Shared Ensembles using Multi-trees

Vicent Estruch, Cèsar Ferri,
José Hernández-Orallo, M.José Ramírez-Quintana
{vestruch, cferri, jorallo, mramirez}@dsic.upv.es
Dep. de Sistemes Informàtics i Computació,
Universitat Politècnica de València,
Valencia, Spain

VII Conferencia Iberoamericana de Inteligencia Artificial
Sevilla, 12-15 November 2002

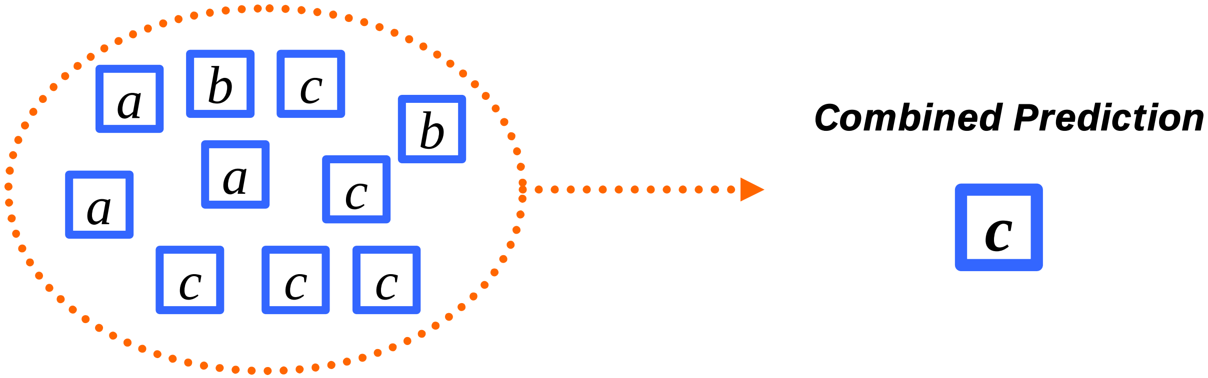
Introduction

- Machine Learning techniques that construct a model/hypothesis (e.g. ANN, DT, SVM, ...):
  - usually devoted to obtain **one** single model:
    - As accurate as possible (close to the “target” model).
    - Other (presumably less accurate) models are discarded.
  - An old alternative has recently been popularised:
    - “Every consistent hypothesis should be taken into account”

But... How?

Ensemble Methods (1/3)

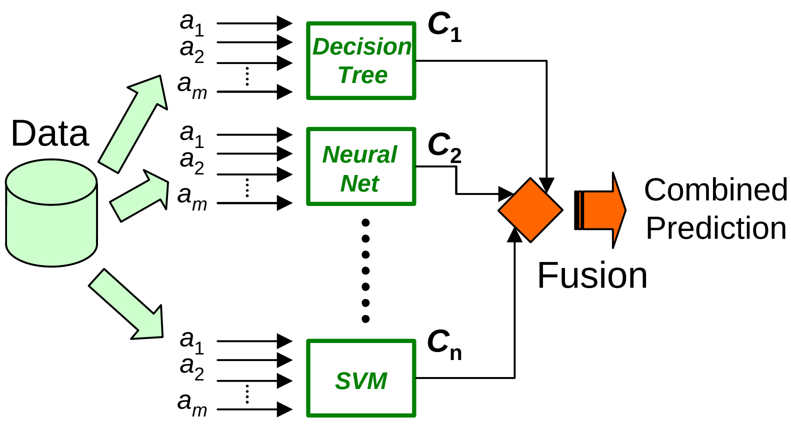
- Ensemble Methods (Multi-classifiers):
  - Generate multiple (and possibly) heterogeneous models and then combine them through *voting* or other fusion methods.



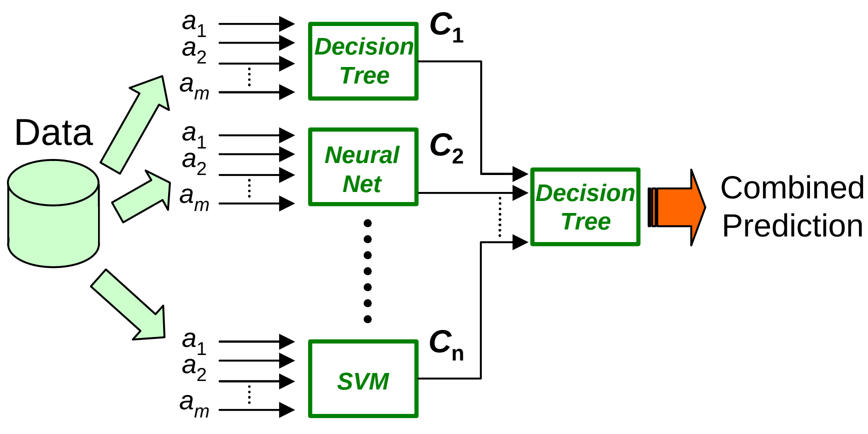
  - Much better results (in terms of accuracy) than single models when the number and variety of classifiers is high.

Ensemble Methods (2/3)

- Ensemble Methods (Multi-classifiers):
  - Different topologies: simple, stacking, cascading, ...



Simple Combination



Stacking

  - Different generation policies: *boosting, bagging, randomisation, ...*
  - Different fusion methods: majority voting, average, maximum, ...

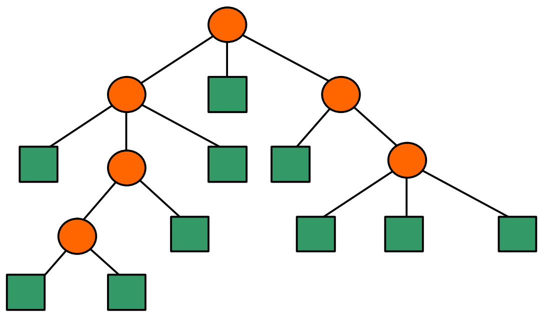
Ensemble Methods (3/3)

- Main drawbacks:
  - **Computational costs**: huge amounts of memory and time are required to obtain and store the set of hypotheses (ensemble).
  - **Throughput**: the application of the combined model is slow.

The solution of these drawbacks would boost the applicability of ensemble methods in machine learning applications.

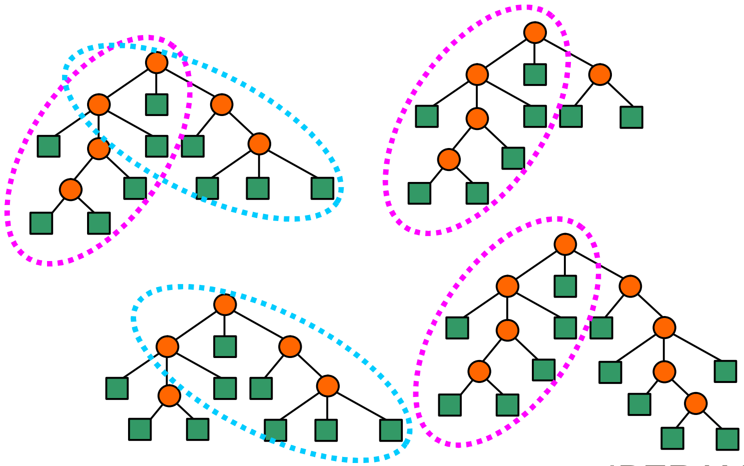
Ensembles of Decision Trees

- Decision Tree:



  - Each internal node represents a condition.
  - Each leaf assigns a class to the examples that fall under that leaf.

- Forest: several decision trees can be constructed.



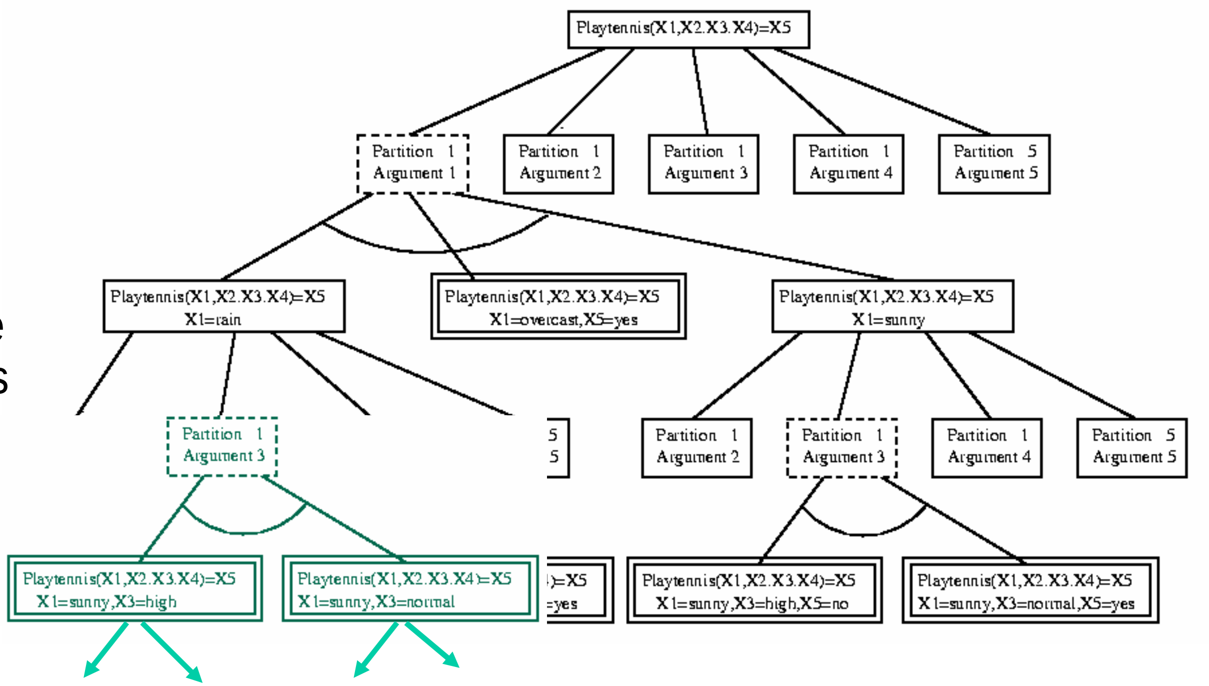
  - Many trees have common parts.
  - Traditional ensemble methods repeat those parts:
    - memory and time ↑↑↑.
    - comprehensibility is lost.

Decision Tree *Shared* Ensembles

- Shared ensemble:
  - Common parts are shared in an AND/OR tree structure.

- Construction space and time resources are highly reduced

- Throughput is also improved by this technique.



Decision Tree *Shared* Ensembles

- Previous work:
  - Multiple Decision Trees (Kwok & Carter 1990)
  - Option Decision Trees (Buntine 1992)
    - The AND/OR tree structure is populated (partially) breadth-first.
  - Combination has been performed:
    - Using weighted combination (Buntine 1992).
    - Using majority voting combination (Kohavi & Kunz 1997).
  - Different conclusions on where alternatives are especially beneficial:
    - At the bottom of the tree (Buntine).
      - Trees are quite similar → Accuracy improvement is low.
    - At the top of the tree (Kohavi & Kunz).
      - Trees share few parts → Space resources are exhausted as in other non-shared ensembles (boosting, bagging, ...).

Decision Tree *Shared* Ensembles

- Previous work:
  - Drawbacks of Option Decision Trees:
    - The number of alternative options is very difficult to be determined during the construction stage → size of the AND/OR structure is mostly unpredictable.
    - The fusion strategy (weighted, majority) determines the policy and number of alternative trees to be explored.
    - An “option factor” is required. The appropriate value highly depends on each particular dataset.
      - For option factor values such as 0.4, some datasets suffer an exponential increase of the number of nodes.
      - “*Soybean was the extreme case, which increased from 68 nodes to 203,577 nodes*” (Kohavi & Kunz 1997).

Multi-tree Construction

- New Way of Populating the AND/OR Tree:
  - The first tree is constructed in the classical eager way.
  - Discarded alternative splits are stored in a list.
    - Repeat *n* times:
      - Once a tree is finished, the best alternative split (according to a “wakening” criterion) is chosen.
      - The branch is finished using the classical eager way.
  - This amounts to a ‘beam’ search → Anytime algorithm.
    - Extensions and alternatives can happen at any part of the tree (top, bottom).
    - The populating strategy can be easily changed.
    - The fusion strategy can also be flexibly modified.
    - The desired size of the AND/OR tree can be specified quite precisely.

Fusion Methods

- Combination on the Multi-tree:
  - The number of trees grows exponentially wrt. the number of alternative OR-nodes explored:
    - Advantages: ensembles are now much bigger with a constant increase of resources. Presumably, the combination will be more accurate.
    - Disadvantages: the combination at the top is unfeasible.
  - Global fusion techniques would be prohibitive.

Local Fusion

- First Stage. Classical top-down:
  - Each example to be predicted is distributed top-down into many alternative leaves.
  - The example is labelled in each leaf (*class vector*).
- Second Stage. The fusion goes bottom-up:
  - Whenever an OR-node is found. The (possibly) inconsistent predictions are combined through a *local fusion method*:

- Fusion of millions or billions of trees can be performed efficiently.

Local Fusion Methods

- Class vector transformation:
  - Good loser, bad loser, majority, difference, ...
- Fusion strategy
  - Sum, arithmean, product, geomean, max, min, ...
- When the *fusioned* vector reaches the top, the class with the greatest value is chosen.

- Examples:

Original	Good loser	Bad loser	Majority	Difference
{ 40, 10, 30 }	{ 80, 0, 0 }	{ 40, 0, 0}	{ 1, 0, 0 }	{ 0, -60, -20 }
{ 7, 2, 10 }	{ 0, 0, 19 }	{ 0, 0, 10 }	{ 0, 0, 1 }	{ -5, -15, 1 }

MIN:	**{ 7, 2, 10 }**	**{ 0, 0, 0 }**	**{ 0, 0, 0 }**	**{ 0, 0, 0 }**	**{ -5, -60, 1 }**
	c	***a b c***	***a b c***	***a b c***	***c***

Experiments (1/4)

- Experimental setting:
  - 15 datasets from the UCI repository.
  - Multi-tree implemented in the **SMILES** system.
  - Splitting criterion: GainRatio (C4.5).
  - Second node selection criterion (wakening criterion): random.
  - Boosting and Bagging from WEKA.

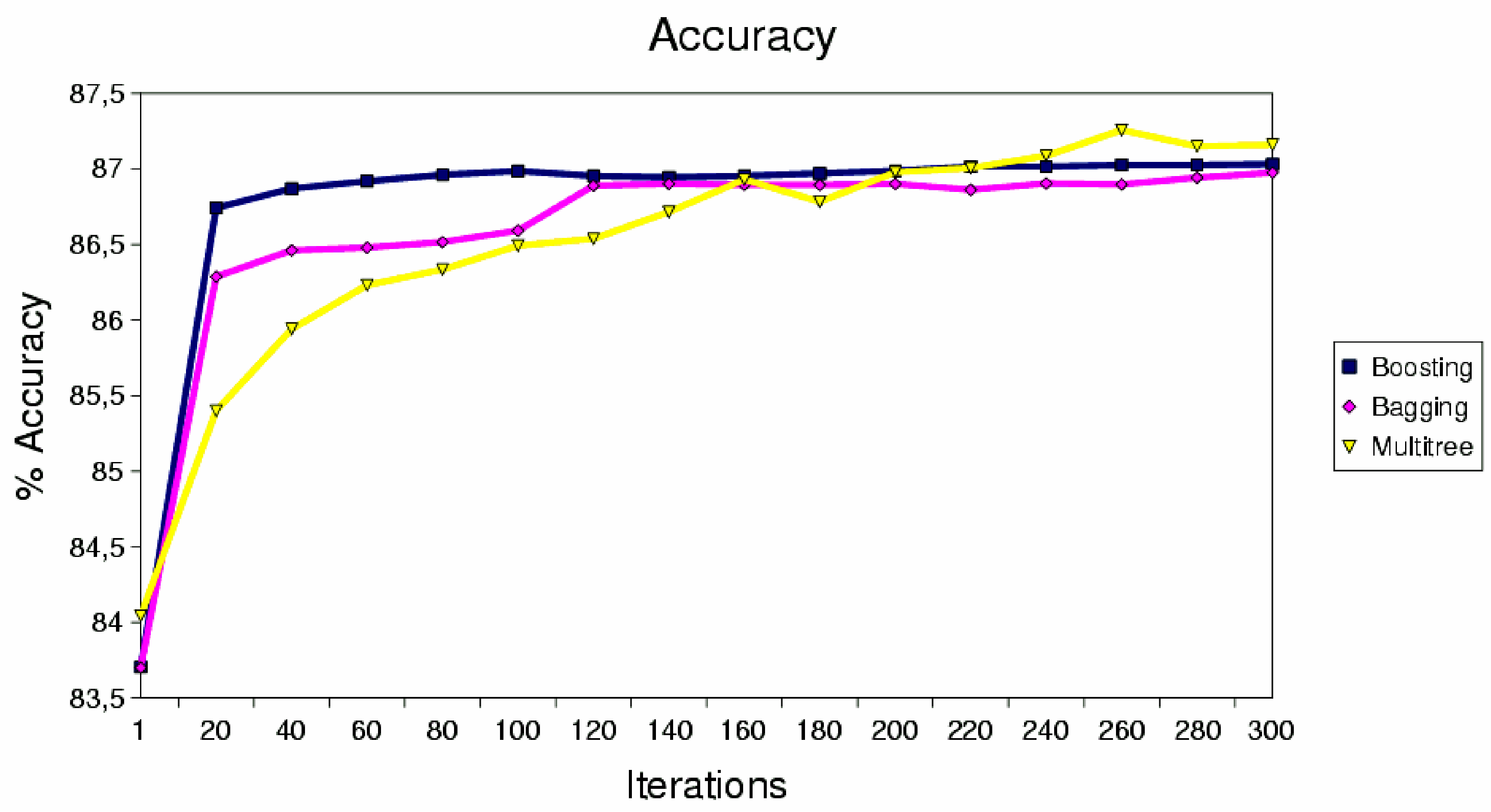
Experiments (2/4)

- Comparison between fusion techniques

	Arit.		Sum.		Prod.		Max.		Min.	
#	Acc.	Dev.	Acc.	Dev.	Acc.	Dev.	Acc.	Dev.	Acc.	Dev.
1	80.69	5.01	81.24	4.66	76.61	5.04	83.02	4.76	76.61	5.04
2	91.22	2.25	91.25	2.26	83.38	3.65	90.90	2.09	83.38	3.65
3	94.17	4.06	94.34	3.87	89.06	5.19	94.00	4.05	89.06	5.19
4	80.09	6.26	79.91	6.13	76.97	7.14	80.09	6.11	76.97	7.14
5	95.63	3.19	95.77	3.18	93.28	3.71	95.93	2.81	93.28	3.71
6	94.53	5.39	94.20	5.66	94.00	5.34	94.47	5.45	94.40	5.34
7	99.67	1.30	99.71	1.18	81.00	8.60	99.89	0.51	81.00	8.60
8	73.35	5.86	73.73	5.82	74.53	5.25	77.15	5.88	74.53	5.25
9	97.87	2.00	97.91	1.80	97.58	2.45	97.62	1.93	97.58	2.45
10	94.52	4.25	93.76	5.10	92.05	5.71	92.57	5.43	92.05	5.71
11	62.50	16.76	63.25	16.93	61.63	17.61	67.13	14.61	61.63	17.61
12	97.50	8.33	97.50	9.06	97.75	8.02	94.75	11.94	97.75	8.02
13	63.60	12.59	64.33	11.74	62.00	12.26	63.93	12.03	62.00	12.26
14	81.73	3.82	82.04	3.78	78.93	3.73	82.68	3.97	78.93	3.73
15	94.06	6.00	93.88	6.42	91.47	7.11	92.53	6.99	91.47	7.11
Geomean	85.83	4.72	85.99	4.71	82.53	5.93	86.40	4.52	82.55	5.93

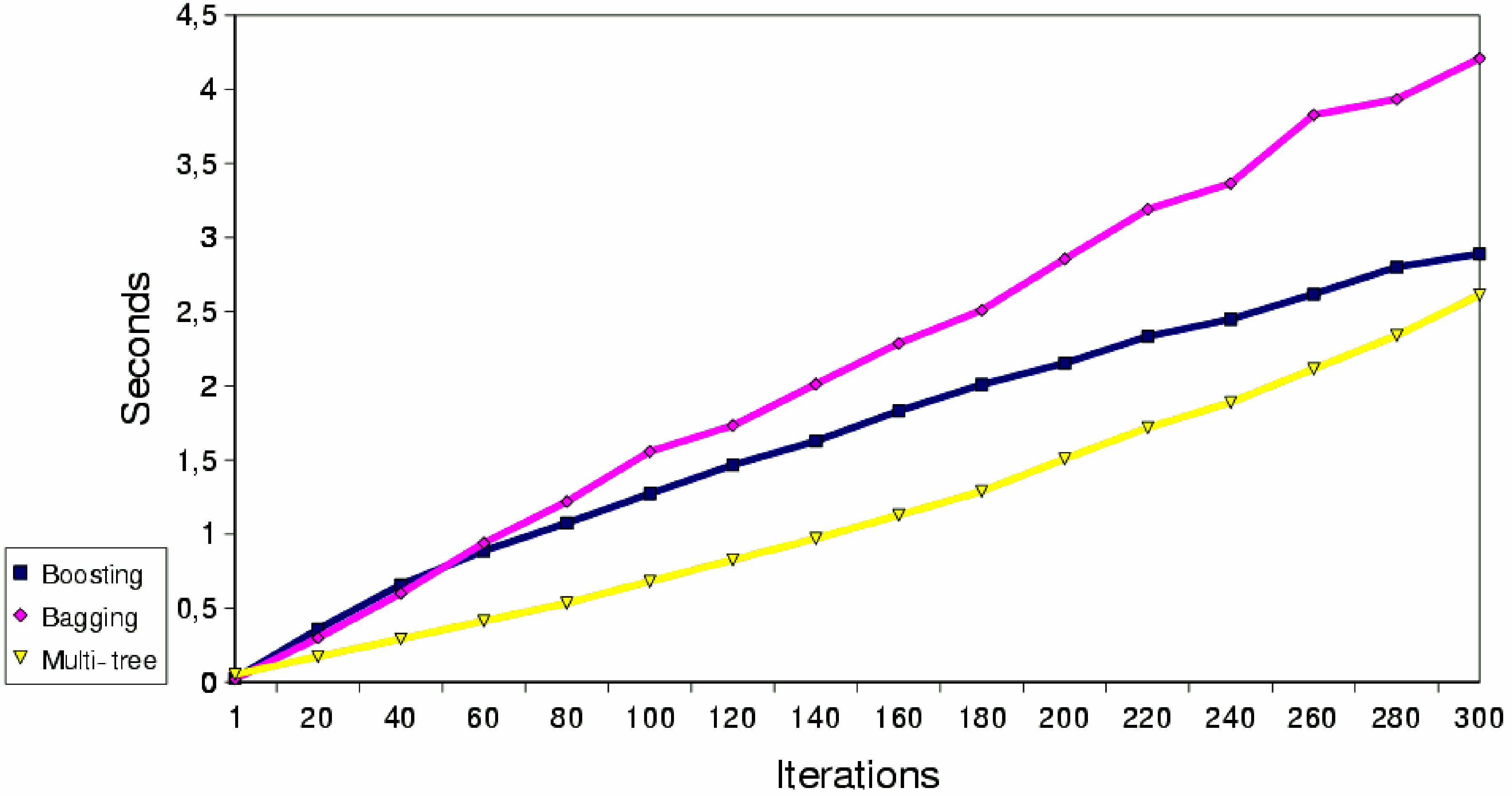
Experiments (3/4)

- Combination Accuracy compared to other Ensemble Methods:



Experiments (4/4)

- Combination Resources compared to other Ensemble Methods:



Conclusions

- Multi-tree as an alternative to other population strategies for shared decision tree ensembles:
  - Anytime character
    - The first tree is obtained in the same way as classical eager decision tree learning.
    - We ask for further solutions on demand.
  - Population (and hence resources) is scalable and easy to be controlled.
  - Fusion strategies are flexible.
    - *Maximum* fusion strategy seems to be the best one.
- Same or even better accuracy results than other ensemble methods with significantly lower resource consumption.

Conclusions

- Some further improvements:
  - *Forgetting*: not all the alternative OR-nodes are stored. Memory and time requirements are reduced even further with the same accuracy results.
  - Other uses of the multi-tree structure: extraction of the "best" single tree (Occam, archetype, ...).
- **SMILES** is freely available at:
  - **`http://www.dsic.upv.es/~flip/smiles/`**