On the Automatic Design of Decision-Tree Induction Algorithms

Rodrigo C. Barros [1]
André C. P. L. F. de Carvalho[1] **(supervisor)**
Alex A. Freitas[2] **(co-supervisor)**

[1]Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo (ICMC–USP)
São Carlos– SP – Brazil

[2]School of Computing – University of Kent
Canterbury, U.K.

{rcbarros,andre}@icmc.usp.br, A.A.Freitas@kent.ac.uk

1. Motivation

A decision tree is a classifier represented by a flowchart-like tree structure that has been widely used to represent classification models, specially due to its comprehensible nature that resembles the human reasoning. In a recent poll from the *kdnuggets* website [1], decision trees figured as the most used data mining/analytic method by researchers and practitioners, reaffirming its importance in machine learning tasks. Decision-tree induction algorithms present several advantages over other learning algorithms, such as robustness to noise, low computational cost for generating the model, and ability to deal with redundant attributes [Rokach and Maimon 2005].

Several attempts on optimising decision-tree algorithms have been made by researchers within the last decades, even though the most successful algorithms date back to the mid-80's and early 90's. Many strategies were employed for deriving accurate decision trees, such as bottom-up induction [Barros et al. 2011], linear programming [Bennett and Mangasarian 1994], hybrid induction [Kim and Landgrebe 1991], ensemble of trees [Breiman 2001], and evolutionary induction [Barros et al. 2012a], just to name a few. Regardless of the strategy we choose to induce decision trees, we are susceptible to the method's inductive bias. Since we know that certain inductive biases are more suitable to certain problems, and that no method is best for every single problem (*i.e.*, the no free lunch theorem [Wolpert and Macready 1997]), there is a growing interest in developing automatic methods for deciding which learner to use in each situation. A whole new research area named *meta-learning* has emerged for solving this problem [Smith-Miles 2009]. Meta-learning is an attempt to understand data *a priori* of executing a learning algorithm. In a particular branch of meta-learning, *algorithm recommendation*, data that describe the characteristics of data sets and learning algorithms (*i.e.*, meta-data) are collected, and a learning algorithm is employed to interpret these meta-data and suggest a particular learner (or ranking a few learners) in order to better solve the problem at hand. Meta-learning has a few limitations, though. For instance, it provides a limited number of algorithms to be selected from a list. In addition, it is not an easy task to define the set of meta-data that will hopefully contain useful information for identifying the best algorithm to be employed.

[1]http://www.kdnuggets.com/polls/2007/data_mining_methods.htm

A recent research area within the combinatorial optimisation field named "hyper-heuristics" (HHs) has emerged with a similar goal to meta-learning: searching in the heuristics space, or in other words, *heuristics to choose heuristics* [Cowling et al. 2001]. HHs are related to metaheuristics, though with the difference that they operate on a search space of heuristics whereas metaheuristics operate on a search space of solutions to a given problem. Nevertheless, HHs usually employ metaheuristics (*e.g.*, evolutionary algorithms) as the search methodology to look for suitable heuristics to a given problem [Pappa et al. 2013]. Considering that an algorithm or its components can be seen as heuristics, one may say that HHs are also suitable tools to automatically design custom (tailor-made) algorithms. In this case, there is a set of human designed components or heuristics, surveyed from the literature, which are chosen to be the starting point for the evolutionary process. The expected result is the automatic generation of new procedural components and heuristics during evolution, depending of course on which components are provided to the EA and the respective "freedom" it has for evolving the solutions.

During my PhD, I developed HEAD-DT – a Hyper-heuristic Evolutionary algorithm for Automatically Designing Decision-Tree induction algorithms, which is the main topic of the defended thesis. We believe HEAD-DT is the solution to properly select the (near-)optimal bias in decision-tree approaches. In addition, to the best of our knowledge, we were the first researchers to develop a hyper-heuristic to automatically design decision-tree induction algorithms. Besides the originality of the theme, we should point out that decision-tree induction algorithms are widely used in a variety of application domains, and that our approach enables the creation of tailor-made algorithms in virtually no time (when compared to the manual approach of designing those algorithms). The amount of different domains that may benefit from HEAD-DT is quite large – for instance, we published papers in which we applied HEAD-DT in areas as diverse as bioinformatics (rational drug design [Barros et al. 2012b] and microarray gene expression classification [Barros et al. 2014]) and software maintenance effort estimation [Basgalupp et al. 2013].

2. Thesis Contributions

The main contribution of this thesis is a new algorithm for automatically designing decision-tree induction algorithms (HEAD-DT). As specific contributions of the thesis regarding the automatic design of decision-tree induction algorithms, we can cite:

- **The Specific Framework – evolution of a decision-tree induction algorithm tailored to one specific data set at a time**. We show that decision-tree algorithms that are designed to excel at a single data set usually outperform traditional decision-tree algorithms such as C4.5 and CART.
- **The General Framework – evolution of a decision-tree induction algorithm from multiple data sets**. We show that decision-tree induction algorithms may be designed to excel at a particular group of data sets, though with distinct objectives.

Specifically regarding the general framework, the thesis presents the following specific contributions:

- **Evolution of a single decision-tree induction algorithm for data sets from a particular application domain**. We performed a detailed empirical analysis on microarray gene expression data, and we show that automatically-designed decision-tree induction algorithms tailored to a particular domain usually outperform traditional decision-tree algorithms such as C4.5 and CART.

- **Evolution of a single decision-tree induction algorithm for a variety of data sets**. We performed a thorough empirical analysis on almost 70 UCI data sets, and we show that automatically-designed decision-tree induction algorithms, which are meant to be robust across very different data sets, show a performance similar to traditional decision-tree algorithms such as C4.5 and CART.
- **Evolution of a single decision-tree induction algorithm for data sets with a particular statistical profile**. After performing an extensive analysis on distinct fitness functions for HEAD-DT, we show that automatically-designed decision-tree induction algorithms tailored to balanced (and imbalanced) data sets usually outperform traditional decision-tree algorithms such as C4.5 and CART in these data sets.

Finally, we can include as specific contributions of the defended thesis a detailed discussion on the cost-effectiveness of automated algorithm design in contrast to the manual algorithm design, as well as an empirical demonstration confirming that the genetic search is significantly more effective than a random search in the space of decision-tree induction algorithms.

3. Publications

In this section, we present the list of published papers that fall within the scope of the thesis, organised by themes:

[Decision-tree induction algorithms] — the following papers refer to decision-tree induction algorithms I developed during my PhD:

Conference papers:

- BARROS, R. C.; CERRI, R.; JASKOWIAK, P. A.; DE CARVALHO, A. C. P. L. F. A Bottom-Up Oblique Decision Tree Induction Algorithm. *: 11th International Conference on Intelligent Systems Design and Applications*, 450–456, 2011. Qualis B2.
- BARROS, R. C.; DE CARVALHO, A. C. P. L. F.; BASGALUPP, M. P.; QUILES, M. G. A Clustering-based Decision Tree Induction Algorithm. *: 11th International Conference on Intelligent Systems Design and Applications*, 543–550, 2011. Qualis B2.
- BARROS, R. C.; DE CARVALHO, A. C. P. L. F.; BASGALUPP, M. P.; QUILES, M. G. Um Algoritmo de Indução de Árvores de Decisão baseado em Agrupamento. *: VIII Encontro Nacional de Inteligência Artificial*, 2011. Qualis B4.

Journal papers:

- BARROS, R. C.; JASKOWIAK, P. A.; CERRI, R.; DE CARVALHO, A. C. P. L. F. A Framework for Bottom-Up Induction of Decision Trees. *Neurocomputing*, in press, 2014. Qualis A1.
- BARROS, R. C.; DE CARVALHO, A. C. P. L. F.; BASGALUPP, M. P.; QUILES, M. G. Clus-DTI: Improving Decision-Tree Classification with a Clustering-based Decision-Tree Induction Algorithm. *Journal of the Brazilian Computer Society*, 18:4, 351–362, 2012. Qualis B2.

Book chapters:

- BASGALUPP, M. P.; BARROS, R. C.; DE CARVALHO, A. C. P. L. F.; FREITAS, A. A. A Beam-Search based Decision-Tree Induction Algorithm. *: Machine Learning Algorithms for Problem Solving in Computational Applications: Intelligent Techniques*, IGI-Global, 2011.

[Evolutionary algorithms to evolve decision trees] — the following papers refer to evolutionary algorithms I developed (or co-developed) to evolve decision trees (either for classification or for regression):

Conference papers:

- BASGALUPP, M. P.; BARROS, R. C.; RUIZ, D. D. Predicting Software Maintenance Effort through Evolutionary based Decision Trees. *:27th Annual ACM Symposium on Applied Computing*, 1209-1214, 2012. Qualis A1.

Journal papers:

- BARROS, R. C.; RUIZ, D. D.; BASGALUPP, M. P. Evolutionary Model Trees for Handling Continuous Classes in Machine Learning. *Information Sciences*, 181, 954–971, 2011. Qualis A1.
- BARROS, R. C.; BASGALUPP, M. P.; DE CARVALHO, A. C. P. L. F.; FREITAS, A. A. A Survey of Evolutionary Algorithms for Decision-Tree Induction. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 42:3, 291–312, 2012. Qualis A2.
- BASGALUPP, M. P.; BARROS, R. C.; DE CARVALHO, A. C. P. L. F.; FREITAS, A. A. Evolving Decision Trees with Beam Search-based Initialization and Lexicographic Multi-Objective Evaluation. *Information Sciences*, in press, 2013. Qualis A1.

[Automatic design of decision-tree induction algorithms] — the following papers refer to the main contribution of the thesis, the automatic design of decision-tree induction algorithms:

Conference papers:

- BARROS, R. C.; BASGALUPP, M. P.; DE CARVALHO, A. C. P. L. F.; FREITAS, A. A. Towards the Automatic Design of Decision-Tree Induction Algorithms. *: 13th Annual Conference Companion on Genetic and Evolutionary Computation (GECCO 2011)*, 567–574, 2011. Qualis A1.
- BARROS, R. C.; BASGALUPP, M. P.; DE CARVALHO, A. C. P. L. F.; FREITAS, A. A. A Hyper-Heuristic Evolutionary Algorithm for Automatically Designing Decision-Tree Algorithms. *: 14th Annual Genetic and Evolutionary Computation Conference (GECCO 2012)*, 1237–1244, 2012. Qualis A1.
- BASGALUPP, M. P.; BARROS, R. C.; DA SILVA, T. S.; DE CARVALHO, A. C. P. L. F. Software Effort Prediction: A Hyper-Heuristic Decision-Tree based Approach. *: 28th Annual ACM Symposium on Applied Computing*, 1109–1116, 2013. Qualis A1.

Journal papers:

- BARROS, R. C.; WINCK, A. T.; MACHADO, K. S.; BASGALUPP, M. P.; DE CARVALHO, A. C. P. L. F.; RUIZ, D. D.; DE SOUZA, O. N. Automatic Design of Decision-Tree Induction Algorithms Tailored to Flexible-Receptor Docking Data. *BMC Bioinformatics*, 13, 310, 2012. Qualis A1.
- BARROS, R. C.; BASGALUPP, M. P.; DE CARVALHO, A. C. P. L. F.; FREITAS, A. A. Automatic Design of Decision-Tree algorithms with Evolutionary Algorithms. *Evolutionary Computation*, 21:4, 2013. Qualis A1.
- BARROS, R. C.; BASGALUPP, M. P.; FREITAS, A. A.; DE CARVALHO, A. C. P. L. F. Evolutionary Design of Decision-Tree Algorithms Tailored to Microarray Gene Expression Data Sets. *IEEE Transactions on Evolutionary Computation*, in press, 2014. Qualis A1.

[Awards]

Best paper in the IGEC+S*S+SBSE tracks at GECCO 2012: BARROS, R. C.; BASGALUPP, M. P.; DE CARVALHO, A. C. P. L. F.; FREITAS, A. A. A Hyper-Heuristic Evolutionary Algorithm for Automatically Designing Decision-Tree Algorithms. *: 14th Annual Genetic and Evolutionary Computation Conference (GECCO 2012)*, 1237–1244, 2012.

This award was theme of a news report by *Agência FAPESP*[2], and also of the technology news report of Diário Braziliense, edition of August 22 2012 (see Fig. 1). Note that the venues that were chosen for publishing the work presented in the defended thesis included the top international conference in evolutionary computation (GECCO - Qualis A1) and also the two top international journals of evolutionary computation (IEEE Transactions on Evolutionary Computation, and MIT Evolutionary Computation, both Qualis A1).

Tecnologia

20 • CORREIO BRAZILIENSE • Brasília, quarta-feira, 22 de agosto de 2012

O computador que gera frutos

Inspirado na Teoria da Evolução de Darwin, aluno de doutorado da USP desenvolve programa que combina diferentes fórmulas e cria algoritmos por conta própria. O projeto, premiado nos EUA, pode ser usado nas mais diversas áreas, incluindo a financeira e a médica

• ROBERTA MACHADO

Se uma companhia de cartões de crédito quer saber uma relação de que apontar quais transações podem ser fruto de uma fraude financeira, não há ferramenta melhor para essa tarefa como um computador. O primeiro passo é alimentar a máquina com os dados históricos a respeito do crime para que ela crie uma relação completa de todas as compras efetuadas no passado. A próxima etapa é mais complicada: ensinar o sistema a calcular a probabilidade de as próximas transações serem criminosas, com base no padrão das fraudes anteriores. É aí que entram os chamados algoritmos.

Essas fórmulas capazes de avaliar milhares de

Calculamos que, com os componentes atuais, o programa é capaz de evoluir até o melhor algoritmo entre 1/10 milhões de combinações possíveis. Nossa meta é aumentar esse espaço de busca para alguns bilhões de combinações, na esperança de conseguirmos resultados

Figure 1. Excerpt from the technology news report of Diário Braziliense, August 22 2012.

[2]http://agencia.fapesp.br/16041

References

Barros, R. C., Basgalupp, M. P., de Carvalho, A. C. P. L. F., and Freitas, A. A. (2012a). A Survey of Evolutionary Algorithms for Decision-Tree Induction. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 42(3):291–312.

Barros, R. C., Basgalupp, M. P., Freitas, A. A., and de Carvalho, A. C. P. L. F. (2014). Evolutionary Design of Decision-Tree Algorithms Tailored to Microarray Gene Expression Data Sets. *IEEE Transactions on Evolutionary Computation*, in press.

Barros, R. C., Cerri, R., Jaskowiak, P. A., and de Carvalho, A. C. P. L. F. (2011). A Bottom-Up Oblique Decision Tree Induction Algorithm. In *11th International Conference on Intelligent Systems Design and Applications*, pages 450–456.

Barros, R. C., Winck, A. T., Machado, K. S., Basgalupp, M. P., de Carvalho, A. C. P. L. F., Ruiz, D. D., and de Souza, O. S. (2012b). Automatic design of decision-tree induction algorithms tailored to flexible-receptor docking data. *BMC Bioinformatics*, 13.

Basgalupp, M. P., Barros, R. C., da Silva, T. S., and de Carvalho, A. C. P. L. F. (2013). Software effort prediction: a hyper-heuristic decision-tree based approach. In *28th Annual ACM Symposium on Applied Computing*, pages 1109–1116.

Bennett, K. and Mangasarian, O. (1994). Multicategory discrimination via linear programming. *Optimization Methods and Software*, 2:29–39.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Cowling, P., Kendall, G., and Soubeiga, E. (2001). A hyperheuristic approach to scheduling a sales summit. In Burke, E. and Erben, W., editors, *Practice and Theory of Automated Timetabling III*, volume 2079 of *Lecture Notes in Computer Science*, pages 176–190. Springer Berlin Heidelberg.

Kim, B. and Landgrebe, D. (1991). Hierarchical classifier design in high-dimensional numerous class cases. *IEEE Transactions on Geoscience and Remote Sensing*, 29(4):518–528.

Pappa, G. L., Ochoa, G., Hyde, M. R., Freitas, A. A., Woodward, J., and Swan, J. (2013). Contrasting meta-learning and hyper-heuristic research: the role of evolutionary algorithms. *Genetic Programming and Evolvable Machines*.

Rokach, L. and Maimon, O. (2005). Top-down induction of decision trees classifiers - a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 35(4):476 – 487.

Smith-Miles, K. A. (2009). Cross-disciplinary perspectives on meta-learning for algorithm selection. *ACM Computing Surveys*, 41:6:1–6:25.

Wolpert, D. H. and Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82.