Improving ontological knowledge with reinforcement in recommending the data mining method for real problems

Karina Gibert and Miquel Sànchez-Marrè

Universitat Politècnica de Catalunya-BarcelonaTech, Jordi Girona 1-3, 08034 Barcelona
karina.gibert@upc.edu, miquel@cs.upc.edu

Abstract. There are many data mining techniques available for a user wishing to discover some model from her/his data. This diversity can cause some troubles to the final non-expert users, who often do not have a clear idea of what are the available methods, and frequently have doubts about the most suitable method for a concrete problem in a domain. In previous works, prior ontological knowledge about data mining methods has been used to describe the main characteristics of a collection of methods and to filter which methods are suitable or not for a given real data mining problem, by matching their characteristics with those hold in the target dataset. In this paper, the concept of reinforcement tables is introduced to move to a multi-criteria scenario in which a measure of relevance is computed for every method. A contribution of the work is to develop an open-frame where both the characteristics of methods considered in the reference ontology and the reinforcement tables may evolve along time according to changes in the methodological state of the art, going beyond classical expert systems. The paper introduces the formal framework and some examples to illustrate the performance of the proposal.

Keywords: recommender system for data mining methods, knowledge-based recommender system, data mining, ontological knowledge, reinforcement

1 Introduction

The classical scheme of Knowledge Discovery from Data (KDD) [1] is well known: (a) Developing and understanding the domain; (b) Creating the target data; (c) Data cleaning and pre-processing; (d) Data reduction and projection; (e) Choosing the data mining task. See [2, 3] for a survey of the most common data mining techniques; (f) Selecting the data mining algorithm/s; (g) Data mining; (h) Interpreting mined patterns; (i) Consolidating discovered knowledge.

One of the most critical decisions to be made to guarantee the best outcome for a given dataset (often not enough carefully addressed) is selecting the more appropriate DM algorithm(s) to be used to analyse the data. Indeed, most of the commercial Data Mining systems provide collections of several data mining methods, and is responsibility of the data miner himself to use them properly in every particular application. Moreover, the map of possibilities changes dynamically and rapidly, and the decision requires more and more expertise from the methodological point of view. However,

those commercial software tools rarely provide intelligent assistance for addressing these decisions or tend to do it in the form of rudimentary "wizard-like" interfaces.
In this paper, we are proposing an open frame able to recommend the best method for a given problem considering the problem goals, the characteristics of data, a knowledge-base of methods and some costs or benefits parameters associated with wrong decisions. The system is able evaluate the relevance of a method for a given real application, that might help the non-expert data miner to choose the proper data mining method at every case, or a subset of best methods to be explored. Although the proposal is generic, the paper illustrates it in a particular case where a certain set of data mining methods are considered..
The structure of the paper is: In the next section, the related work is introduced. In section 3, a conceptual map of data mining methods is briefly presented. An associated method properties knowledge base is described in section 4. In section 5, the proposed reinforcement measures and policies are described. In section 6, some experimental testing examples are detailed and the results illustrate the benefits of this new methodology. Finally, in section 7 the conclusions and future work are discussed.

2 Related Work

There are some works in the literature addressing these issues. In [4], Ontological Learning Assistant (OLA) is introduced, as an ontology-based KDD Support Environment for carrying out the knowledge discovery process. In [5], authors proposed and implemented a hybrid data mining assistant, based on the use of Case-Based Reasoning (CBR) paradigm and the use of a formal OWL-DL ontology. In a previous work [6] they proposed another hybrid intelligent data mining assistant, based on the combination of both declarative (Description Logic) and procedural (SWRL Rules) ontology knowledge to assist in the whole KDD process. Some approaches in the literature have proposed the use of Ontologies and CBR techniques like in the CBR platform jCOLIBRI2 [7]. Oprean [8] proposed a general architecture for assisting in the KDD process; nevertheless, only the first two DM steps of the CRISP-DM methodology [9] are implemented: Business Understanding and Data Understanding. Moreira Pereira [10] proposed a system to allow recommendation and use of the most promising DM algorithms, through the improvement of the DM Advisor tool, a plug-in of RapidMiner tool [11], and its integration with OpenML [12] to get a distributed Intelligent Discovery Assistant. DM4D is a pure CBR system using Spearman correlation to rank the methods before recommendation. In [13] a survey on intelligent assistants for data analysis is provided. All systems developed along 28 years of research follow either pure CBR approaches, or pure Knowledge-based approaches, but none uses a reinforcement policy. In [14], a high-level description of a number of most popular Data Mining techniques was presented. A conceptual map of data mining techniques regarding the parameters used by humans to make decisions was introduced as a help to non-expert data miners to find the more suitable techniques for a particular application. In [15] a case-based reasoning (CBR) intelligent recommender (INtelligent DAta MIner TEchniques REcommender, i.e. INDAMITERE) was introduced, in such a way that past experiences using the GESCONDA tool [16] could be

used to suggest more suitable algorithms for a given problem. However, the CBR performance is not taking into account any previous domain knowledge available, whereas, on the domain of data mining methods, high quantity of theoretical knowledge is suitable to be used to be taken into account.

In this work the use of an ontological description of data mining methods is improved by introducing reinforcement measures that increases accuracy and consistency of the recommendations. Learning from meta-characteristics of data mining methods is the main principle of meta-learning [25], and lots of efforts were done to find the right meta-characteristics to be considered [26, 27] and to address a specific set of methods. The proposed approach works as a generic meta-learner where both the set of methods considered as well as the meta-characteristics used to describe the methods are parameterized and might be easily updated along time, according to the improvements in the Data Mining State of the Art.

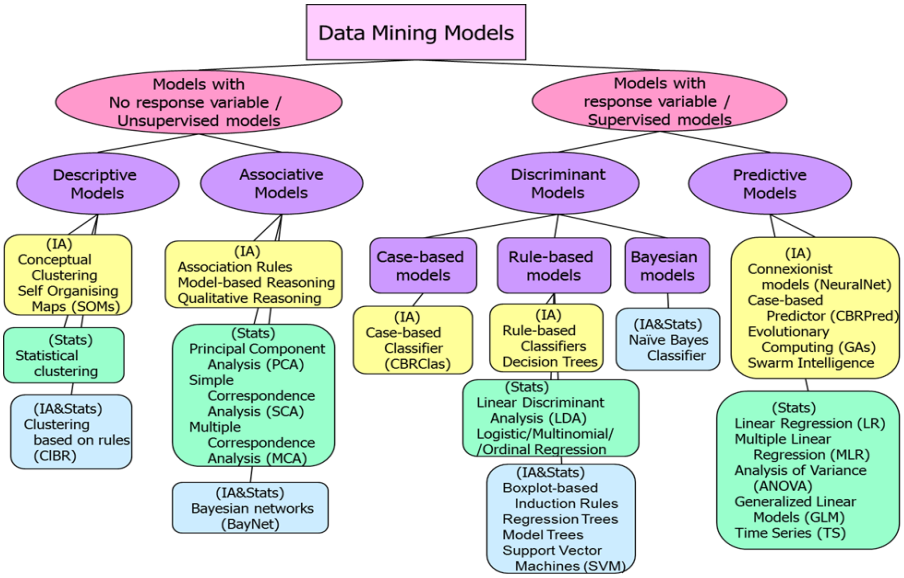

Fig. 1: Conceptual Taxonomy Map of Data Mining Techniques (DMMCM map)

3 Data Mining Methods Conceptual Map (DMMCM map)

Althoug the presented proposal can work with any conceptual map containing a catalogue of data mining methods, in [14] a conceptual map for more popular DM techniques was introduced, based in the different DM problems solved by the methods (see Fig. 1). The taxonomical knowledge is organized in to two main levels:

- The first level of the taxonomy splits the DM models without response variable on the left (unsupervised models), and the DM models with response variable (supervised models) to the right. The left part of the chart refers to non-supervised methods, without response variable, in where the main goal is cognition, a better understanding of the target phenomenon, whereas in the right part of the tree, classical

modelling methods are placed, referring those models oriented to re-cognition of a known but probable non-well understood response.

- In a second level of the taxonomical map for non-supervised methods, the kind of relationships targeted is used for the split. Hidden concepts behind the instances are targeted (i.e. to find relationships among the rows of data matrix), this points to methods that we labelled as descriptive methods. If the interest is focused on describing relationships between variables (columns of data matrix), this points to what we called associative methods.

  For supervised methods, the second level of the taxonomy regards the nature of the response variable: predictive methods for numerical response variables while discriminant methods apply for qualitative response variables (fig. 1). An additional split of discriminant models among case-based models, rule-based models and Bayesian models is related with more specific characteristics of methods.

The proposed map is not exhaustive, but it contains the more popular DM methods used in most real applications, as shown in [17]. For each of the four branches of the map (descriptive, associative, discriminant and predictive) there are several boxes with the names of several families of alternative DM methods, coming from different disciplines which are suitable for certain types of problems. Different colours have been used for methods coming from the field of Artificial Intelligence (yellow boxes) or from Statistics (green boxes) or hybrid approaches (blue boxes). The map helps to identify a class of methods on the basis of problem goals and data structure.

4 Method Properties Knowledge Base

Once a class of methods is identified, additional ontological knowledge is required to decide which of the possible alternatives in the map brach is more appropriate for the given problem. Then, theoretical properties of each Data Mining method must be transferred to the system to make the more specific decision. Although the present proposal is designed in such a way that any Methods properties Knowledge Base can be used, provided that it describes all the methods included in the conceptual map, in [14,15] a deep analysis was conducted to identify a set of relevant characteristics to be taken into account for recommendations on correct usage of DM methods. In addition, the following characteristics where considered and an ontological KB was built describing the 31 data mining methods of the DMMCM:

- Type of response variable accepted by the method (Num, Qualitative, Both, None)
- The method accepts numerical explanatory variables (YES/NO)
- The method accepts ordered qualitative explanatory variables (YES/NO)
- The method accepts non-ordered qualitative explanatory variables (YES/NO)
- Suitable data size (small data set, medium size data set, large data set)
- Attribute independency required (Yes: explanatory variables must be independent, No: independence is not assumed by the method)
- Speed of method (fast, medium, slow)
- Preferred interpretability of results (high, medium, low)

The recommendations were based on two steps:

1) Determining the main branch of the DMMCM suitable for the target problem
2) Determining the specific methods' family of a given branch by comparing the characteristics of the method described in the Method Properties Knowledge Base with those hold in data.

A set of suitable methods was proposed to the user.

Thus, given a conceptual map of methods (like DMMCM) and a corresponding Methods' Properties KB, the system was able to reason about the most suitable data mining tasks. Having a parametric pair, conceptual map/Method's Properties KB, the impracticable requirement (basic in classical expert systems) of building an exhaustive Method Properties Knowledge Base on DM techniques can be avoided. Also, the implicit hard assumption on the stability of this prior knowledge can be relaxed, which seems to be a better approach in current context, where the number of data mining methods available grows incredibly fast every day, and the Method's KB is constrained to continuous reviews and upgrades. Incompleteness of the map is not a limitation of the presented system, provided that the Method Properties KB might be updated as an external data file by adding new methods and new properties to be considered as soon as new methods appear, or more specific characteristics evaluated like normality or homoscedasticity, or balanced groups, etc. are to be taken into account.

The system is able to propose a subset of suitable alternatives to recommend.

5 Introduction of reinforcement policy

Soon, it was realized that different kind of characteristics play different roles for the recommendations. Some methods involve technical assumptions that must be mandatorily hold in data to guarantee validity. Others have some properties that guarantee better performance under certain scenarios. While a mismatch in the former implies incorrect use of DM method and produces invalid results, a mismatch in the later only implies worse performance. This suggests that the output of the recommendation should be a ranking of methods rather than a simple list of them. And a relevance measure must be computed to build this ranking.

This means, for instance, that algorithms suitable for small data sets will lose relevance if the target dataset is very large and should appear later in the ranking. Similarly, algorithms requiring independent variables will be lowered in the ranking, if the target dataset has high redundant data or high data correlation.

Multi-criteria approaches are available to build global relevance measures that consider all methods properties. However, these approaches usually assume a specific set of criteria to be considered and fixed, and we are proposing an open approach where the set of characteristics contained in the Method's Properties KB may change along time, according to the development of the State of the Art.

To this purpose we introduce the concept of Reinforcement table (RT), which can be defined externally for every characteristic considered in the Method Properties Knowledge Base (see table 1). For each property a two-way table is defined that compares the characteristic of a given method with the one of the target dataset (numeric variables), or the one expressed by the user as preferred (interpretable model).

Table 1: Reinforcement/Penalization tables

Data Size		Observed dataset size			**Interpretability of results**		User Preference		**Running Speed**		User Preference	
		Large	Med.	Small			Yes	No			Yes	No
Data size requirement of method	Large	1	0	-1	Interpretation is provided by the method	Yes	1	0	Average speed of the method	Very Slow	-3	0
	Irrelevant	0	0	0		No	-1	0		Slow	-1	0
	Small	-1	0	1						Fast	1	0

A mismatching in these properties produces a penalization on the case relevance, whereas matching produces reinforcement. The penalty/reinforcement is obtained from the Reinforcement tables specifically designed for each property (see Table 1).
Currently, the final relevance is computed by direct addition of the penalty/reinforcement obtained by all characteristics, but other global measurements can be considered and different weights might be used to the different characteristics.
This approach is highly flexible to prior knowledge updates of the system, provided that both Method Properties KB and reinforcement tables become codified in external files instead of internally hardcoded, and the entries of each reinforcement table coincides with the values of the corresponding column of the Method Properties Knowledge Base. The idea is that the system is configured by data mining experts with a conceptual map, a Method's Properties KB and their corresponding Reinforcement tables, and this fixed, it can be used by medium trained data mining users to help in real applications (without privileges to change maps, properties nor RTs).

6 Experimental design and results

To assess the behaviour of the proposal and the corresponding recommendation tool, several tests have been done. Three UCI datasets with different structure have been used to figure out some DM scenarios where a recommendation is required.
IRIS Tests: The well-known Iris dataset is composed by four independent variables, all of them continuous, regarding measures of the petals and sepals of iris flowers. The response variable is categorical and represents the three different species under analysis. The data set contains 150 instances, making it a small-sized data set. Three scenarios were designed over Iris data set. In Test1 and Test2 it is supposed that the user wants to recognize the specie of each flower. Whereas in Test1 interpretability of results is priorized (Test1) in (Test2) user is supposedly non interested in interpretability of final results. In (Test3) the user wants to learn flower's associations.
WWTP Tests: The Wastewater Treatment Plant (WWTP) dataset contains 38 different independent variables corresponding to the operation of one activated-sludge wastewater treatment plant, all of which are numeric and continuous. The sample size is 527, we will assume as a medium-sized data set. There is no response variable and

data can only be mined with descriptive purposes. In Test4 the user is interested in discovering relationships among days whereas in Test5 he wants to learn relationships among variables.

Flares Tests: Provides information about solar flares given a set of categorical variables. Sample size is 1389, considered in this experiment as a large data set. The response variable is the number of solar flares observed in a day and it likely follows the law of rare phenomena (Poisson distribution). It takes natural, positive values but has a high asymmetry towards the left hand side. There are actually just a few possible values (i.e., p(Flares > 8) = 0). This particular situation allows a user considering the daily solar flares' tax as quantitative (Test6) or qualitative (Test7).

Table 2: Experimental design with different combination of goals, technical requirements, non-restrictive technical properties and user preferences for all the test cases.

		Goals		**Technical Requirements**		**Non-Restrictive Technical Properties**	**Non-Restrictive Preference Properties**	
Test	**Data Set**	**DM Task**	**Associat.**	**Response Variable**	**Explanatory Variables**	**Data Size**	**Interpretability**	**Speed**
Test1	Iris	Discriminant	NA	Qualitative	Numerical	Small	No Priority	Priority
Test2		Discriminant	NA	Qualitative		Small	Priority	Priority
Test3		Descriptive	No	No response		Small	Priority	No Priority
Test4	WWTP	Associative	Yes	No response	Numerical	Medium	Priority	Priority
Test5		Descriptive	No	No response		Medium	Priority	Priority
Test6	Flares	Predictive	NA	Continuous	Qualitative	Large	Priority	Priority
Test7		Discriminant	NA	Qualitative		Large	No Priority	Priority

In table 2, the experimental design is presented. Seven different scenarios are tested according to different user goals, user preferences and data characteristics.The system is dinamically generating interface questions according to the method's characteristics appearing in the Method's Properties KB to get the user goals and preferences. For each independent scenario, the recommender was used and DM methods priorized, by the relevance obtained by linking the RTs to the Method Properties KB. The ranking of suitable DM methods proposed in each scenario is in table 3.

In both Test1 and Test2 methods located in the Discriminant branch of the DMMCM are proposed. In Test2 Linear Discriminant Analysis (LDA) and Support-Vector Machine (SVM) get lower relevance than in Test1 as interpretability of results becomes important for the user. For the same reason, Case-based Classifier (CBRClas) and Classification and Regression Tree (CART [18]) increase their relevance in Test2 as they are easier models to interpret. In both tests, CBRClas and SVM are ranked later since they are more time consuming and speed was a priority. As iris only contains numerical explanatory variables, none of the rule induction methods (RULES [19], PRISM [20], CN2 [21], RISE [22]) is proposed as a solution. In Test3, clustering

methods appear as more suitable, as no response variable is specified. Methods suitable for small data and giving more interpretable results appear first in the list of recommended methods, according to the iris data size and the declared user preferences.

Table 3: Methods proposed for each test with the reinforcement/penalization values.

Iris						WWTP				Flares			
Test1		Test2		Test3		Test4		Test5		Test6		Test7	
Method	**ΔRel**	**Method**	**ΔRel**	**Method**	**ΔRel**	**Method**	**ΔRel**	**Method**	**ΔRel**	**Method**	**ΔRel**	**Method**	**ΔRel**
LDA	1	CBRClas	1	Kmeans	1	PCA	0	Kmeans	2	CBRPred	3	CBRClas	2
CBRClas	0	LDA	0	KNN	1	BayNet	-2	KNN	2	NeuralNet	-3	BagClas	0
CART	-2	CART	-1	Isodata	0			Isodata	0			NaïveBay	-2
SVM	-2	SVM	-3	BagClus	0			BagClus	-1			RULES	-2
				Cobweb3	-1			Cobweb3	-2			PRISM	-2
												CN2	-2
												RISE	-4
												CART	-4
												ID3	-4
												C4.5	-4

Associative methods or clustering methods are favoured by the recommender depending if the user wants a model for discovering associations between variables (Test4) or between individuals (Test5). In Test4, although interpretability is important for the user, Principal Component Analysis (PCA) has more relevance than Bayesian networks (BayNet) because method speed is also a preference. If the user marks that speed is not a priority, BayNet comes up to the first place in the recommendation list of Test4. The system is able to know, from the ontological knowledge, that neither predictive methods like Artificial Neural Networks (NeuralNet) nor Linear Regression, Time Series nor discriminant methods like SVM are suitable, and prunes all these DM methods from possible solutions. Association rules methods are not recommended because the current implementation is not incorporating them yet. Comparing Test3 vs Test5 shows that defining speed as a priority provides more relevance to KMeans or KNN and penalizes COBWeb3 and Bagging over clustering (BagClus). In Test6 (prediction goal) only CBRPred and NeuralNet methods are proposed because all other predictive methods like ANOVA or Linear Regression require quantitative independent variables, whereas in this case all explanatory variables are qualitative. NeuralNet method is penalized because it is a rather slow and non-easily interpretable method, mismatching user preferences. In Test7 (discrimination goal), a classification task, in which speed is important but interpretability not so much is recommended, as expected. This causes CBRClas and Bagging over Classifiers (BagClas) to appear first while all classical rule inductive methods (RULES, PRISM, CN2) appear in a second layer, indicating not to be very much suitable for large data sets, as it is the case. The worst DM methods are those algorithms highly time-consuming like RISE rule-induction method or decision tree classifiers (CART, ID3 [23], C4.5 [24]). Later DM methods are not favoured as interpretability is not a priority for the user.

5. Conclusions and future work

Choosing the proper data mining method is one of the most critical and difficult tasks in the KDD process. In this paper, an open conceptual frame to use ontological knowledge about Data Mining techniques is proposed to describe the relevant characteristics of the Data Mining methods in flexible input tables (Method Properties Knowledge Base) linked with reinforcement tables that permit comparison among methods' properties and data characteristics, or user goals/preferences in a flexible framework that can be easily updated to developments of the State of the Art.
The ontological knowledge has been introduced through three different formalisms: The DMMCM map, the Method's Properties Knowledge base and the Reinforcement tables. Authors are not aware of any other methodological recommender using this open approach and the ontology of data mining methods is decision-based, as DMMCM is. In the literature, most of the classification of data mining methods follows an organization based on the nature of the methods itself; the discipline from which they come or the kind of formalisms used, but this do not provide much help to the selection of most appropriate methods in real applications. Main decisional criteria used by human experts in real decisions have been taken into account in the organization of methods in the DMMCM. Several expert data miners were involved in the validation of the results. They concluded that rankings provided by the 7 experimental scenarios tested were aligned with their own priorization of methods.
Better results can be obtained with larger Method's Properties KB including more characteristics, like normality requirements or balanced dataset requirements, and the description of more families of methods not yet included in the DMMCM, but preliminary results are already available in this direction. Current work is in progress towards a hybrid approach combining the information provided by the proposed ontological frame with a case base reasoning framework that can complement the current recommendations with past experiences in which specific parameters used to run the recommended methods can be also recommended.

Acknowledgements. Authors thank the contribution of Darío García, Aleix Canals and Alexandra Mansilla in the research work.

References

1. Fayyad, U, Piatetsky-Shapiro, G. and Smyth, P. From Data Mining to Knowledge Discovery: An overview. In *Advances in Knowledge Discovery and Data Mining*, 1-34. AAAI/MIT Press, 1996.
2. http://www.kdnuggets.com/polls/2006/data_mining_methods.htm. Data Mining Methods (2006).
3. Kantardzic, M. *Data Mining: Concepts, Models, Methods, and Algorithms*. 2nd Edition, Wiley-IEEE Press, ISBN: 978-0-470-89045-5, 2011.
4. Choinski, M. and Chudziak, J.A. Ontological Learning Assistant for Knowledge Discovery and Data Mining. *Procc. of IEEE Int. Multiconference on Computer Science and Information Technology*, pp. 147-155. ISBN 978-83-60810-22-4.
5. Charest, M., Delisle, S., Cervantes, O. and Shen, Y. Bridging the gap between data mining and

decision support: A case-based reasoning and ontology approach. *Intelligent Data Analysis* 12(2):211-236, 2008.

6. Charest, M., and Delisle, S. Ontology-guided intelligent data mining assistance: Combining declarative and procedural knowledge. Artificial Intelligence and Soft Computing 2006: 9-14.
7. Recio-García, J.A., Díaz-Agudo, B., González-Calero, P.A. and Sánchez-Ruiz, A.A.. Ontology based CBR with jCOLIBRI, in: Applications and Innovations in Intelligent Systems XIV. *Proceedings of AI-2006, the Twenty-sixth SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*, Springer, 2006.
8. Oprean, C. Towards user assistance in Data Mining. Master's Thesis. Telecom Bretagne, FR, 2011.
9. Chapman P., Clinton J., Kerber R., Khabaza T., Reinartz T., Shearer C., Wirth, R. CRISP-DM 1.0. Step-by-step data mining guide, NCR Systems Engineering Copenhagen (USA and Denmark), DaimlerChrysler AG (Germany), SPSS Inc. (USA) and OHRA Verzekeringen en Bank Groep B.V (The Netherlands), 2001.
10. Moreira Pererira, T.M. DataMining4Dummies: a web application for automatic selection of data mining algorithms for new problems. Master's Thesis. Universidade do Porto, Portugal, 2014.
11. RapidMiner, `http://rapidminer.com`. January 2015.
12. OpenML, `http://openml.org`. January 2015.
13. Serban F., Vanschoren, J. Kietz, J.-U. and Bernstein, A. A survey of Intelligent Assistants for Data Analysis. *ACM Computing Surveys*, Vol.45, No. 3, Article 31, June 2013.
14. Gibert, K., Sànchez-Marrè, M. and Codina, V. Elección de la técnica de minería de datos: Mapa conceptual de técnicas (In Spanish). *Actas del V Taller de Minería de Datos y Aprendizaje (TAMIDA'10), in CEDI 2010 (Ed. A. Troncoso y J.C. Riquelme),* IBERGARCETA PUBLICACIONES, S.L., Madrid, pp. 37-43, ISBN: 978-84-92812-60-8, September 2010.
15. Gibert, K., Sànchez-Marrè, M. and Codina, V. Choosing the Right Data Mining Technique: Classification of Methods and Intelligent Recommendation. *Proc. of 5th International Environmental Modelling and Software Society Conference (iEMSs'2010)*. ISBN 978-88-903574-1-1. Ottawa, Canada. July 2010.
16. Sànchez-Marrè, M., Gibert, K. and Sevilla, B. Evolving GESCONDA to an Intelligent Decision Support Tool. *Proc. of 5th International Congress on Environmental Modelling and Software (iEMSs'2010)*, Vol. 3, pp. 2015-2024. ISBN 978-88-903574-1-1. Ottawa, Canada. July 2010.
17. Bache, K. and Lichman, M. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science, 2013.
18. Breiman, L., Friedman, J.H., Olshen R.A. and Stone, C.J. *Classification and Regression Trees*, Wadsworth, Belmont, 1984.
19. Pham, D.T and Aksoy, M.S. RULES: a simple ruler extraction system. *Expert Systems with Applications* 8(1):59-65, 1995.
20. Cendrowska, J. PRISM: an algorithm for inducing module rules. *Int'l Journal of Man-Machine Studies* 27(4):349-370, 1987.
21. Clark, P. and Niblett, T. The CN2 induction algorithm. *Machine Learning* 3:261-283, 1989.
22. Domingos. Unifying Instance-Based and Rule-Based Induction, *ML* 24(2):141-168, 1996.
23. Quinlan J.R. Induction of decision trees, *Machine Learning* 1(1):81- 106, 1986.
24. Quinlan J.R. C4.5: Programs for Machine Learning, Morgan Kaufmann, CA, USA, 1993
25. R. Vilalta, C. Giraud-Carrier, P. Brazdil, and C. Soares, Using Meta-Learning to Support Data Mining. Journal of Computer Science Applications, 2004, 31-45
26. C. Castliello, G. Castellano, and A. Fanelli, Meta-Data: Characterization of Input Features for Meta-Learning. Modeling Decisions for Artificial Intelligence, LNAI, 2005, 457-468
27. Peng, Y., Flach, P., Soares, C., Brazdil, P.: Improved dataset characterisation for meta-learning. LNCS, vol. 2534, pp. 193:208. Springer Berlin / Heidelberg (2002).