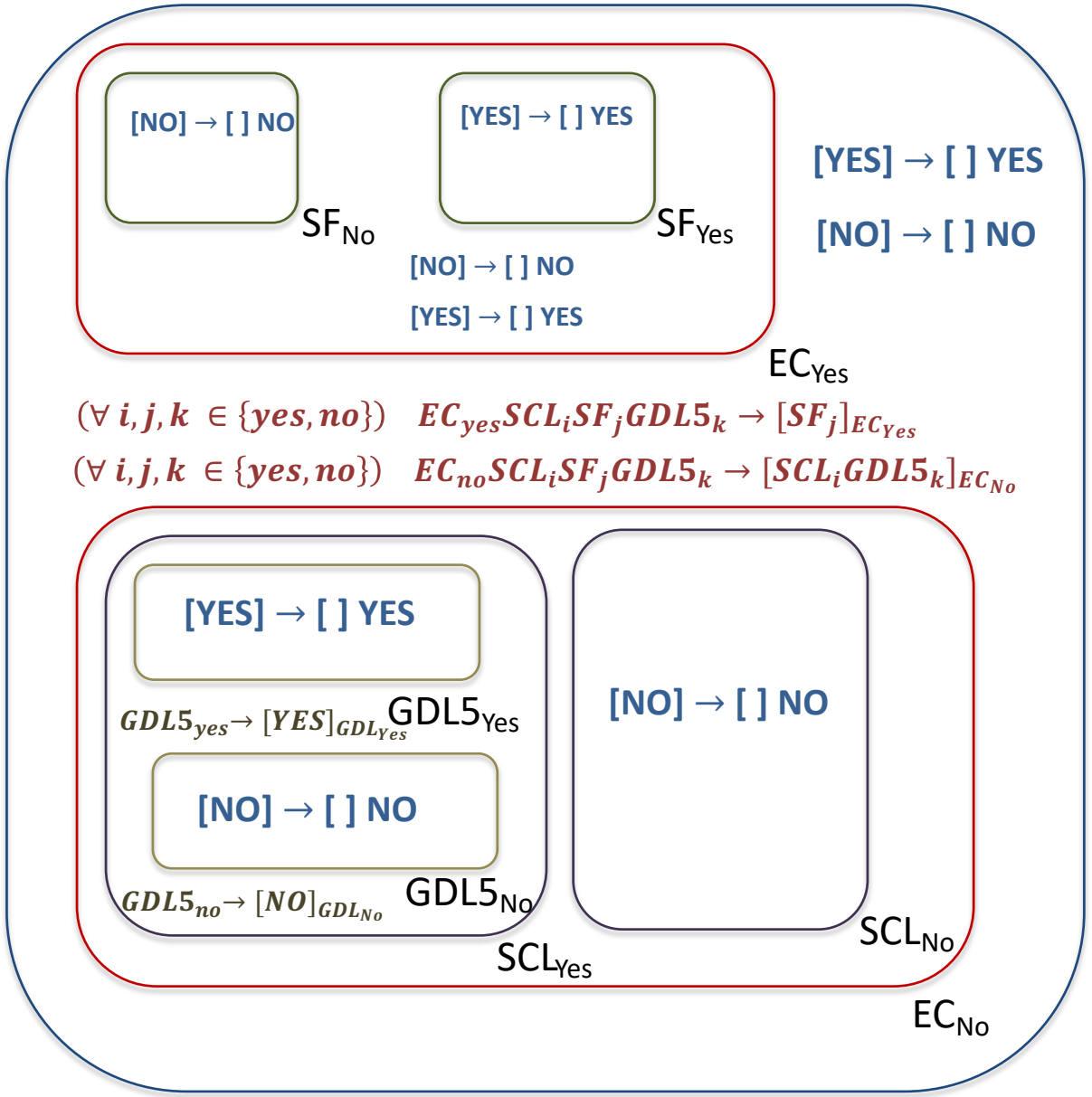
Implementing decision trees by membrane structures (V)

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



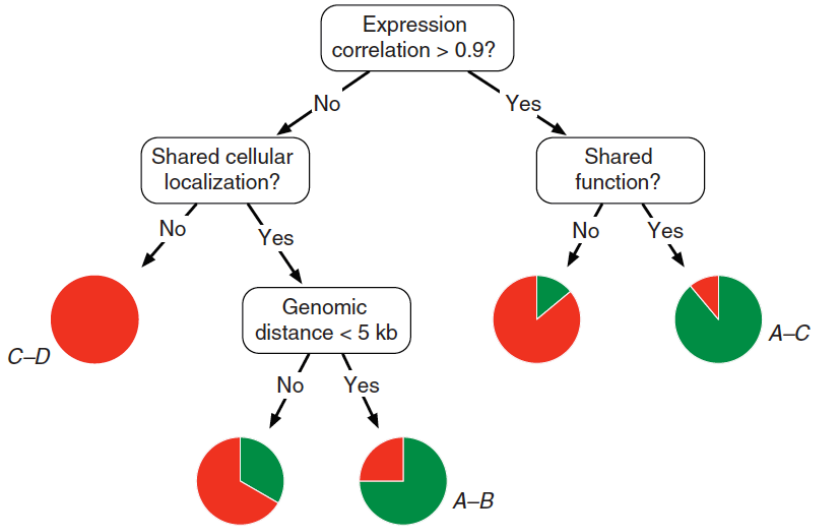


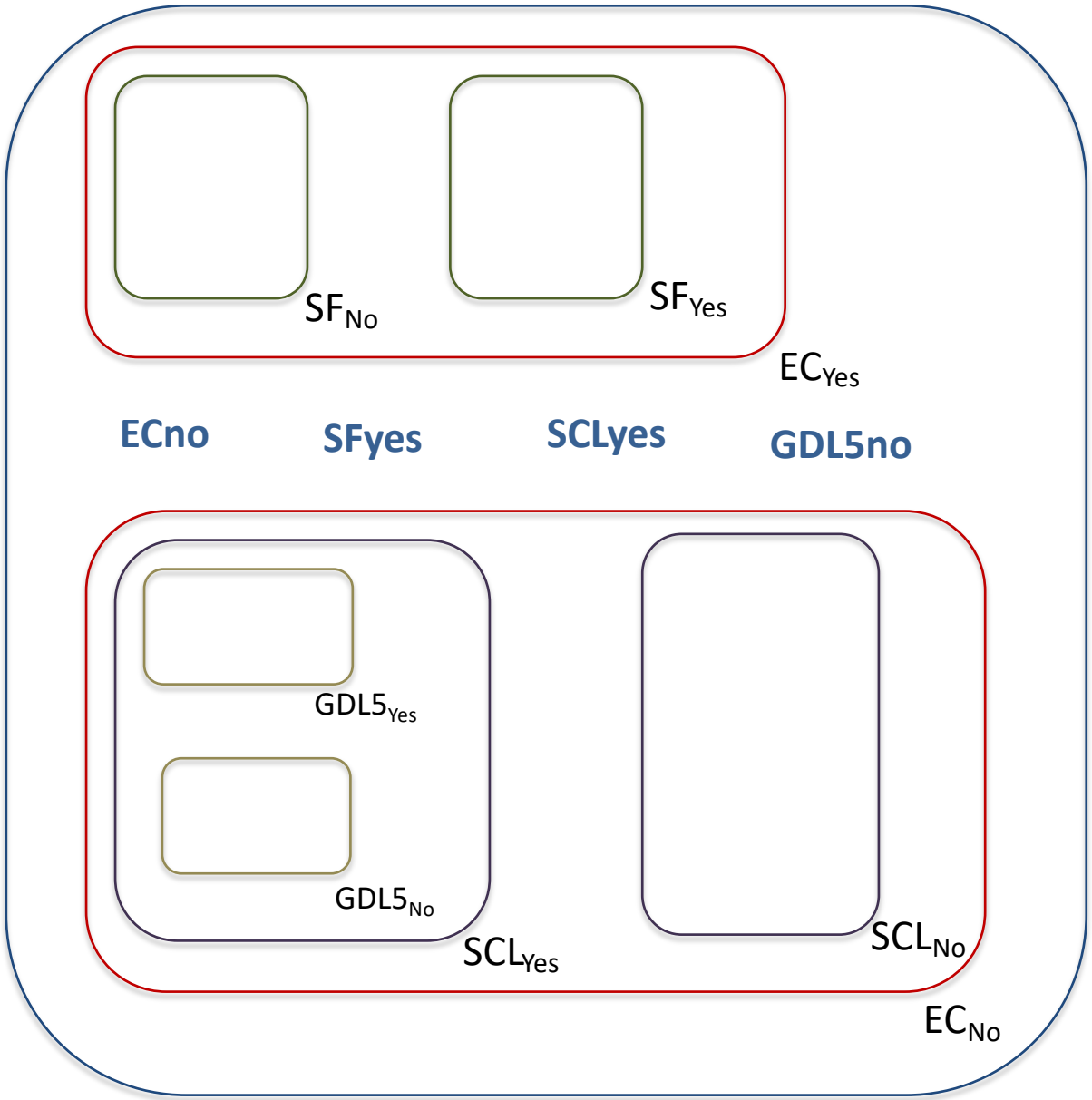
A parsing example

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



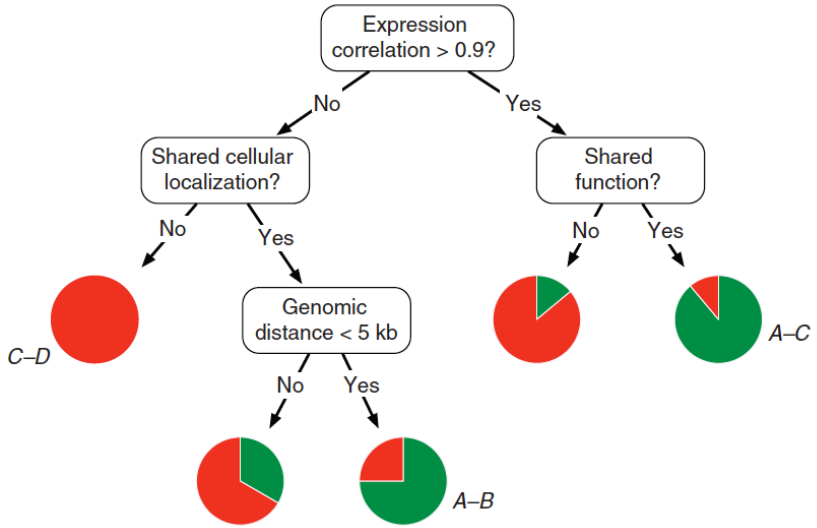


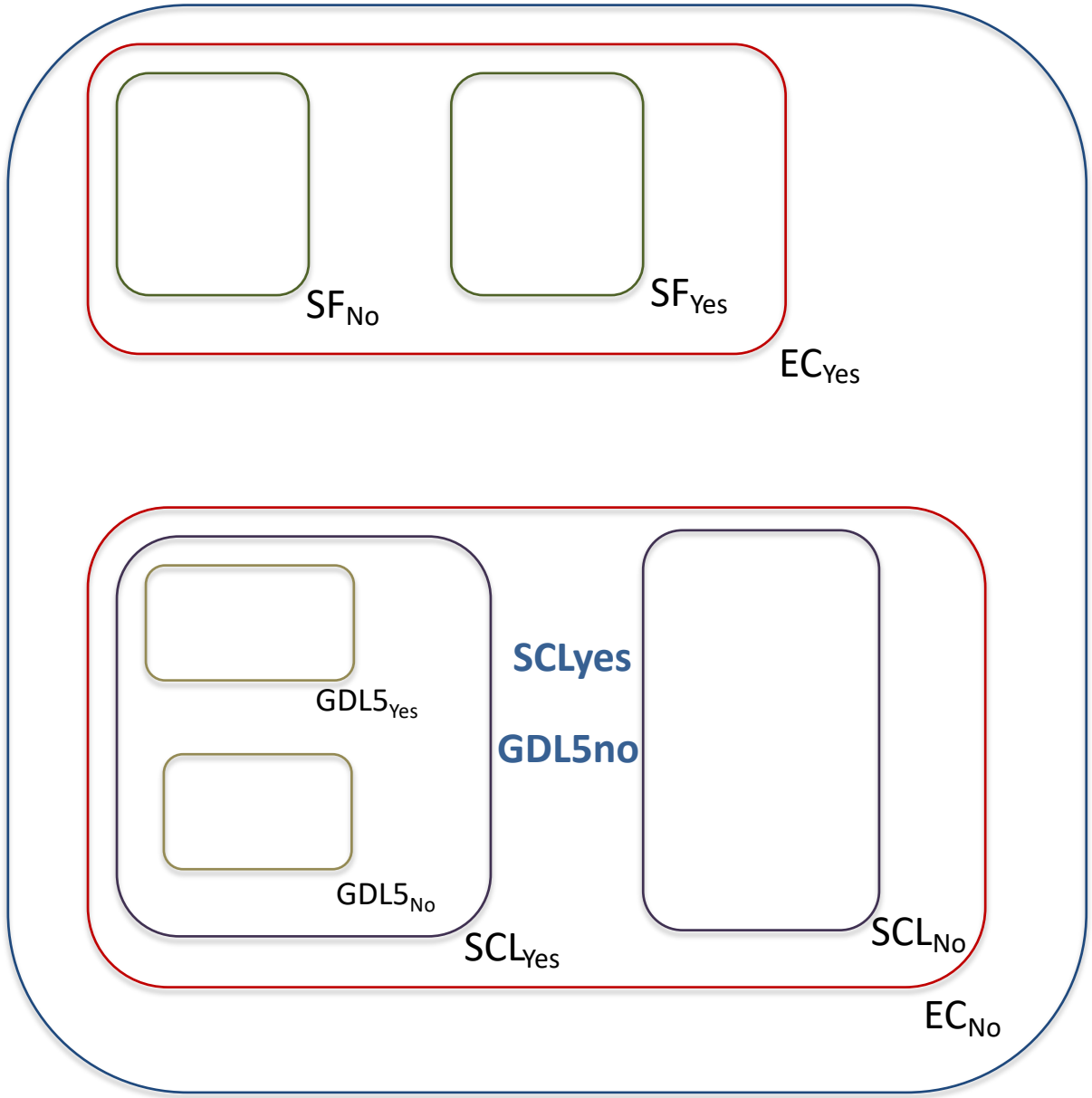
A parsing example

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



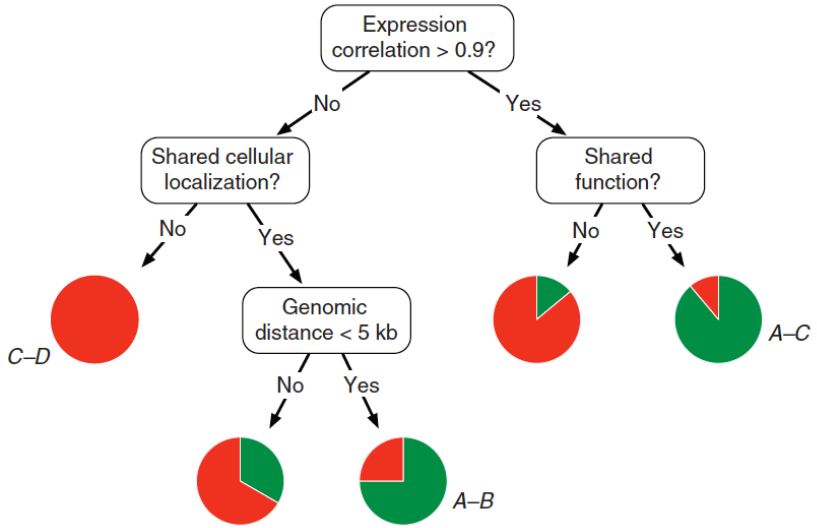


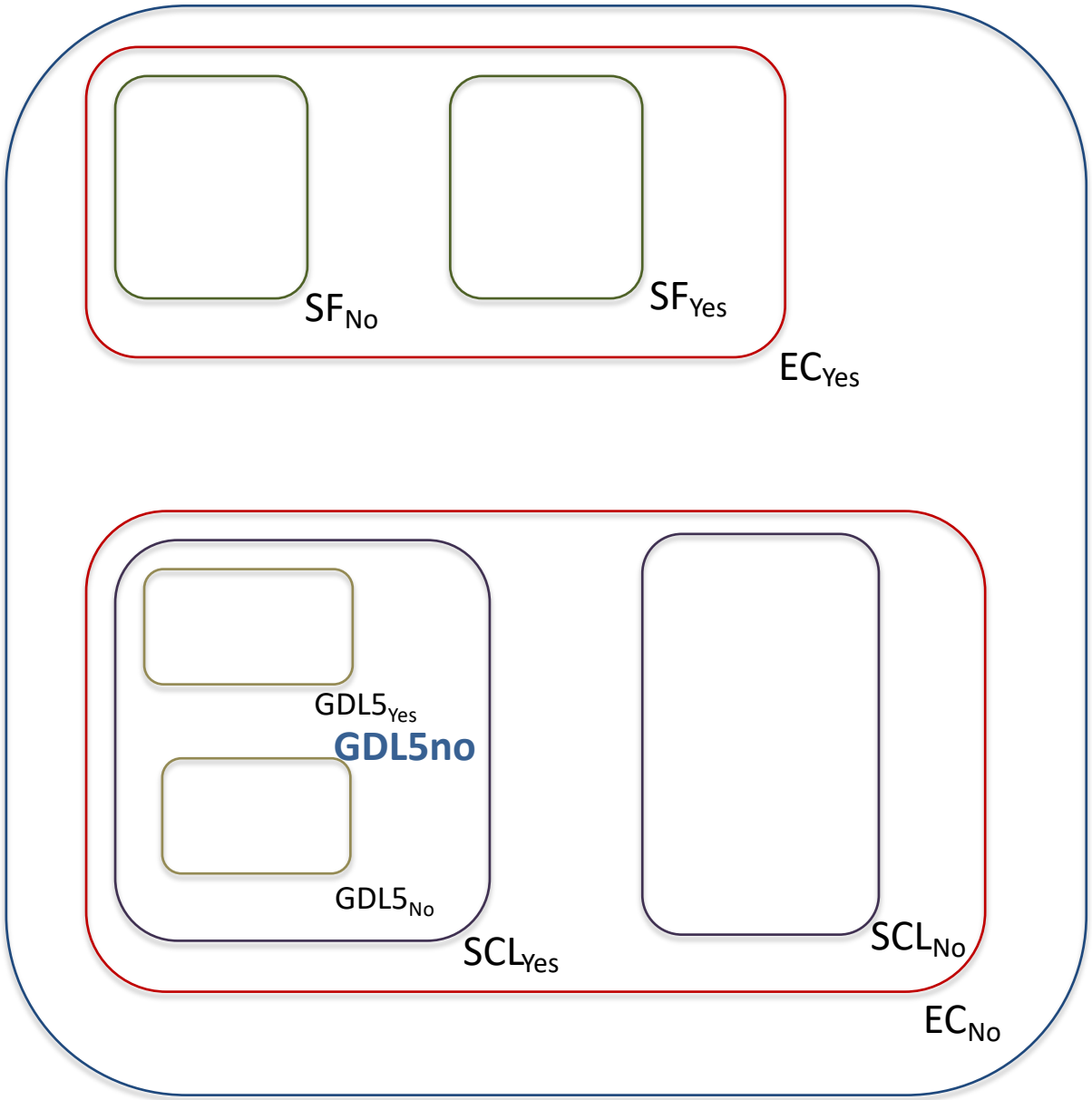
A parsing example

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



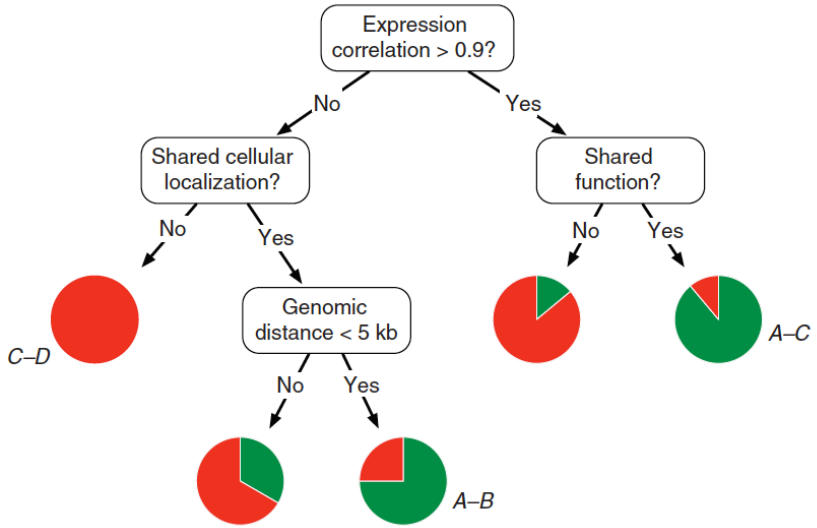


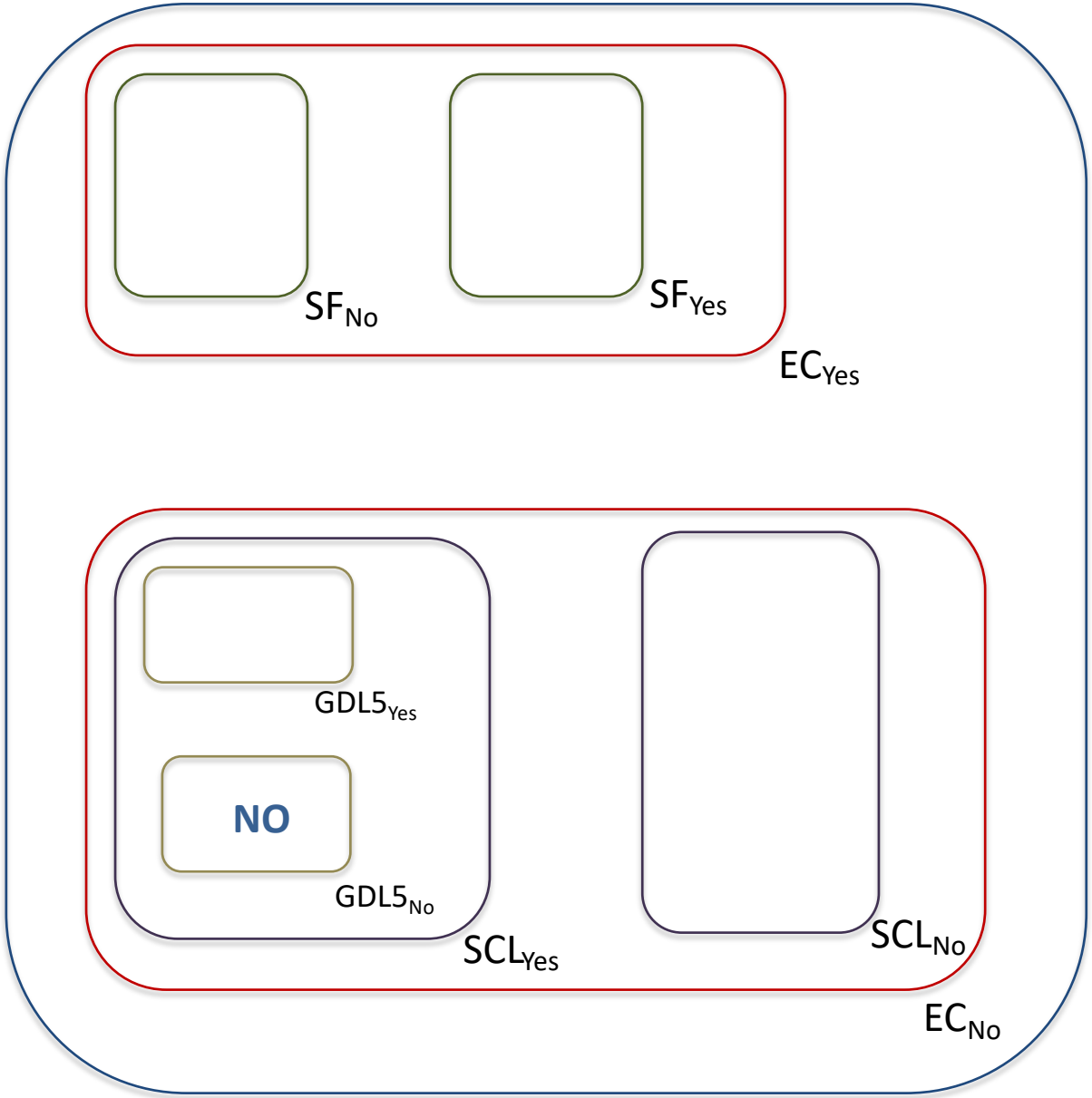
A parsing example

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



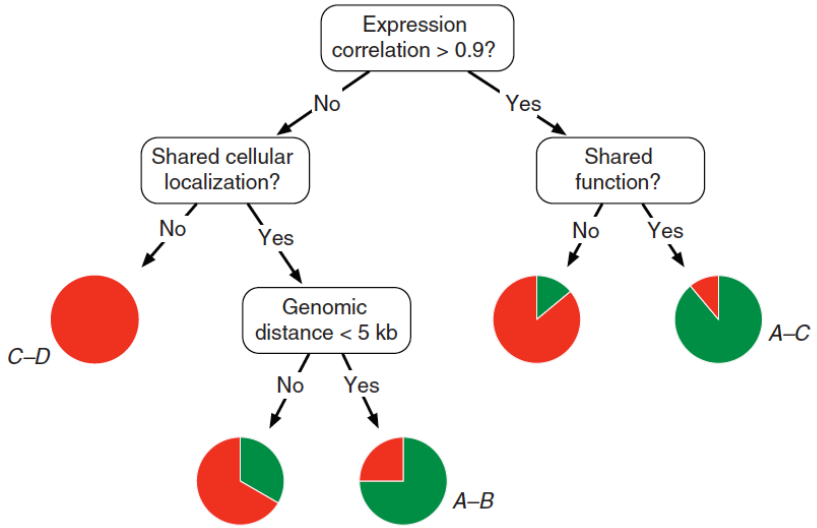


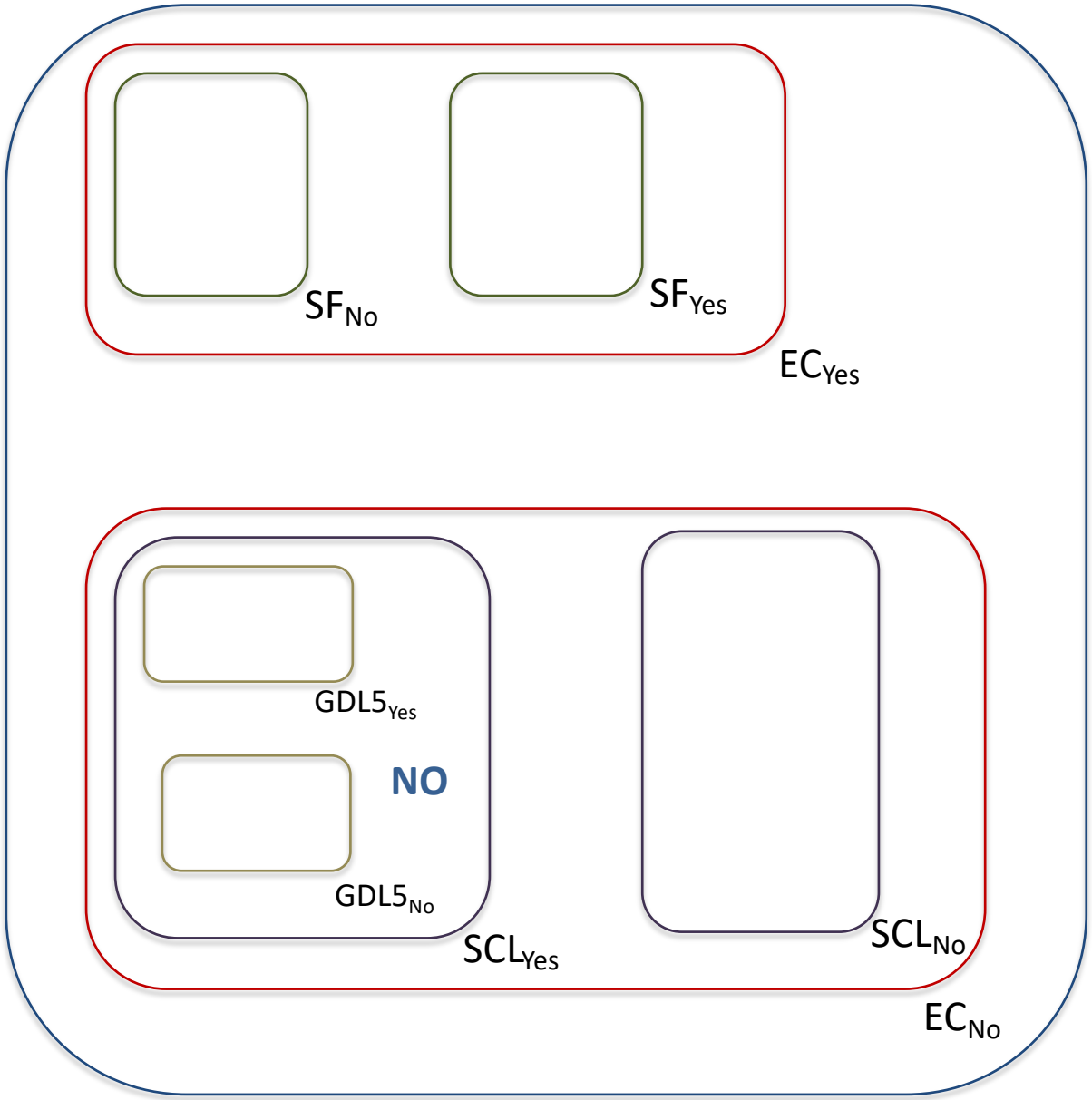
A parsing example

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



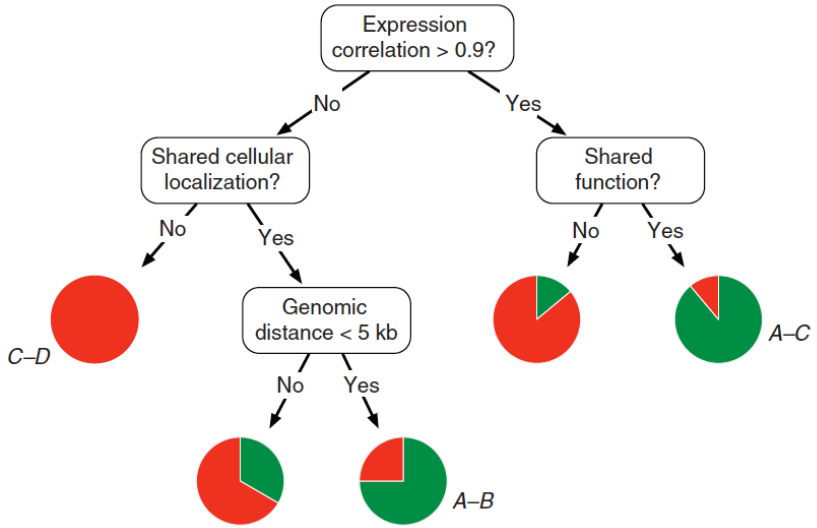


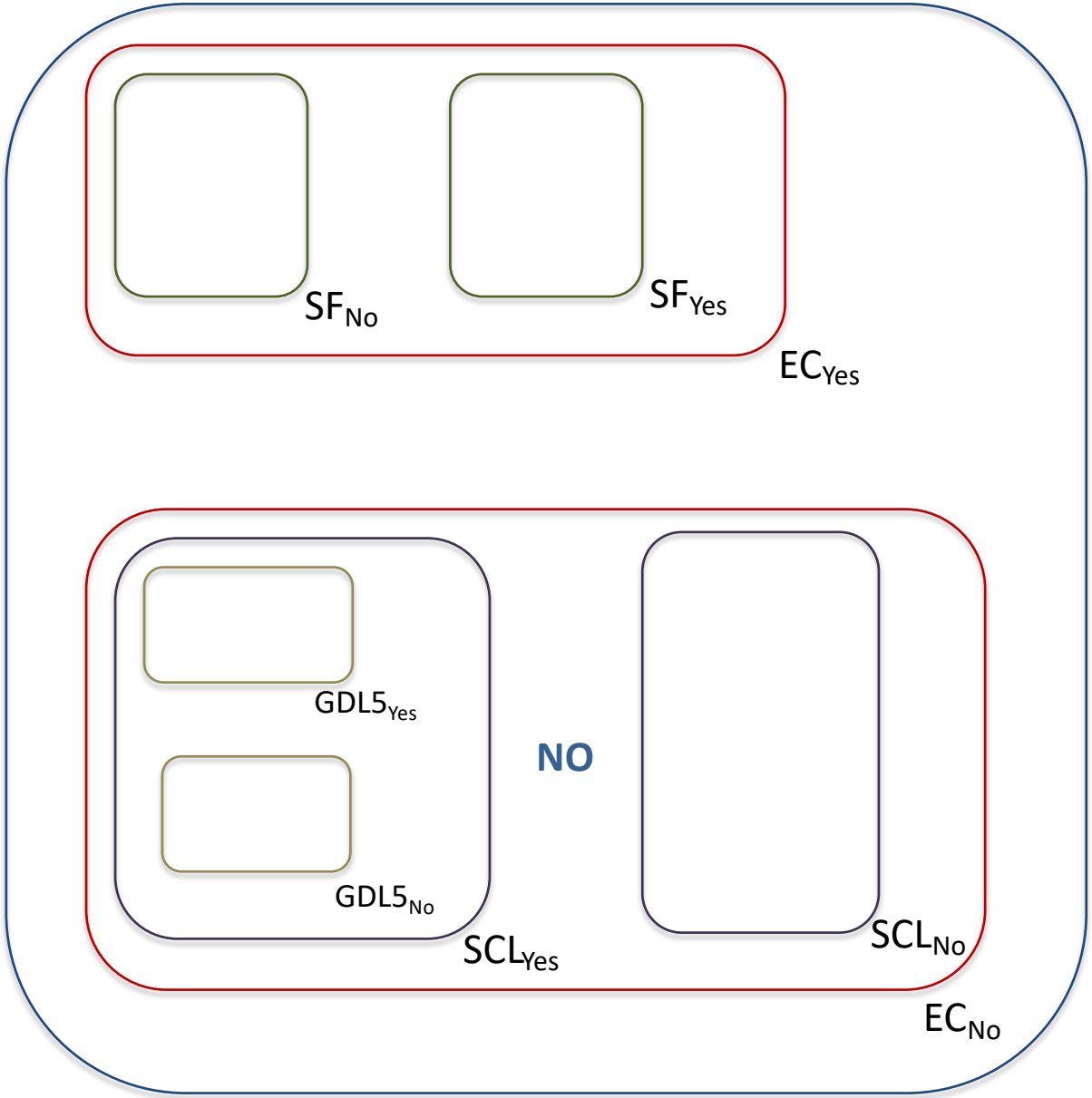
A parsing example

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



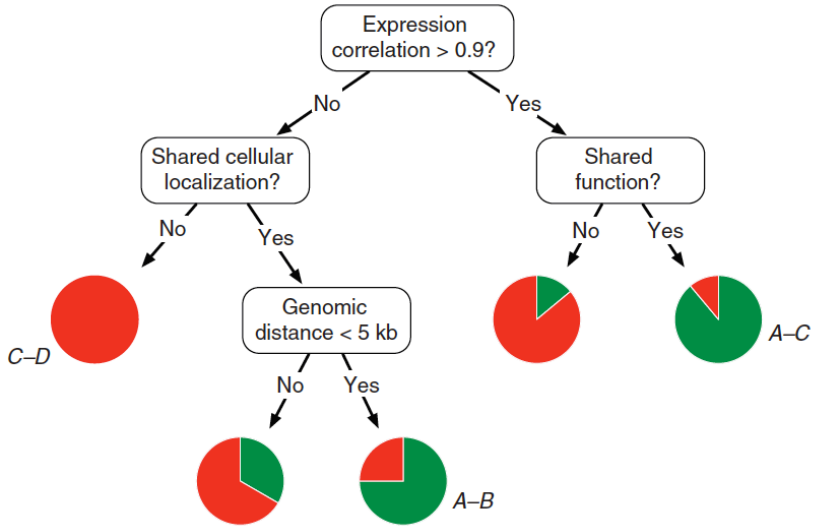


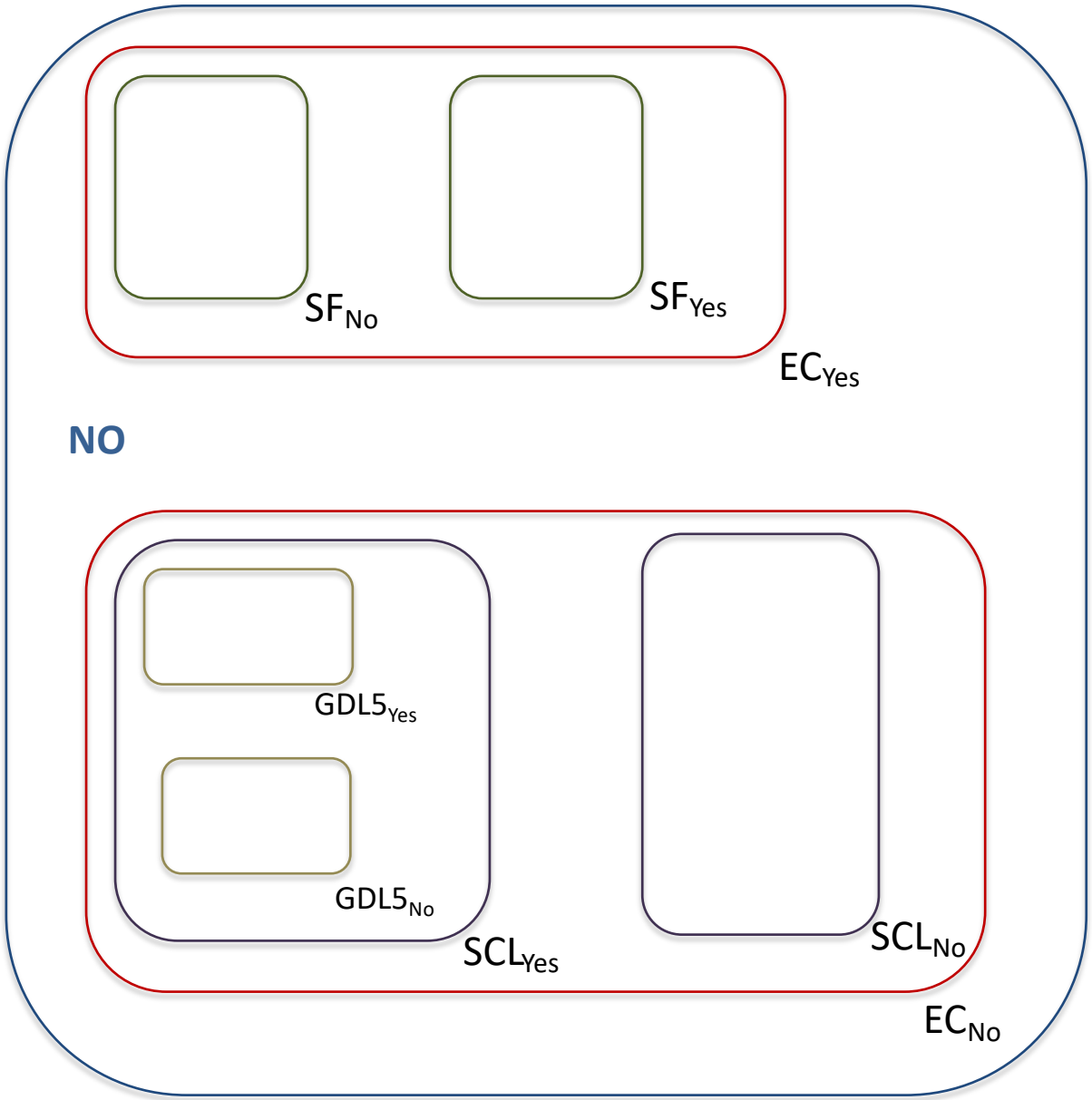
A parsing example

a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



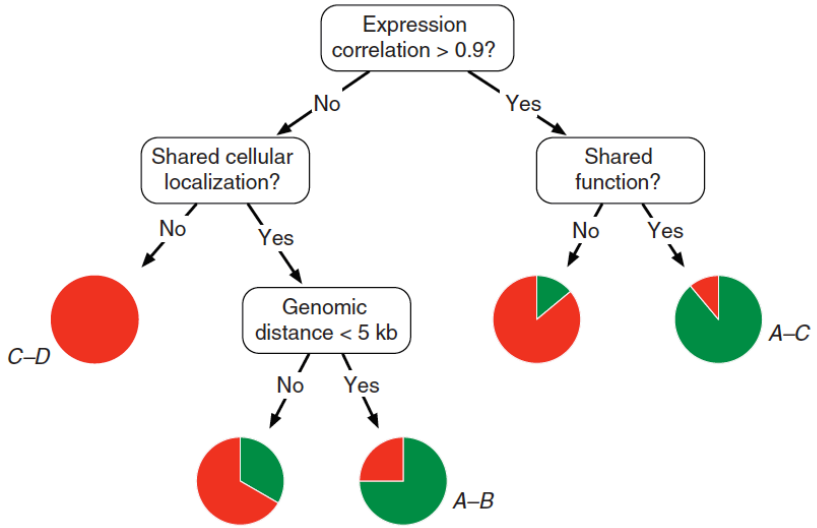


A parsing example

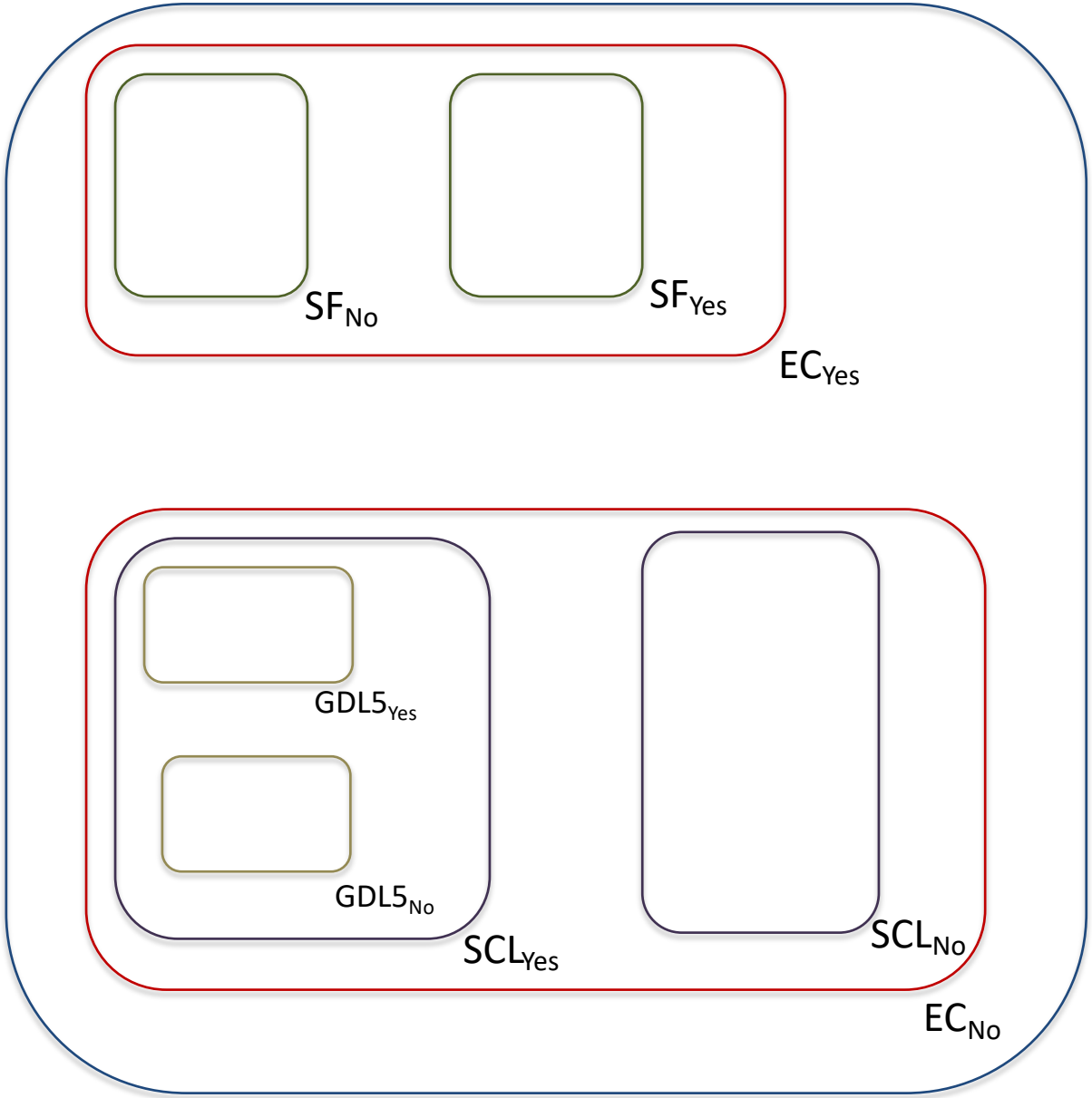
a

Gene Pair	Interact?	Expression correlation	Shared localization?	Shared function?	Genomic distance
A-B	Yes	0.77	Yes	No	1 kb
A-C	Yes	0.91	Yes	Yes	10 kb
C-D	No	0.1	No	No	1 Mb
⋮					

b



NO



Applying a machine learning technique inside a P system

Sample	X	Y	Z	Decision
1	High	High	High	Yes
2	High	High	High	Yes
3	Low	High	Low	Yes
4	Medium	High	High	Not

Applying a machine learning technique inside a P system

Sample	X	Y	Z	Decision
1	High	High	High	Yes
2	High	High	High	Yes
3	Low	High	Low	Yes
4	Medium	High	High	Not

$$X^1_{High}\ Y^1_{High}\ Z^1_{High}\ Decision^1_{YES}$$

$$X^2_{High}\ Y^2_{High}\ Z^2_{High}\ Decision^2_{YES}$$

$$X^3_{Low}\ Y^3_{High}\ Z^3_{Low}\ Decision^3_{YES}$$

$$X^4_{Medium}\ Y^4_{High}\ Z^4_{Low}\ Decision^4_{NO}$$

Applying a machine learning technique inside a P system

Sample	X	Y	Z	Decision
1	High	High	High	Yes
2	High	High	High	Yes
3	Low	High	Low	Yes
4	Medium	High	High	Not

$$X^1_{High}\ Y^1_{High}\ Z^1_{High}\ Decision^1_{YES}$$

$$X^2_{High}\ Y^2_{High}\ Z^2_{High}\ Decision^2_{YES}$$

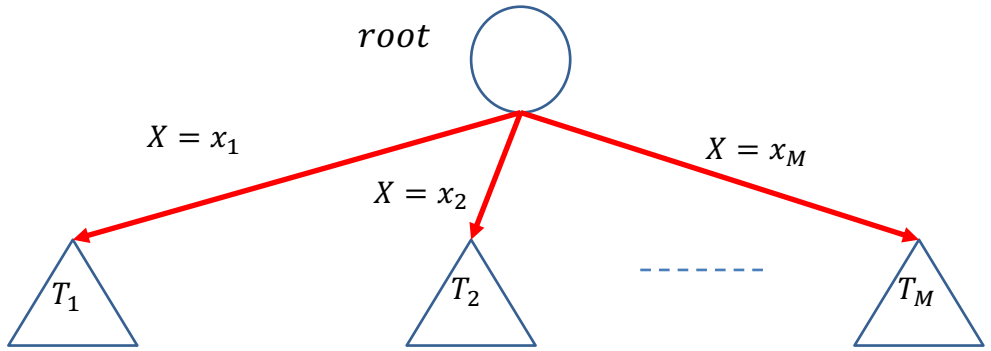
$$X^3_{Low}\ Y^3_{High}\ Z^3_{Low}\ Decision^3_{YES}$$

$$X^4_{Medium}\ Y^4_{High}\ Z^4_{Low}\ Decision^4_{NO}$$

A generic algorithm to build decision trees from data

Input: A finite set of supervised tuples E
Output: A decision tree T
Method:
1) Create an arbitrary root
2) If all the tuples belong to class C_j
   then *return*(root, C_j)
   else
   1. Select an attribute X with values $x_1, x_2, \dots, x_M$
   2. Make a partition of E according to the attribute value $E_1, E_2, \dots, E_M : E = \cup_{i=1}^{M} E_i$
   3. Build decision trees for every subset: $T_1, T_2, \dots, T_M$

endMethod



Applying a machine learning technique inside a P system

Sample	X	Y	Z	Decision
1	High	High	High	Yes
2	High	High	High	Yes
3	Low	High	Low	Yes
4	Medium	High	High	Not

$X^1_{High}\ Y^1_{High}\ Z^1_{High}\ Decision^1_{YES}$

$X^2_{High}\ Y^2_{High}\ Z^2_{High}\ Decision^2_{YES}$

$X^3_{Low}\ Y^3_{High}\ Z^3_{Low}\ Decision^3_{YES}$

$X^4_{Medium}\ Y^4_{High}\ Z^4_{Low}\ Decision^4_{NO}$

A generic algorithm to build decision trees from data

Input: A finite set of supervised tuples E
Output: A decision tree T
Method:
1) Create an arbitrary root
2) If all the tuples belong to class C_j
   then *return*(root, C_j)
   else
   1. Select an attribute X with values $x_1, x_2, \dots, x_M$
   2. Make a partition of E according to the attribute value $E_1, E_2, \dots, E_M : E = \cup_{i=1}^{M} E_i$
   3. Build decision trees for every subset: $T_1, T_2, \dots, T_M$

endMethod

Suppose that the attribute for the node root is X, then we apply the following three rules (membrane creation)

$$X^1_{High}\ Y^1_{High}\ Z^1_{High}\ Decision^1_{YES}\ X^2_{High}\ Y^2_{High}\ Z^2_{High}\ Decision^2_{YES} \rightarrow [\ Y^1_{High}\ Z^1_{High}\ Decision^1_{YES} Y^2_{High}\ Z^2_{High}\ Decision^2_{YES}]_{X_{High}}$$

$$X^3_{Low}\ Y^3_{High}\ Z^3_{Low}\ Decision^3_{YES} \rightarrow [\ Y^3_{High}\ Z^3_{Low}\ Decision^3_{YES}]_{X_{Low}}$$

$$X^4_{Medium}\ Y^4_{High}\ Z^4_{Low}\ Decision^4_{NO} \rightarrow [\ Y^4_{High}\ Z^4_{Low}\ Decision^4_{NO}]_{X_{Medium}}$$

Applying a machine learning technique inside a P system

How can the appropriate rules be selected ?

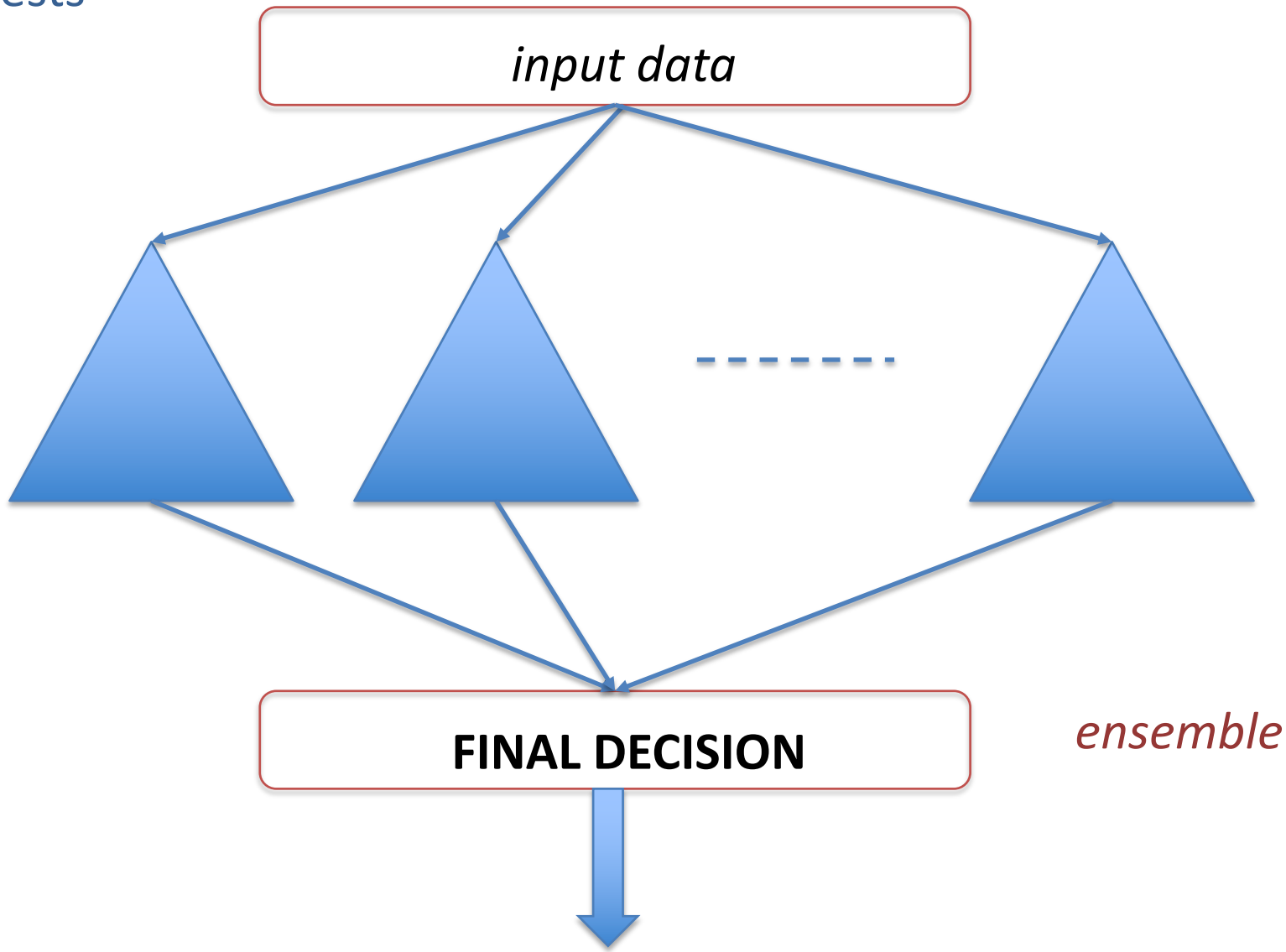
- External algorithm selects the rule and we apply a translation scheme from decision trees to P systems
- Apply a P system working within an entropic manner (CMC17)
- Apply rules according to a functional criterium: (FUNCTIONAL WORKING MANNER !!!)

$f: M(V) \rightarrow R$ — A function defines the selections of the rules (instead of selection by priority)

$(u_1, \ldots, u_q): u \rightarrow v$ — Every rule has some parameters (elements from V) and the function f is calculated over them.

The rule to be applied has the highest value for the function f.

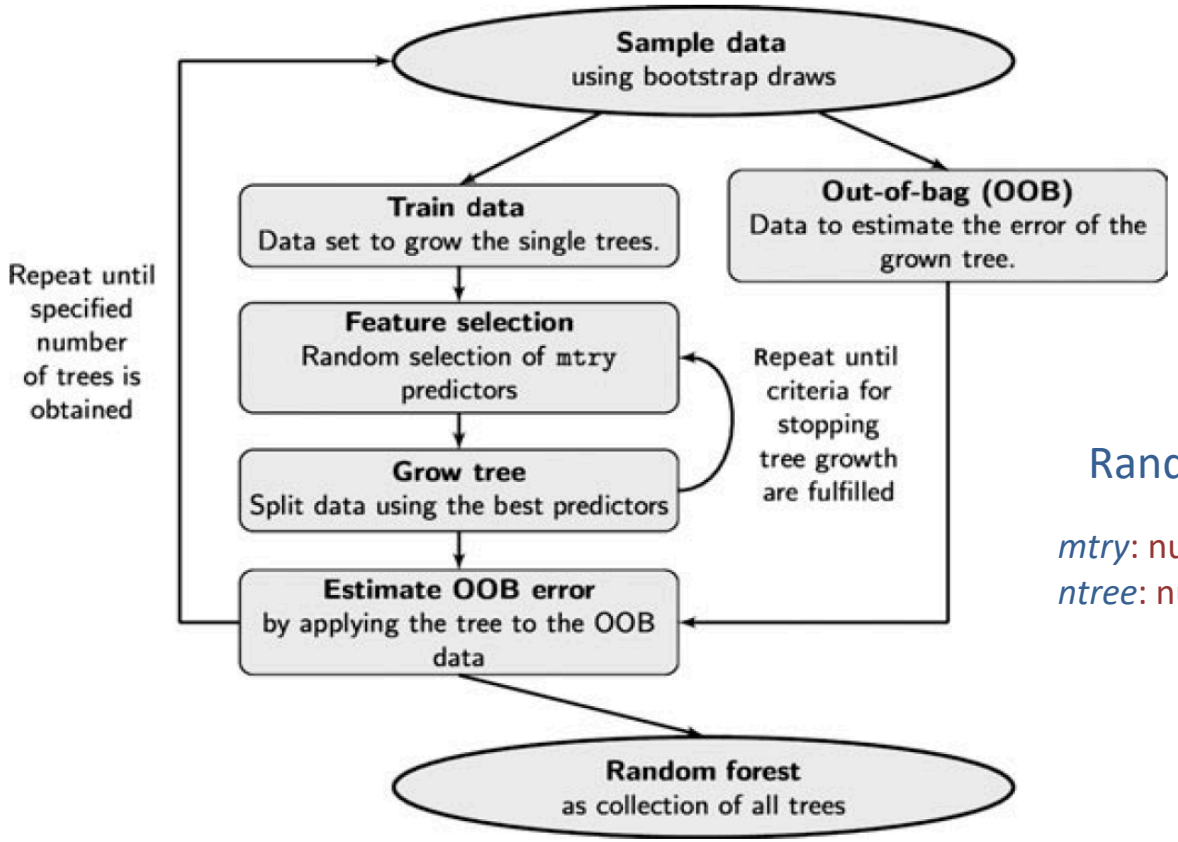
Random forests



decision trees

ensemble decision criteria

From decision trees to random forests



Random Forest algorithm

mtry: number of features for node splitting
ntree: number of trees in the forest

From cell-like P systems to tissue-like P systems

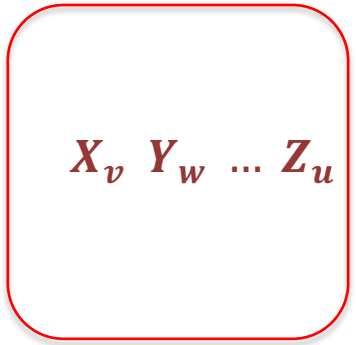
$X_v\ Y_w\ \dots\ Z_u$

Initially we have only one cell with cell-creation rules and all the input data

$$X_v\ Y_w\ \dots\ Z_u \rightarrow X_v\ Y_w\ \dots\ Z_u(X_v\ Y_w)$$

$$X_v\ Y_w\ \dots\ Z_u \rightarrow X_v\ Y_w\ \dots\ Z_u(Y_w\ Z_u)$$

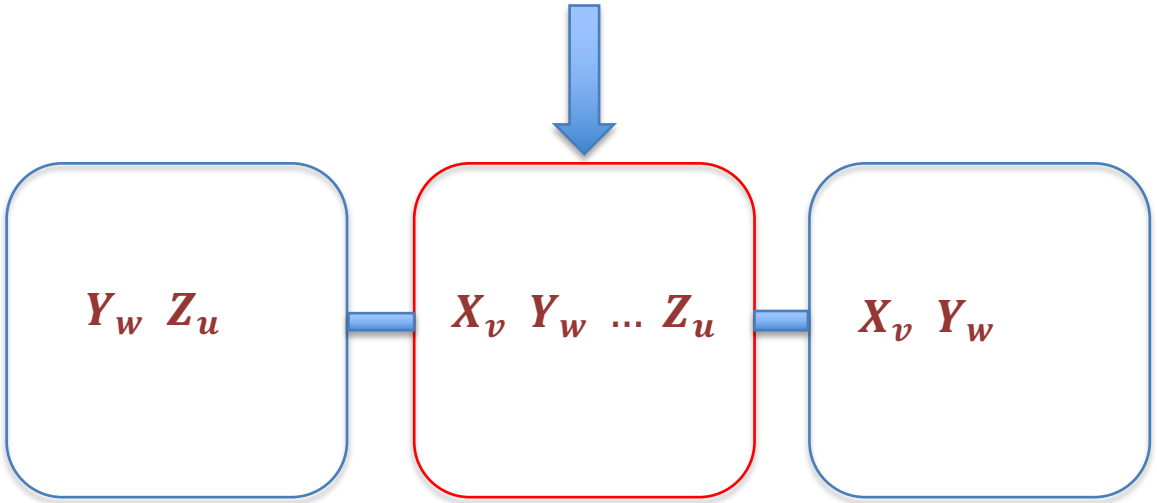
From cell-like P systems to tissue-like P systems



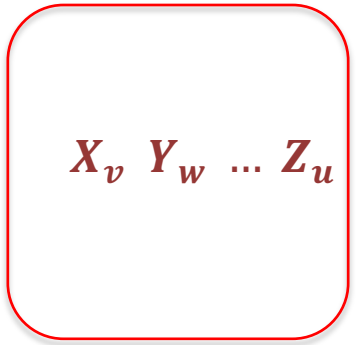
Initially we have only one cell with cell-creation rules and all the input data

$$X_v\ Y_w\ \ldots\ Z_u \rightarrow X_v\ Y_w\ \ldots\ Z_u(X_v\ Y_w)$$

$$X_v\ Y_w\ \ldots\ Z_u \rightarrow X_v\ Y_w\ \ldots\ Z_u(Y_w\ Z_u)$$



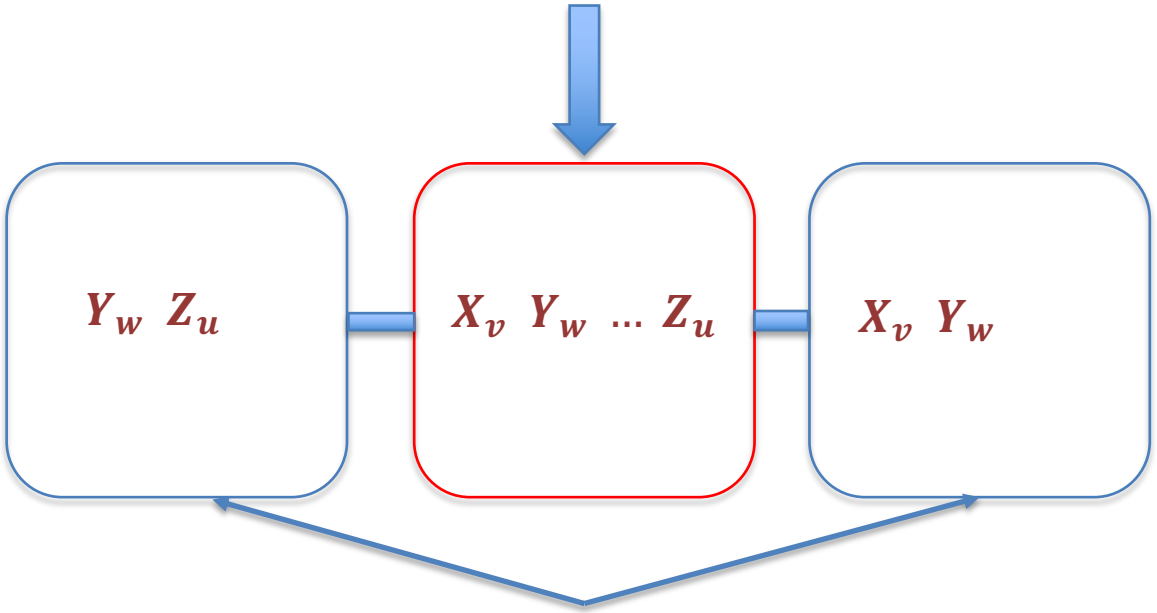
From cell-like P systems to tissue-like P systems



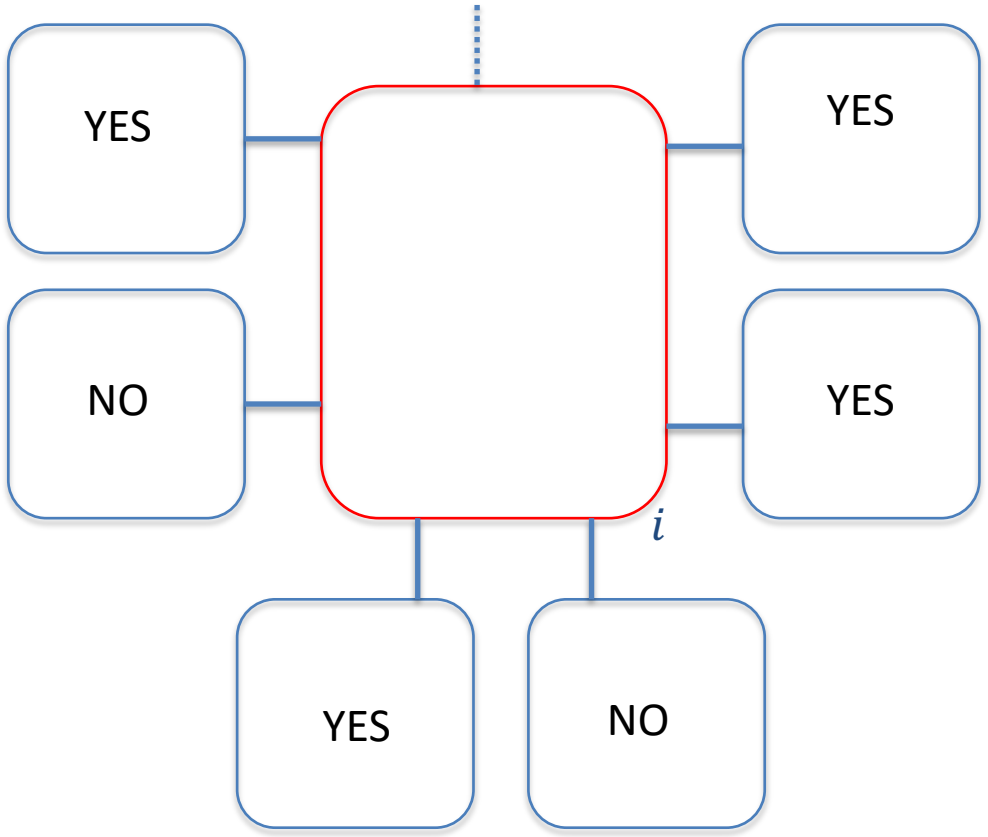
Initially we have only one cell with cell-creation rules and all the input data

$$X_v\ Y_w\ \dots\ Z_u \rightarrow X_v\ Y_w\ \dots\ Z_u(X_v\ Y_w)$$

$$X_v\ Y_w\ \dots\ Z_u \rightarrow X_v\ Y_w\ \dots\ Z_u(Y_w\ Z_u)$$



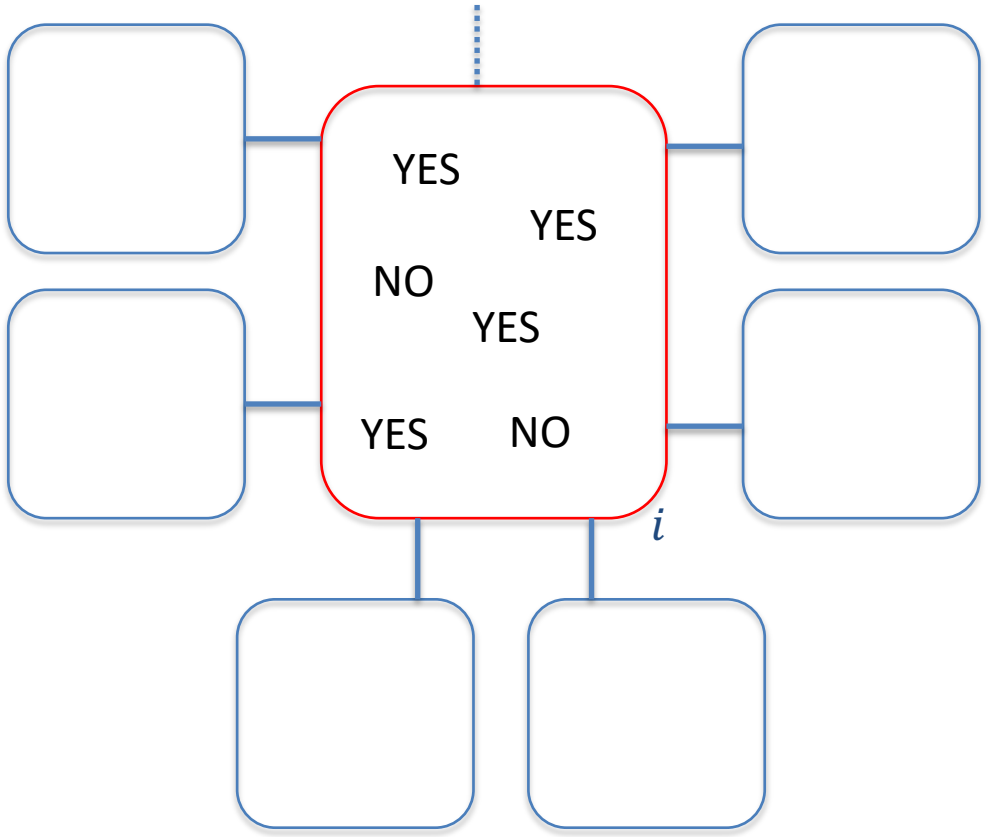
Apply rules to create a decision tree

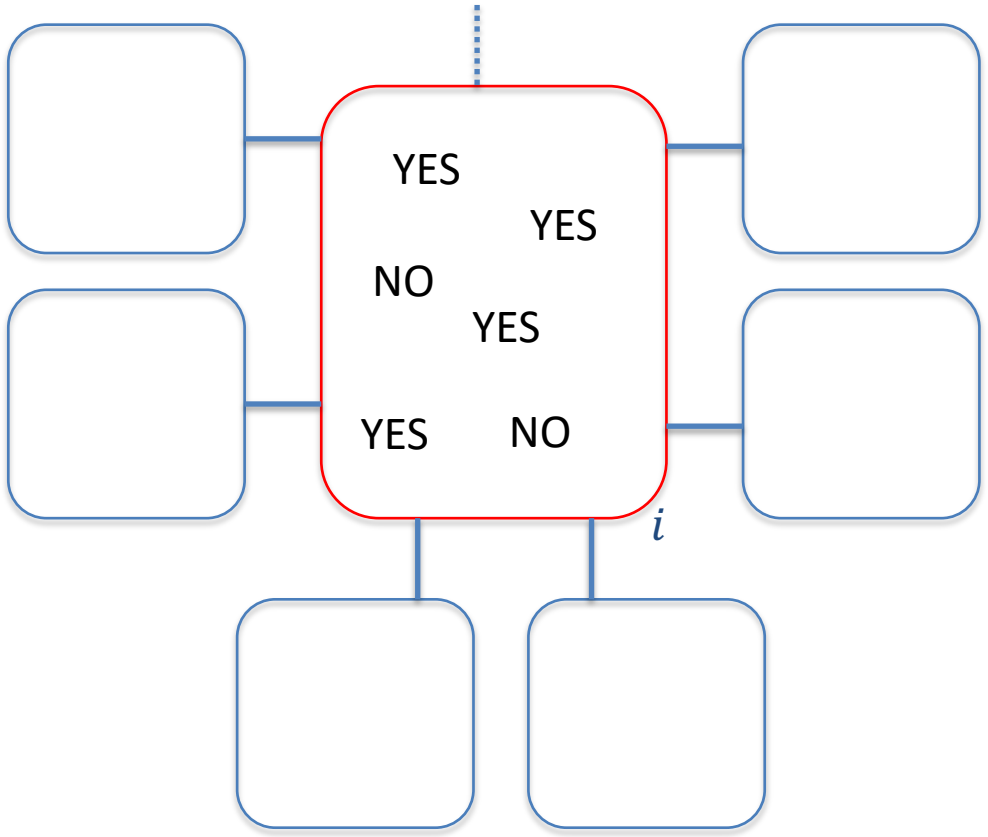


Rules inside every P system at skin region (communication rules)

YES $\rightarrow (\boldsymbol{YES})_i$

NO $\rightarrow (\boldsymbol{NO})_i$

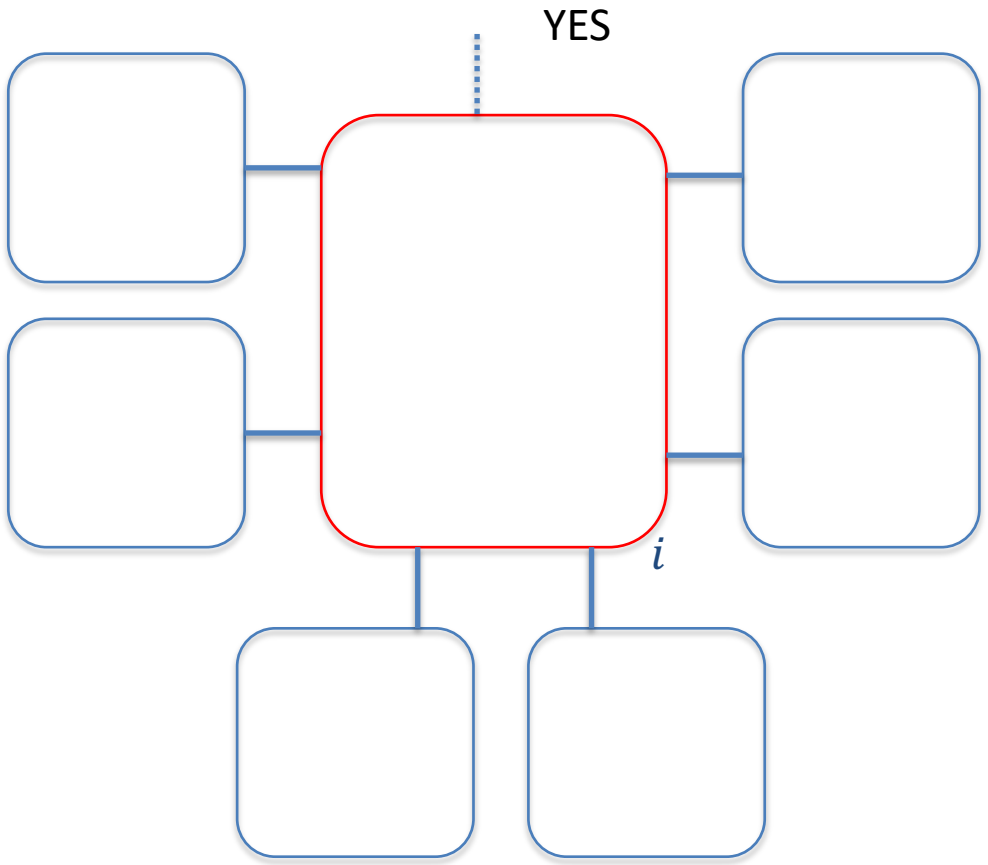

YES
YES
NO
YES
YES
NO
i



Majority rules

$$\#YES > \#NO \rightarrow YES_{out}$$

$$\#NO > \#YES \rightarrow NO_{out}$$



Majority rules

$$\#YES > \#NO \rightarrow YES_{out}$$

$$\#NO > \#YES \rightarrow NO_{out}$$

Conclusions (... work in progress)

- A definition of decision trees by P systems
- Introducing machine learning approach by P systems working in an entropic/functional manner
- A definition of random forests by (tissue) P systems
- Introducing ensemble decision criteria by (majority) rules

Thank you for your attention !!!