

Decision Tree-Based Data Characterization for Meta-Learning

Yonghong Peng¹ Peter A Flach¹ Pavel Brazdil² Carlos Soares²

¹Department of Computer Science, University of Bristol, UK

{yh.peng, peter.flach}@bristol.ac.uk

²LIACC/Fac. of Economics, University of Porto, Portugal

{pbrazdil, csoares}@liacc.up.pt

Abstract. Appropriate selection of learning algorithms is essential for the success of data mining. Meta-learning is one approach to achieve this objective by identifying a mapping from data characteristics to algorithm performance. Appropriate data characterization is, thus, of vital importance for the meta-learning. To this effect, a variety of data characterization techniques, based on three strategies including simple measure, statistical measure and information theory based measure, have been developed, however, the quality of them is still needed to be improved. This paper presents new measures to characterise datasets for meta-learning based on the idea to capture the characteristics from the structural shape and size of the decision tree induced from the dataset. Their effectiveness is illustrated by comparing to the results obtained by the classical data characteristics techniques, including DCT that is the most wide used technique in meta-learning and Landmarking that is the most recently developed method and produced better performance comparing to DCT.

1 Introduction

Extensive research has been performed to develop appropriate machine learning techniques for different data mining problems, and has led to a proliferation of different learning algorithms. However, previous work has shown that no learner is generally better than another learner. If a learner performs better than another learner on some learning situations, then the first learner must perform worse than the second learner on other situations [18]. In other words, no single learning algorithm can perform well and uniformly outperform other algorithms over all data mining tasks. This has been confirmed by the ‘no free lunch theorems’ [31,32]. The major reasons are that a learning algorithm has different performance in processing different dataset and different learning algorithms are implemented with different search heuristics, which results in variety of ‘inductive bias’ [15]. In real-world applications, the users need to select an appropriate learning algorithm according to the mining task that they are going to

perform [17,18,1,7,20,12]. An inappropriate selection of algorithm will result in slow convergence, or even produce a sub-optimal solution due to a local minimum.

Meta-learning has been proposed to deal with the issues of algorithm selection [5, 8]. One of the aims of meta-learning is assisting the user to determine the most suitable learning algorithm(s) for the problem at hand. The task of meta-learning is to find functions that map datasets to predicted data mining performance (e.g., predictive accuracies, execution time, etc.). To this end meta-learning uses a set of attributes, called meta-attributes, to represent the characteristics of data mining tasks, and search for the correlations between these attributes and the performance of learning algorithms in general or the optimal learning algorithm in particular [5,10,12]. Instead of executing all learning algorithms to obtain the optimal one, meta-learning is performed on the meta-data characterising the data mining tasks. Algorithm selection is performed by executing the meta-model induced on the characteristics of the dataset.

Three basic procedures are involved in meta-learning: 1) describing the characteristics of learning tasks using a set of meta-attributes; 2) developing the correlations between the meta-attributes and the performance of learning algorithms or the optimal learning algorithms, which is called meta-knowledge; 3) to search, given a new learning task, the optimal learning algorithm(s) according to the developed meta-knowledge. It is obvious that the effectiveness of meta-learning is largely dependent on the description of tasks (i.e., meta-attributes). Several techniques have been developed, such as data characterisation techniques (DCT) [13] to describe the problem to be analyzed, including simple measures (e.g. number of attributes, classes et al.), statistical measures (e.g. mean and variance of numerical attributes), and information theory-based measures (e.g. entropy of classes and attributes). There is, however, still a need for improving the effectiveness of meta-learning by developing more predictive meta-attributes and selecting the most informative ones [9].

In [3], the authors suggested to characterize dataset by measuring the characteristic of models induced on the dataset. Inspired by this idea, this paper presents new methods to measure the complexity of classification data mining tasks. The complexity of data mining tasks is related to the characteristics of datasets and the inductive bias of learning algorithms. The basic idea is to investigate the possibility of capturing dataset characteristics by measuring the properties of a decision tree induced from the dataset, i.e., to measure the structural shape and size of the tree generated by standard methods (c5.0 [22] is used in this paper). More specifically, these measures capture the structural properties of decision tree by some simple measures counting the number of nodes, leaves and attributes in the tree. The extracted meta-attributes have been applied in ranking-based meta-learning for classification algorithm selection. The experimental results clearly show the enhancement of ranking performance compared to the DCT techniques, which is the most commonly used technique, and landmarking, a recently introduced technique [19,2].

This paper is organized as following. In section 2, some related work is introduced, including meta-learning methods for algorithm selection and data characterisation techniques. The proposed method for characterising the datasets is stated in detail in section 3. Experiments illustrating the effectiveness of the proposed method are de-

scribed in section 4. Section 5 concludes the paper, and points out interesting possibilities for future work.

2 Related Work

Two basic factors are involved in meta-learning: the description of the learning tasks (datasets), and the correlation between the task description and the optimal learning algorithm. The first aspect is associated to techniques to characterise datasets with meta-attributes, whilst the second is the learning at meta-level, which develops the meta-knowledge for selecting appropriate algorithm in classification.

2.1 Work Related to Meta-Learning for Algorithm Selection

For algorithm selection, several meta-learning strategies have been proposed [6,25,26]. In general, there are three options in generating the output of the meta-learner. One is to select a single learning algorithm, i.e. to select the algorithm that is expected to produce the best model for the dataset. The second is to select a subgroup of learning algorithms, including not only the best algorithm but also the algorithms that are not significantly worse than the best one. The third possibility is to rank the learning algorithms according to their performance. The ranking will assist the user to finally select the learning algorithm. This ranking-based meta-learning is the main approach in the Esprit Project MetaL (www.metal-kdd.org).

Ranking the preference order of algorithms is performed based on estimating the performance of algorithms. In data mining, performance can be measured not only in terms of accuracy but also time or understandability of model generated. In this paper, we assess performance with the Adjusted Ratio of Ratios (ARR) measure, which combines the accuracy and time. ARR gives a measure of the advantage of a learning algorithm over another algorithm in terms of their accuracy and the execution time for a specific dataset. The user can adjust the importance of accuracy relative to time by a tunable parameter. The ‘zoomed ranking’ method proposed by Soares [26] based on ARR, which will be described briefly in section 4.1, is used in this paper for algorithm selection, taking into account of accuracy and execution time simultaneously.

2.2 Work Related to Dataset Characterization

As different learners exhibit sensitivity to specific characteristics of the dataset, the task of meta-learning is to model how these characteristics affect the relative performance of different learning algorithms, and then predict the preference for each learning algorithm before performing data mining process. The methods used to describe the characteristics of the dataset were called Data Characterization Tool (DCT) [13].

The first attempt to characterise datasets in order to predict the performance of classification algorithm was done by Rendell et al. [23]. So far, two main strategies

have been developed in order to characterise a dataset for algorithm selection. First one describes the properties of datasets using statistical and informational measures. In the second one a dataset is characterised using the performance (e.g. accuracy) of a set of simple learners, called landmarker [19,2].

The description of a dataset in terms of its information/statistical properties, appeared for the first time within the framework of the STATLOG project [14]. The authors used a set of 15 characteristics, spanning from simple ones, like the number of attributes or the number of examples, to more complex ones, such as the first canonical correlation between the attributes and the class. This set of characteristics was later applied in various studies, aimed at solving the problem of algorithm selection [5,29,27]. They distinguish three categories of dataset characteristics, namely simple, statistical and information theory based measures. Statistical characteristics are mainly appropriate for continuous attributes, while information theory based measures are more appropriate for discrete attributes. Linder and Studer [13] provide an extensive list of information and statistical measures of a dataset computed for each attribute or pairs of attributes. They provide a tool for the automatic computation of these characteristics, which was called Data characterisation Tools (DCT). However, they pointed out that only a limited set of these measures is relevant in providing recommendation, which in fact was very similar to the one defined in STATLOG. Sohn [27], also uses the STATLOG set as a starting point, and she proceeds with careful evaluation of their properties in a statistical framework. She discovers that some of the characteristics are highly correlated, and she omits the redundant ones from her study. Furthermore, she introduces new features that are transformation or combinations of the existing ones, like ratios or second powers, with the goal of providing successful predictions.

An alternative approach to characterise datasets called landmarking was proposed in [19,2]. The intuitive idea behind landmarking is that the performance of simple learner, landmarker, can be used to predict the performance of given candidate algorithms. That is, given landmarker A and B, if we know landmarker A outperforms landmarker B on the present task, then we could select the learning algorithms that has the same inductive bias of landmarker A to perform this data mining task. It has to be ensured that the chosen landmarkers have quite distinct learning biases. As a closely related approach, Bensusan [3, 33] had also used the information computed from the induced decision trees to characterise tasks in meta-learning, such as the ratio of the number of nodes to the number of the attributes, the ratio of number of nodes to the number of training instances. He listed 10 measures based on the unpruned tree, but the performance of these measures in algorithm selection was not evaluated.

3 The proposed measures for describing data characteristics

The task of characterizing dataset for meta-learning is to capture the information about learning complexity for the dataset. This information should enable the estimation of performance of the given learning algorithms. It should also be computable within a relative short time comparing to the whole learning process, which is desired to be

predictive in estimating the performance of the given learning algorithms. In this section we introduce new measures to measure the characteristics of the dataset based on measuring a variety of properties of a decision tree induced from that dataset.

The major idea here is to measure the complexity of learning by measuring the structure and size of decision tree, and use these measures to predict the model complexity generated by other learning algorithms. We employed the standard decision tree learner, c5.0tree. There are several reasons for selecting decision trees. The major reason is that decision tree has been one of the most popularly used machine learning algorithms in classification, and the induction of decision tree is deterministic, i.e. the same training set always produces the similar structure of decision tree.

Definition. A standard *tree* induced with c5.0 (or possibly ID3 or c4.5) consists of a number of *branches*, one *root*, a number of *nodes* and a number of *leaves*. One branch is a chain of *nodes* from *root* to a *leaf*, and each node involves one attribute. The *occurrence* of an attribute is the number of times the attribute occurs in the tree, which provides the information about the importance of the associated attribute. The *tree width* is defined as the number of lengthways partitions divided by parallel nodes or leave from the leftmost to the rightmost nodes or leave. The *tree level* is defined as the breadth-wise partition of tree at each success branches, and the *tree height* is defined by the number of tree levels, as shown in Fig.1. The *length of a branch* is defined as the number of nodes in the branch minus one.

We propose, based on above notations, to describe decision tree in term of the following three aspects: a) outer-profile of tree; b) statistic for intra-structure: including tree levels and branches; c) statistic for tree elements: including nodes and attributes.

To describe the outer-profile of the tree, the width of tree (*treewidth*) and the height of the tree (*treeheight*) are measured according to the number of nodes in each level and the number of levels, as illustrated in Fig.1. Also, the number of nodes (*NoNode*) and the number of leaves (*NoLeave*) are used to describe the overall property of a tree. In order to describe the intra-structure of the tree, the number of nodes at each level and the length of each branch are counted. Let us represent them with two vectors denoted as $NoinL=[v_1, v_2, \dots, v_l]$ and $LofB=[L_1, L_2, \dots, L_b]$ respectively, where v_i is the number of nodes at the i th level, L_j is the length of j th branch, l and b is the number of levels (*treeheight*) and number of branches. Based on $NoinL$ and $LofB$, four measures are generated. The maximum and minimum number of nodes at one level:

$$maxLevel = \max(v_1, v_2, \dots, v_l) \quad (1)$$

$$minLevel = \min(v_1, v_2, \dots, v_l)$$

(As the $minLevel$ is always equal to 1, it is not used.) The mean and standard deviation of the number of nodes and leaves on levels:

$$meanLevel = \left(\sum_{i=1}^l v_i \right) / l, \quad (2)$$

$$devLevel = \sqrt{\sum_{i=1}^l (v_i - meanLevel)^2 / (l-1)}$$

The length of longest and shortest branches:

$$LongestBranch = \max(L_1, L_2, \dots, L_b) \quad (3)$$

$$ShortestBranch = \min(L_1, L_2, \dots, L_b)$$

The mean and standard deviation of the length of each branch:

$$meanBranch = \left(\sum_{j=1}^b L_j \right) / b, \quad (4)$$

$$devBranch = \sqrt{\sum_{j=1}^b (L_j - meanBranch)^2 / (b-1)}$$

Besides the distribution of nodes, the frequency of attributes used in a tree provides further information regarding the dataset. For that, we calculate the times each attribute is used in a tree, which is represented by a vector $NoAtt = [nAtt_1, nAtt_2, \dots, nAtt_m]$, where $nAtt_k$ is the number of times the k th attribute is used and m is the total number of attributes in the tree. Again, the following measures are used:

The maximum and minimum occurrence of attributes:

$$maxAtt = \max(nAtt_1, nAtt_2, \dots, nAtt_m) \quad (5)$$

$$minAtt = \min(nAtt_1, nAtt_2, \dots, nAtt_m)$$

Mean and standard deviation of the number of occurrences of attributes:

$$meanAtt = \left(\sum_{i=1}^m nAtt_i \right) / m, \quad (6)$$

$$devAtt = \sqrt{\sum_{i=1}^m (nAtt_i - meanAtt)^2 / (m-1)}$$

As a result, a total of 15 meta-attributes is used in our experiments.

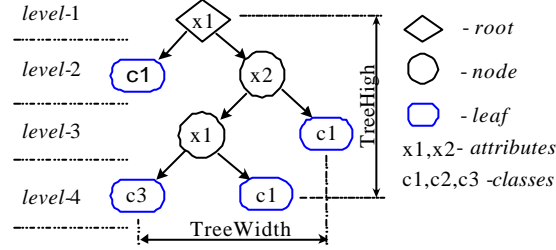


Fig. 1. Structure of Decision Tree.

4 Experimental Evaluation

In this section we experimentally evaluate the proposed data characteristics. In section 4.1 we describe our experimental set-up, in section 4.2 we compare our proposed meta-features with DCT and landmarking, and in section 4.3 we study the effect of meta-feature selection, and compare the performance of DCT and our methods for selected number of meta-features.

4.1 Experimental set-up

The technique of meta-learning employed in this paper is called ranking with *zooming* [26], which includes two phases: 1) training phase to collect the meta-data; 2) reasoning phase to rank the candidate learning algorithms for a given data mining task.

In the training phase, all the benchmark datasets are characterised using the data characterisation methods (e.g., DCT, landmarking or the method proposed in this paper). As a result, one dataset is described with a vector of a set of meta-attributes. These meta-attributes together with the analyzed performance (including accuracy and time) constitute the meta-data. In the reasoning phase, two steps are involved: 1) given a data mining problem (a dataset to analyze), the k-Nearest Neighbor (kNN) algorithm is used to select a subset with k dataset from the benchmark datasets, whose characteristics are similar to the characteristics of the present dataset according to some distance function; this step is called zooming [26]; 2) ranking the order of preference of candidate learning algorithms according to their performance on these datasets selected in zooming phase; this step is named ranking. The ranking is performed based on the *adjusted ratio of ratios* (ARR), a multi-criteria evaluation measure that combine the predicated accuracy and time. ARR has a parameter to enable the user to adjust the relative importance of accuracy and time according to fulfill his particular data mining objective. More details can be found in [26].

To evaluate a recommended ranking, we calculate its correlation to an ideal ranking obtained for the same dataset. The ideal ranking is obtained by estimating the performance of the candidate learning algorithms using 10-fold cross-validation. The similarity between the generated ranking and the ideal ranks is measured using the Spearman's rank correlation coefficient [30].

$$r_s = 1 - \frac{6D^2}{n(n^2 - 1)}, D^2 = \sum_{i=1}^n D_i^2 = \sum_{i=1}^n (r_i - \bar{r}_i)^2 \quad (7)$$

where the r_i and \bar{r}_i are the predicted ranking and actual ranking for algorithm i respectively. The bigger r_s is, better of ranking result is, with $r_s = 1$ if the ranking is same as the ideal ranking.

4.2 Comparison with DCT and Landmarking

In our experiments, a total of 10 learning algorithms, including *c5.0tree*, *c5.0boost* and *c5.0rules* [21], Linear Tree (*ltree*), linear discriminant (*lindiscr*), MLC++ Naive Bayes classifier (*mlcnb*) and Instance-based learner (*mlcib1*) [11], Clementine Multilayer Perceptron (*clemMLP*), Clementine Radial Basis Function (*clemRBFN*) and rule learner (*ripper*), have been evaluated on 47 datasets, which are mainly from the UCI repository [4]. The error rate and time were estimated using 10-fold cross-validation. Our aim in this paper is to evaluate the effect of new proposed meta-attributes (called *DecT* from now on) on ranking of these 10 learning algorithms. In other words, we are interested in comparing the rankings generated by DecT (15 meta-attributes) to the ranking generated by DCT (25 meta-attributes) and Landmarking (5 meta-attributes).

The first experiment is performed to rank the given 10 learning algorithms on the 47 datasets. The leave-one-out method is used to evaluate the performance of ranking, i.e., the performance for ranking the 10 given learning algorithms for each dataset on the basis of the other 46 datasets. In the first experiment, the parameters $k=10$, $Kt=100$, meaning that we are willing to trade 1% in accuracy for a 10 times speed-up or slow-down. The ranking performance is measured with r_s (Eq. (15)). The results of ranking performance of using DCT, landmarking and DecT are shown in Fig. 2. The overall average performance for DCT, Landmarking and DecT are 0.613875, 0.634945 and 0.676028 respectively, which demonstrates the improvement of using DecT in ranking algorithms, comparing to DCT and Landmarking.

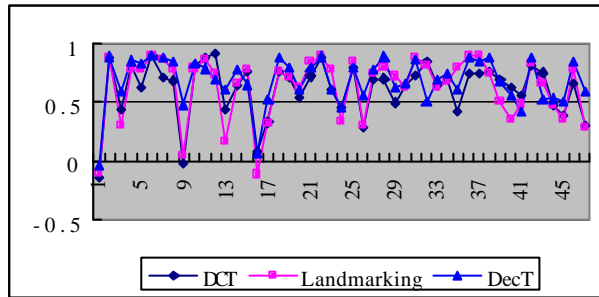


Fig. 2. Ranking performance for 47 datasets using DCT, landmarking and DecT.

In order to look in more detail at the improvement of DecT over DCT and Landmarking, we performed the experiment of ranking using different values of k and Kt . As stated in [26], the parameter Kt represents the relative importance of accuracy and execution time in selecting the learning algorithm (i.e., higher Kt means the accuracy is more important and time is less important). Fig.3 shows the ranking performances of DCT, landmarking and DecT along with different values of $Kt=\{10, 100, 1000\}$, from which it is observed that, for all the used Kt , DecT improves the performance with different increased degree, comparing to DCT and landmarking.

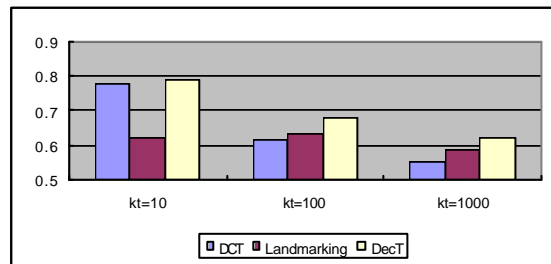


Fig. 3. The ranking performance for different values of Kt .

Fig. 4 shows the performance of ranking based on different zooming degree (different k), i.e., selecting different number of similar datasets, based on which the ranking is performed. From these results, we observe that 1) for all different values of k , DecT

produces better ranking performance than DCT and landmarking; 2) best performance is obtained by selecting 10-25 datasets among 46 datasets.

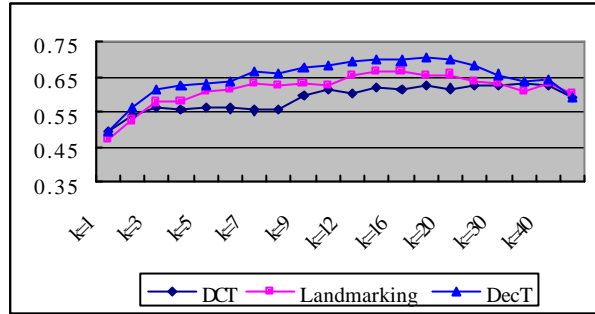


Fig. 4. The ranking performance for different values of k.

4.3 Performing meta-feature selection

The k-nearest neighbor learning method, employed to select k datasets for ranking the performance of learning algorithms for the given dataset, is known to be sensitive to the irrelevant and redundant features. Using smaller number of features could help to improve the performances of k-nearest neighbor learning, as well as to reducing the time used in meta-learning. In our experiments, we manually reduced the number of DCT meta-features from 25 to 15 and 8, and compare their results to those obtained based on the same number of DecT meta-features. The reduction for DCT meta-features is performed by removing the features thought to be redundant, and the features having a lot of *non-appl* values, and the reduction for DecT meta-features are performed by removing redundant features.

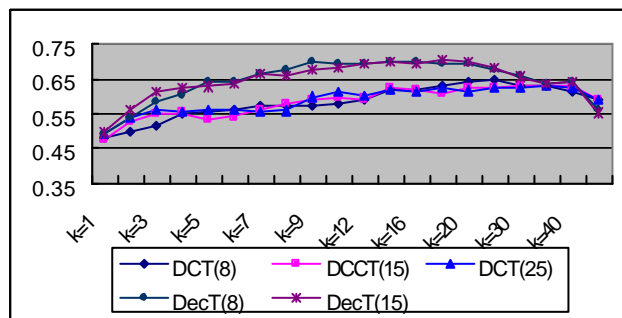


Fig.5. Results for reduced meta-features.

The ranking performances for these reduced meta-features are shown in Fig.5, in which DCT(8), DCT(15), DecT(8) represent the reduced 8, 15 DCT meta-features and 8

DecT meta-features, DCT(25) and DecT(15) represent the full DCT and DecT meta-features respectively. From Fig.5, we can observe that feature selection did not significantly influence the performance of either DCT or DecT, and that the latter outperforms the former across the board.

5 Conclusions and Future Work

Meta-learning strategy, under the framework of MetaL, aims at assisting the user in selecting appropriate learning algorithm for the particular data mining task. Describing the characteristics of dataset in order for estimating the performance of learning algorithm is the key to develop a successful meta-learning system.

In this paper, we proposed new measures to characterise the dataset. The basic idea of is to process the dataset using a standard tree induction algorithm, and then to capture the information regarding the dataset's characteristics from the induced decision tree. The decision tree is generated using standard `c5.tree` algorithm. A total of 15 measures, which constitute the meta-attributes for meta-learning, have been proposed for describing different kind of properties of a decision tree.

The proposed measures have been applied in ranking the learning algorithms based on accuracy and time. Extensive experimental results have illustrated the improvement of ranking performance by using the 15 meta-attributes generated by the proposed method, compared to the 25 DCT and 5 Landmarking meta-features. In order to reduce the effect of redundant or irrelevant features on the performance of zooming ranking, we also compared the performance based on selected 15 DCT meta-features and DecT, and selected 8 DCT and DecT meta-features. The results suggest that feature selection does not significantly change the performance of either DCT or DecT.

In other experiments, we observed that the combination of DCT with DecT or Landmarking with DCT and DecT did not produce better performance than DecT. This is an issue that we are interested in further investigation. The major reason may come from the use of k-nearest neighbor learning in zooming based ranking strategy. One possibility is to test the performance of the combination of DCT, landmarking and DecT in other meta-learning strategies, such as best algorithm selection. Another interesting subject is to look at the change of shape and size of the decision tree along with the change of examples used in tree induction, as it will be useful if it is possible to capture the data characteristics based on sampled dataset. This is especially important for large datasets.

Acknowledgements: this work is supported by the MetaL project (ESPRIT Reactive LTR 26.357).

References

1. Brodley, C. E.: Recursive automatic bias selection for classifier construction. *Machine Learning*, (1995) 20:63-94.
2. Bensusan, H., and Giraud-Carrier, C.: Discovering Task Neighbourhoods through Landmark Learning Performances. In Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery. 325-330. Springer. (2000)
3. Bensusan, H., Giraud-Carrier, C., and Kennedy, C.: Higher-order Approach to Meta-learning. In Proceedings of the ECML'2000 workshop on Meta-Learning: Building Automatic Advice Strategies for Model Selection and Method Combination, 109-117. ECML'2000. (2000)
4. Blake, C., Keogh, E., and Merz, C.: www.ics.uci.edu/~mllearn/mlrepository.html. University of California, Irvine, Dept. of Information and Computer Sciences. (1998)
5. Brazdil, P., Gama, J. and Henery, R.: Characterizing the Applicability of Classification Algorithms using Meta Level Learning. *Machine Learning-ECML94*. (1994) 83-102, Springer Verlag.
6. Dietterich, T G.: Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, , (1998) 10(7):1895-1924.
7. Gordon F. and desJardin, M.: Evaluation and selection of biases. *Machine Learning*, (1995) 20:5-22.
8. Kalousis A. and Hilario, M.: Model Selection via Meta-learning: a Comparative Study. In *Proceedings of the 12th International IEEE Conference on Tools with AI*, Vancouver. IEEE press. (2000)
9. Kalousis, A. and Hilario, M.: Feature Selection for Meta-Learning. In *Proceedings of the 5th Pacific Asia Conference on Knowledge Discovery and Data Mining*. Springer. (2001)
10. Koepf, C., Taylor, C. and Joerg Keller J.: Meta-analysis: Data characterisation for classification and regression on a meta-level. In Antony Unwin, Adalbert Wilhelm, and Ulrike Hofmann, editors, *Proceedings of the International Symposium on Data Mining and Statistics*, Lyon, France, (2000).
11. Kohavi, R.: Scaling up the Accuracy of Naïve-bayes Classifier: a Decision Tree hybrid. Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining, (1996) 202-207.
12. Lagoudakis, M.G. and Littman, M. L.: Algorithm selection using reinforcement learning. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML-2000)*, 511-518, Stanford, CA. (2000)
13. Linder, C. and Studer, R.: AST: Support for Algorithm Selection with a CBR Approach. Proceedings of the 16th International Conference on Machine Learning, Workshop on Recent Advances in Meta-Learning and Future Work. (1999).
14. Michie, D., Spiegelhalter, D., and Taylor, C.: *Machine Learning, Neural Network and Statistical Classification*. Ellis Horwood Series in Artificial Intelligence. (1994)
15. Mitchell, T.: *Machine Learning*. MacGraw Hill. (1997)
16. Salzberg. S.: On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Mining and Knowledge Discovery, Vol. 1*. (1997)

17. Schaffer, C.: Selecting a Classification Methods by Cross Validation, *Machine Learning*, 13, 135-143. (1993)
18. Schaffer, C.: Cross-validation, stacking and bi-level stacking: Meta-methods for classification learning. In P. Cheeseman and R. W. Oldford, editors, *Selecting Models from Data: Artificial Intelligence and Statistics IV*, pages 51-59. Springer-Verlag. (1994)
19. Pfahringer, B., Bensusan, H., and Giraud-Carrier, C.: Landmarking various Learning Algorithms. Proceedings of the 17th International Conference on Machine Learning. 743-750. Morgan Kaufman. (2000)
20. Provost F.J. and Buchanan B. G: Inductive policy: The pragmatics of bias selection. *Machine Learning*, 20:35-61. (1995)
21. Quinlan, J. R.: C4.5: Programs for Machine Learning, Morgan Kaufman. (1993)
22. Quinlan, J. R.: c5.0: An Informal Tutorial, RuleQuest, www.rulequest.com/see5-unix.html. (1998).
23. Rendell, L. Seshu, R., and Tchong, D.: Layered Concept Learning and Dynamically Variable Bias Management. 10th Inter. Join Conference on AI. 308-314. (1987).
24. Schaffer, C.: A Conservation Law for Generalization Performance. Proceedings of the 11th International Conference on Machine Learning. (1994).
25. Soares, C.: Ranking Classification Algorithms on Past Performance. Master's Thesis, Faculty of Economics, University of Porto. (2000)
26. Soares, C.: Zoomed Ranking: Selection of Classification Algorithms based on Relevant Performance Information. Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery, 126-135. Springer. . (2000)
27. Sohn, S.Y.: Meta Analysis of Classification Algorithms for Pattern Recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, (1999) 21, 1137-1144.
29. Todorovski, L. and Dzeroski, S.: Experiments in Meta-Level Learning with ILP. Proceedings of the 3th European Conference on Principles on Data Mining and Knowledge Discovery, 98-106. Springer. (1999)
30. Webster, A.: *Applied Statistics for Business and Economics*, Richard D Irwin Inc, 779-784. (1992).
31. Wolpert, D.: The lack of a Priori Distinctions between Learning Algorithms. *Neural Computation*, 8, 1341-1390. (1996)
32. Wolpert, D.: The Existence of a Priori Distinctions between Learning Algorithms. *Neural Computation*, 8, 1391-1420. (1996).
33. Bensusan, H. God doesn't always shave with Occam's Razor - learning when and how to prune. In *Proceedings of the 10th European Conference on Machine Learning*, pages 119--124, Berlin, Germany, April 1998. Springer.