Meta-learning with kernels and similarity functions for planning of data mining workflows

Alexandros Kalousis KALOUSIS@CUI.UNIGE.CH
Abraham Bernstein BERNSTEIN@IFI.UZH.CH
Melanie Hilario HILARIO@CUI.UNIGE.CH

Keywords: planning, meta-learning, kernels, similarities, relational learning

Abstract

We propose an intelligent data mining (DM) assistant that will combine planning and meta-learning to provide support to users of a virtual DM laboratory. A knowledge-driven planner will rely on a data mining ontology to plan the knowledge discovery workflow and determine the set of valid operators for each step of this workflow. A probabilistic meta-learner will select the most appropriate operators by using relational similarity measures and kernel functions over records of past sessions meta-data stored in a DM experiments repository.

1. Introduction

We propose an architecture that combines a planning-based and a meta-learning approach in providing data mining support to end users. By adding AI-planning to meta-learning, we can ensure support for the complete knowledge discovery process. Contrary to previous efforts where the dominant focus was on either learning (Statlog, Metal) or preprocessing (MiningMart), our data mining assistant will propose workflows that start with the raw data, select and sequence the different preprocessing operations, select a suitable learning algorithm and output trained models. On the other hand, by adding meta-learning to planning-based data mining (DM) support, it will make the planner adaptive to changes in the data and capable of improving its advice over time; this improvement will apply to the planner's decision-making at any node of the knowledge discovery workflow.

Meta-learning will be based on multiple and diverse types of meta-data. Statlog and Metal meta-learners relied mainly on quantitative (e.g., statistical, information-theoretic) characteristics of data to select appropriate learning algorithms (Michie et al., 1994; Metal, 2002). MiningMart described datasets in terms of domain concepts but did not use these to metalearn (Morik & Scholz, 2004). The proposed system will meta-mine both quantitative and qualitative, domain ontology based metadescriptions of the application dataset. In addition, the meta-learner's ken will, for the first time, go beyond the dataset to take into account a significantly extended learning context the application task, performance criteria, workflow quality indicators, and the user's profile as defined by quantitative results and qualitative feedback from his past historical record of data mining experiments.

Generalizing from these heterogeneous factors requires defining similarity measures and data mining operators over complex structures. We will explore elaborate task descriptors such as operator trees (Mierswa et al., 2006) or multirelational experiment descriptors that integrate information concerning datasets, algorithms, and evaluation strategies, (Kalousis & Hilario, 2003; Hilario & Kalousis, 2001).

We propose a novel meta-learning technique which blends probabilistic reasoning and kernel-based learning from complex structures. We will exploit a framework that we have recently developed for kernel-based learning over complex structures using the language of relational algebra (Woznica et al., 2007; Woznica et al., 2005). To meta-learn from the diverse factors described above, we will weave state transition probabilities into kernel-based learning over relational schemas and devise methods for adjusting these probabilities to improve the data mining assistant's choices as it gains experience.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

2. An Intelligent Assistant for Data Mining

The intelligent data mining assistant will be at the helm of a virtual DM laboratory designed for a community of users with common data-analytical needs in a specific application domain. Essential components of this e-lab will be a DM ontology and a DM experiments repository. The ontology will provide a formal specification of the knowledge discovery process – its different phases, the set of operators that can be legitimately applied at each phase, and so forth. The repository will be the e-lab's long-term memory; detailed records of all experiments performed in the e-lab will be stored in the repository to allow for replication and comparative meta-analysis of data mining experiments.

The DM assistant will take in user specifications of the knowledge discovery task and available data, plan a methodologically correct learning process, and suggest ranked workflows that the user can enact to achieve the pre-specified objectives. To plan the workflow and determine the operator or algorithm to apply for a given data mining step, the assistant will harness prior knowledge stored in the DM ontology. Metadata stored in the DM experiments repository will be leveraged to improve the data mining process itself, for instance by incrementally refining the DM planner's search in the design space of candidate DM operators (and workflows). The kernel-based, probabilistic meta-learner will dynamically adjust transition probabilities between DM operators, conditioned on the current application task and data, user-specified performance criteria, quality scores of workflows applied in the past to similar tasks and data, and the user's profile (based on quantified results from, and qualitative feedback on, her past DM experiments). The proposed meta-learning method will be evaluated against the baseline of a case-based DM planner, which retrieves and adapts workflows from the most similar past experiments. By comparing the DM planner's evolution over time based on these two approaches, we hope to gain insights into the patterns that govern the efficacy of data mining workflows, operators and parameters.

2.1. The Knowledge Driven Planner

A DM ontology and repository will ease the task of constructing a complicated KD process by simplifying scientists' access to the plethora of data mining concepts, algorithms, data sources, and past experiments.

We will devise a tool that helps data miners (and data mining scientists) to navigate the space of KD processes systematically, and more effectively. In particular, we will develop an intelligent discovery assistant (IDA) that helps a data miner with the exploration of the space of valid[1] DM processes (Bernstein & Provost, 2005). The discovery assistant's intelligence comes mainly from its awareness of the full knowledge discovery context and its capacity to learn incrementally from experience. The KD context is available to the IDA in the form of the user's task specification and domain-ontology based semantic annotations on the dataset. In addition, the IDA can extract quantitative characteristics of the dataset such as the number of explanatory variables or the percentage of missing values. The IDA uses this contextual information, together with knowledge from the DM ontology and knowledge base (e.g., applicability conditions of DM operators), to search for and enumerate the valid and effective DM processes. It does this by (i) retrieving and adapting them from the DM experiments repository using case-based reasoning approaches, or (ii) using AI planning type approaches to construct new valid data mining processes.

Once the IDA has listed a variety of alternatives it also assists the user in choosing workflows to execute, for example, by ranking the workflows (heuristically) according to what is important to the user. In addition, the IDA will also allow for some open-ended, statistical/exploratory data analysis, as has been addressed by Amant and Cohen (1998). In such explorations, the IDA does not necessarily provide the user with finished KD-workflows, but provides guidance at each step in the exploration of a KD-process - a type of support that is suitable in data mining endeavors that are exploratory and/or where the case-based/planning based IDA does not provide satisfactory KD-workflows. In this exploratory mode, scientists (or data miners) would first assemble underlying data-sources after which the IDA would try to provide advise on what possible next steps could be. As soon as scientists would choose one of these steps the IDA would execute it in the background and try to advise on next steps or suggest backtracking to the previous decisions if some newly arisen information would warrant this.

To allow the planner-based IDA to improve with experience, we introduce a set of probabilistic parameters that will be automatically adjusted by meta-

[1] A *valid* DM workflow violates no fundamental constraints of its constituent techniques. An automated system can take advantage of an explicit ontology of data-mining techniques, which defines the various techniques and their properties.

mining the DM experiments repository. The DM-workflow planner is essentially a breadth-first search algorithm that starts from an initial state and tries to reach a final stage by sequencing data mining operators. At each state the search algorithm will add all DM operators that can be legitimately applied. Prior knowledge about operator application constraints is obtained from the DM ontology and modeled in a state transition table $\mathbf{T}$ with dimensionality $K \times K$ (K is the number of DM operators). The T_{ij} element of this table is defined as $T_{ij} = P(O_j|O_i, \mathbf{D}, \mathbf{KDT})$. In words, it denotes the transition probability from state (DM operator) i, to state (DM operator) j, given the description $\mathbf{D}$ of the data and the description of the knowledge discovery task, $\mathbf{KDT}$. These probabilities sum to one over a given row i, $\sum_j T_{ij}$. In the simple breadth-first search all valid transitions are equiprobable (will all be expanded and explored), since there are no preferred sequences of operators.

The planner should establish the sequence of data mining operators $WF = [S_1, S_2, ..., S_N]$ with maximal joint probability distribution given the data description D, and the knowledge discovery task description, that is: $WF = argmax_{WF} P(S_1, S_2, ..., S_N|\mathbf{D}, \mathbf{KDT})$. Under the assumption that the transition to the next stage depends only on the current stage, the data description, $\mathbf{D}$, and the knowledge discovery task, $\mathbf{KDT}$, the joint probability distribution $P(S_1, S_2, ..., S_N|\mathbf{D}, \mathbf{KDT})$ factorizes as: $P(S_1|\mathbf{D}, \mathbf{KDT}) \prod_{i=2}^{N} T_{S_{(i-1)}S_i}$ The initial stage S_1 is governed by a probability distribution defined over the different data mining operators, given the data $\mathbf{D}$ and the knowledge discovery task $\mathbf{KDT}$, i.e. $S_1 \sim P(\mathbf{O}|\mathbf{D}, \mathbf{KDT}) = [P(O_i|\mathbf{D}, \mathbf{KDT})|i = 1 \ldots N]$. It is straightforward to adapt the search algorithm to output together with the DM-workflows their joint probabilities. Assuming $\mathbf{T}$ is in its original state, i.e. ignoring all past experiments in the DM experiments repository, all DM workflows of equal length will also have equal joint probability, as the breadth search algorithm examines all states without distinction. Note here that longer workflows will have lower joint probability than shorter since their joint probability is a product of a larger number of terms, one issue that arises here is whether this should be factored out by an appropriate normalization, nevertheless intuitevely one would prefer shorter and simpler workflows over more complex.

We have expressed the way in which the search algorithm moves around the space of states, i.e. data mining operators, in terms of state transition probabilities, i.e the set of parameters $\mathbf{T}$, $P(\mathbf{O}|D, KDT)$. It is this set of parameters that will be the target of the incremental meta-learning.

2.2. Meta-Learning

The general problem that IDA tries to solve can be formulated as follows: given a user, U, of the platform facing a knowledge discovery problem, $\mathbf{A}$, and a description of $\mathbf{A} = (\mathbf{D}, \mathbf{KDT})$, where, $\mathbf{D}$ corresponds to a description of the data, both in terms of their semantics, as these are established after their annotation with respect to the domain ontology, and their quantitative characteristics, $\mathbf{KDT}$ is a description of the Knowledge Discovery Task that the user is trying to accomplish, establish a data mining workflow $\mathbf{WF}$ which will address the knowledge discovery task and will optimize some performance criteria, $\mathbf{PC}$, specific to the user. The description of $\mathbf{KDT}$ can be as high level as simply stating the learning task to be performed, e.g. classification, or more specific such as stating that the goal is classification using a reduced set of features. Its description will be given in the form of a workflow although at a high level. At the beginning IDA will rely solely on the data mining ontology and the planner to propose and rank a number of alternative WFs; essential to the planning process is the set of parameters, $\mathbf{T}$, $P(\mathbf{O}|D, \mathbf{KDT})$. The initial state of these parameters will be determined by the data mining experts. To allow the planner to recommend the most appropriate workflow(s), we propose an infrastructure to adapt these parameters to the requirements of a given knowledge discovery problem as these can be gathered from $(\mathbf{A}, U, \mathbf{PC})$. The adaptation process will take account of previous data mining experiments and performance results, as well as other factors such as users' feedback, context and reputation.

More precisely, the system is confronted with a number of data mining experiments performed by various users, which are eventually stored in the DMER. Each data mining experiment, $\mathbf{DME_k}$, is a complex structure described by a number of components that will eventually resemble to something like: $\mathbf{DME_k} = (U_k, \mathbf{WF_k}, \mathbf{D_k}, \mathbf{KDT_k}, \mathbf{PC_k}, \mathbf{UF_k})$. U_k is the identifier of the user that performed the given experiment, the remaining variables have a complex structure. $\mathbf{WF_k}$ denotes the workflow that was applied on the given $\mathbf{DME_k}$, and is actually a sequence of data mining operators; $\mathbf{D_k}$ is the description of the data analysed in the experiment$\mathbf{DME_k}$; $\mathbf{KDT_K}$ is the description of the knowledge discovery task that is to be performed; $\mathbf{PC_k}$ is a vector containing different performance measurements obtained by applying workflow WF_k to dataset $\mathbf{D_k}$, with respect to a number of performance criteria; and $\mathbf{UF_k}$ is some qualitative user feeback along a number of different dimensions, such as understandability, ease of use, complexity etc.

Learning the planner's parameters for a new Knowledge Discovery Problem The meta-learning module will establish functions $f_{\mathbf{T}}(\mathbf{A})$ and $f_{\mathbf{O}}(\mathbf{A})$ that, given the description, $\mathbf{A} = (\mathbf{D}, \mathbf{KDT})$, of a new, potentially unseen, knowledge discovery problem, will estimate $\mathbf{T}$ and $P(\mathbf{O}|\mathbf{A})$, respectively. These estimates will then be used by the planner to provide a ranked list of workflows for $\mathbf{A}$. The main learning paradigm that we will use is that of kernel-based estimation. Let $\mathbf{X_k} = (\mathbf{D_k}, \mathbf{KDT_k})$ denote the description of the knowledge discovery problem associated with data mining experiment $\mathbf{DME_k}$, then:

$$f_{\mathbf{T}}(\mathbf{A}) = \frac{\sum_{\mathbf{X_k}} \overline{\mathbf{T_k}} K_{KDP}(\mathbf{A}, \mathbf{X_k})}{\sum_{\mathbf{X_k}} K_{KDP}(\mathbf{A}, \mathbf{X_k})}, \qquad (1)$$

$$f_{\mathbf{O}}(\mathbf{A}) = \frac{\sum_{\mathbf{X_k}} \overline{\mathbf{P(O|X_k)}} K_{KDP}(\mathbf{A}, \mathbf{X_k})}{\sum_{\mathbf{X_k}} K_{KDP}(\mathbf{A}, \mathbf{X_k})}$$

where the summations are taken over all $\mathbf{X_k} \in DMER$. $K_{KDP}(\mathbf{A}, \mathbf{X_k})$ is a kernel[2] function that provides a measure of similarity of $\mathbf{A}$ and $\mathbf{X_k}$; $\overline{\mathbf{T_k}}$ and $\overline{P(\mathbf{O}|\mathbf{X_k})}$ are estimations of $\mathbf{T}$ and $P(\mathbf{O})$ derived from $\mathbf{DME_k}$. The workflow, $\mathbf{WF_k}$, of a data mining experiment, $\mathbf{DME_k}$, can give rise to an estimation, $\overline{\mathbf{T_k}}$, of $\mathbf{T}$ simply by counting the times that a transition happens from one operator, O_i, to another operator, O_j, within that workflow and normalizing it by the total number of transitions. Similarly it can provide us with an estimation of $\overline{P(\mathbf{O}|\mathbf{X_k})}$. In both estimates we can imagine adding a Laplace correction so that the probability of the transitions that do not appear is greater than zero.

Kernels on Descriptions of Knowledge Discovery Problems The estimates given in equations 1 can be readily used by the planner to construct workflows which are tailored to $\mathbf{A}$. We will design kernel functions which are appropriate for this type of problem exploiting similarity measures defined over datasets in the context of meta-learning, (Kalousis & Hilario, 2003), but also more general kernel functions for complex objects, (Woznica et al., 2007; Woznica et al., 2005) Since the description of a knowledge discovery problem consists of two quite different parts, the data description and the knowledge discovery task description, we envisage that the $K_{KDP}(\mathbf{A}, \mathbf{X})$ kernel will be the composition of two very different kernels, one, K_D, defined on the data part of the description and another, K_{WF}, defined on the knowledge discovery task description part. The first kernel will have to account for similarities defined not only with respect to quantifiable characteristics of the datasets, but also with respect to their annotations within the domain ontology. The second kernel will be defined over the language used to described knowledge discovery tasks, potentially over different abstraction levels, and will exploit similarities of operators and concepts derived from the data mining ontology. Note here that the definition and availability of these kernel—similarity—functions will also serve the needs of the case-base and will result in similarity measures for the retrieval of similar knowledge discovery problems, datasets, and workflows—knowledge discovery tasks. We will design, test, and evaluate different ways of defining the K_{KDP} kernel, exploring and/or even learning the importance of its constituents, (Woznica et al., 2007). We will also explore different estimations of $\overline{\mathbf{T_k}}$ and $\overline{P(\mathbf{O}|\mathbf{X_k})}$ from a given workflow description $\mathbf{WF_k}$.

Accounting for Qualitative and Quantitative Performance Indicators of WFs The estimations derived from the equations of 1 are based only on the similarity of the description of the current knowledge discovery problem $\mathbf{A}$ with the descriptions of the knowledge discovery problems $\mathbf{X_k}$, thus ignoring any quality indicator for the $\mathbf{WF_k}$ workflow associated with $\mathbf{X_k}$. The quality indicators come into two flavors: quantitive performance measures, contained in $\mathbf{PC_k}$, that are estimated from the actual application of the $\mathbf{WF_k}$ workflow on the data, and qualitative indicators, contained in $\mathbf{UF_k}$, that are given by the user U_k, concerning non-easily quantifiable dimensions such as understandability and/or simplicity of the final models produced by the workflow, ease of use of the workflow etc. Yet a third, indirect, quality indicator of a workflow $\mathbf{WF_k}$ can come from the "quality" of the user U_k associated with the workflow. The quality of a user U_k will be given by a function $Q(U_k)$ that will account for various factors, such as how often workflows designed by this user have been adopted by other users, how the workflows of this user have been qualified by other users, how this user has been qualified by other users, etc. By accounting for such quality indicators of a $\mathbf{WF_k}$ workflow we can accordingly favor or penalise the estimations $\overline{\mathbf{T_k}}$ and $\overline{P(\mathbf{O}|\mathbf{X_k})}$ derived from $\mathbf{WF_k}$.

Moreover we should account for the fact that different users might have different preferences concerning the desired quantitative and qualitative performance indicators of the workflows, e.g. trading accuracy for

[2] A kernel function, $k(x, y)$, provides the similarity of the images of x and y in some feature space without having to compute explicitly the mapping.

understandability. In order to address such differences we will design user dependent parameterizable functions $f_u(\mathbf{PC_k}, \mathbf{UF_k})$ of the quality indicators. These functions will weight heavily the $\overline{\mathbf{T_k}}$ and $\overline{P(\mathbf{O}|\mathbf{X_k})}$ estimations derived from workflows that exhibit the desired performance while they will reduce towards zero the weights of estimations derived from workflows with poor performance. Incorporating these functions into equations of 1 results in estimates of the form:

$$f_{\mathbf{T}}(\mathbf{A}) = \frac{\sum_{\mathbf{X_k}} \overline{\mathbf{T_k}} K_{KDT}(\mathbf{A}, \mathbf{X_k}) f_u(\mathbf{PC_k}, \mathbf{UF_k}) Q(U_k)}{\sum_{\mathbf{X_k}} K_{KDT}(\mathbf{A}, \mathbf{X_k}) f_u(\mathbf{PC_k}, \mathbf{UF_k}) Q(U_k)} \tag{2}$$

$$f_{\mathbf{O}}(\mathbf{A}) = \frac{\sum_{\mathbf{X_k}} \overline{\mathbf{P}(\mathbf{O}|\mathbf{X_k})} K_{KDT}(\mathbf{A}, \mathbf{X_k}) f_u(\mathbf{PC_k}, \mathbf{UF_k}) Q(U_k)}{\sum_{\mathbf{X_k}} K_{KDT}(\mathbf{A}, \mathbf{X_k}) f_u(\mathbf{PC_k}, \mathbf{UF_k}) Q(U_k)}$$

We will thus design, test and evaluate, performance aware estimates of the parameters of the planner, according to the equations of 2 by incorporating user dependent functions of qualitative and quantitative performance indicators of the workflows as well as user quality indicators through the $Q(U_k)$ function. The latter will draw heavily on the definition of authority indexes described later.

User's Profile—Context The profile of a user consists of the information stored about the user within the system. This information consists of all the previous knowledge discovery projects that he/she has undertaken, the data mining experiments that he performed within each knowledge discovery project, the datasets associated with these experiments, the descriptions of these datasets, the workflows that he/she has chosen for final deployment or publication on the platform, the feedback that he/she has provided on previous suggestions of the system, the workflows he/she has designed from scratch without relying on the system's support, his/her level of authority, as this is determined by the frequency of use by other users of the workflows he/she has published. We will define precisely the different information sources that collectively constitute a user's profile. An important part of this task will be the definition of authority indexes for the users of the system. We will construct kernel functions, $K_U(U_k, U_l)$, to measure the similarities of the profiles—contexts— of any pair of users, U_k, U_l. Since the profile of a user consists of datasets, knowledge discovery task descriptions, and more, the K_U kernel will be actually a set kernel based on agreegations of K_{KDP} kernels on the different problems that the user has encountered, potentially including other kernels defined on other aspects of a user's profile.

Incorporating User's Feedback and Context So far the estimations of the parameters given by $f_{\mathbf{T}}(\mathbf{A})$ and $f_{\mathbf{O}}(\mathbf{A})$ were adapted to the descriptions of the knowledge discovery problem that should be solved, accounting for the quality of the previous solutions, but they do not incorporate any existing feedback from the user that is performing the current data mining experiments on previous analysis episodes and workflows within there, nor any information about his/her profile. In order to do that we will incorporate the $K_U(U_k, U_l)$ kernel in the computation of $f_{\mathbf{T}}(\mathbf{A})$. Like that the qualitative and quantitative indicators given by the user U, who is performing the actual experiment, in his/her past interactions with the system, will be given maximum weight since $K_U(U, U)$ atains the maximum possible similarity value. Moreover the incorporation of $K_U(U, U_k)$ will assign greater importance to users that exist in similar contexts as the U, thus modeling the assumption that users in similar context will probably find interesting similar tools. The new estimations will be functions of both the description of the knowledge discovery application problem, $\mathbf{A}$, and the user, U, who is faced with $\mathbf{A}$.

$$f_{\mathbf{T}}(\mathbf{A}, U) = \frac{\sum_k \overline{\mathbf{T_k}} K_{KDT}(\mathbf{A}, \mathbf{X_k}) f_u(\mathbf{PC_k}) f_u(\mathbf{UF_k}) Q(U_k) K(U, U_k)}{\sum_k K_{KDT}(\mathbf{A}, \mathbf{X_k}) f_u(\mathbf{PC_k}) f_u(\mathbf{UF_k}) Q(U_k) K_U(U, U_k)} \tag{3}$$

$$f_{\mathbf{O}}(\mathbf{A}, U) = \frac{\sum_k \overline{\mathbf{P}(\mathbf{O}|\mathbf{X_k})} K_{KDT}(\mathbf{A}, \mathbf{X_k}) f_u(\mathbf{PC_k}) f_u(\mathbf{UF_k}) Q(U_k) K(U, U_k)}{\sum_k K_{KDT}(\mathbf{A}, \mathbf{X_k}) f_u(\mathbf{PC_k}) f_u(\mathbf{UF_k}) Q(U_k) K_U(U, U_k)}$$

We will define, test, and evaluate, the final form of the adaptive estimations of the parameters of the planner which will incorporate the user's feedback on previous analysis episodes and suggestions of the system, as well as information about the user's context similarity with that of other users. In the latter the idea is that users that exist in similar contexts will have similar data analysis needs.

3. Evaluation

We will systematically evaluate the different strategies for estimating the parameters of the DM-workflow planner. The basic evaluation strategy will be to examine how well the suggestions of the planner, under the different estimation strategies, correlate with the users' actual feedback. Standard hold-out or resampling-based strategies will be used to estimate this correlation. The key idea will be to use a part of the available data to build the estimates for unseen cases and compute the correlation with the user feedback. Different levels of evaluation will be of interest, namely, evaluating performance on completely new users for which nothing is known, and evaluating performance for users that have a recorded history.

4. Discussion

In this paper we propose a system that will combine planning and meta-learning to provide support to users of a virtual laboratory. Standard planning approaches return a number of different solutions, typically unranked. Our planner will rank these solutions according not only to their probabilities among different users and different user communities, as these are depicted in the state transition matrix, but also with respect to a number of qualitative and quantitative performance indicators on past problems. Equally important we account for the different degrees of relevance that these performance indicators might have for different users, or even for the same user in different contexts, by incorporating as a part of the establishement of the final ranking of plans parameterizable, according to user preferences, functions of these quality indicators.

A crucial factor for the success of the system, especially if one considers the enormous size of hypothesis space of the metalearning problem, is the construction of a large repository of Data Mining Experiments. In order to address this issue we plan to exploit ideas from social networking coupled with e-science platforms. One such platform is MyExperiment, (Goble & De Roure, 2007), which is an e-science social network that supports the exchange of complex workflows that address bioinformatics problems. Exploiting the idea of social networks for the construction of a network of Data Mining Scientists provides a very promising way to address the problem of data collection for metalearning. Such social networks already exist in the form of forums build around specific data analysis tools such as Weka (Witten & Frank, 2005), RapidMiner (Mierswa et al., 2006). Moving to the next stage where participants will exhange not only comments and suggestions but in fact complete data mining workflows, such as Weka or RapidMiner workflows, that can be readily applied is not such a great leap. We believe that the benefits for the data analysis community would be great and analysts will have every reason to participate to such a community by contributing content benefiting from the collective intelligence.

References

Amant, S., & Cohen, P. (1998). Intelligent support for exploratory data analysis. *Journal of Computational and Graphical Statistics*, *7*, 545–558.

Bernstein, A., & Provost, F. andHill, S. (2005). Towards intelligent assistance for a data mining process: An ontology-based approach for cost-sensitive classification. *IEEE Transactions on Knowledge and Data Engineering*, *17*.

Goble, C., & De Roure, D. (2007). myexperiment: social networking for workflow-using e-scientists. *In Proceedings of the 2nd workshop on Workflows in support of large-scale science*.

Hilario, M., & Kalousis, A. (2001). Fusion of meta-knowledge and meta-data for case-based model selection. *In Proceedings of the Fifth European Conference on Principles and Practice of Knowledge Discovery in Databases* (pp. 180–191).

Kalousis, A., & Hilario, M. (2003). Representational issues in meta-learning. *Proc. 20th International Conference on Machine Learning (ICML-2003)* (pp. 313–320). Morgan Kaufmann.

Metal (2002). Metal related bibliography. http://www.ofai.at/research/impml/metal/metal-publications.html.

Michie, D., Spiegelhalter, D. J., & Taylor, C. C. (Eds.). (1994). *Machine learning, neural and statistical classification*. Ellis-Horwood.

Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M., & Euler, T. (2006). YALE: Rapid prototyping for complex data mining tasks. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2006)*.

Morik, K., & Scholz, M. (2004). *Intelligent technologies for information analysis*, chapter The MiningMart Approach to Knowledge Discovery in Databases. Springer.

Witten, I. H., & Frank, E. (2005). *Data mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Woznica, A., Kalousis, A., & Hilario, M. (2005). Kernels over relational algebra structures. *Proceedings of the Ninth Pasific Asia Conference on Knowledge Discovery, PAKDD*. Springer.

Woznica, A., Kalousis, A., & Hilario, M. (2007). Learning to combine distances for complex representations. *Proceedings of the 24th International Conference on Machine Learning*.