Collaborative data mining needs centralised model evaluation

Hendrik Blockeel HENDRIK.BLOCKEEL@CS.KULEUVEN.AC.BE
Katholieke Universiteit Leuven, Department of Computer Science, Celestijnenlaan 200A, B-3001 Leuven, Belgium

Steve Moyle STEVE.MOYLE@COMLAB.OX.AC.UK
Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford, OX1 3QD, UK

Abstract

Collaborative data mining refers to a data mining setting where different groups are geographically dispersed but work together on the same problem in a collaborative way. Such a setting requires adequate software support in order to be efficient. In this paper we describe lessons learnt from an experiment with a simple implementation of such a collaborative data mining environment. These lessons relate not to individual data mining approaches but to problems that arise specifically from the collaborative setting. One of these, concerning evaluation of models, is discussed in more detail and a number of possible solutions are proposed. This discussion can contribute to a better understanding of how collaborative data mining is best organized.

1. Introduction

Many different approaches to data mining exist. Some are based on statistical techniques, some on machine learning, etc. They have arisen from different communities (databases, statistics, machine learning, ...). Thus, data mining nowadays is performed by people with highly different backgrounds, each of whom have their preferred techniques. Very few people are experts in all these domains, so to get the most out of a data mining process, ideally one would make use of multiple experts, so that their combined expertise covers all of these domains. These different experts should work together on the same knowledge discovery task. Under the assumption that even experts in a single of these different domains may be relatively rare, such a group of experts may not be available in a single location.

These observations provide motivation for the development of a methodology for *collaborative data mining*. Our point of departure is that groups with different expertise who are geographically distributed should be able to collaborate on a certain problem, thus jointly achieving better results than any of them could individually.

Having different experts collaborate on the same task requires some supporting environment. In the context of the European SolEuNet project, ideas have evolved about what functionality such an environment should have, resulting in a proposal for a collaborative data mining methodology and supporting system called RAMSYS (Jorge et al., 2002). A first implementation of RAMSYS was made using a groupware system called Zeno (Gordon et al., 2001).

In this paper we report on a collaborative data mining experiment in which the proposed RAMSYS methodology and its implementation on Zeno were used. Several lessons have been learnt from this experiment regarding the methodology itself as well as its current implementation. We briefly mention our main conclusions from the experiment, and then discuss in further detail one of the shortcomings we have observed in the current system. This shortcoming relates to model evaluation. We propose multiple possible solutions for this shortcoming, and indicate one which represents an easy-to-implement and concrete improvement to the RAMSYS methodology.

The remainder of this paper is structured as follows. In Section 2 we discuss the RAMSYS and Zeno systems. In Section 3 we describe the data mining problem that was considered in our collaborative data mining experiment. In Section 4 we mention the main problems encountered when trying to apply our collaborative data mining methodology on this case, discussing in detail the problem of model evaluation. In Section 5 we propose a number of possible solutions for the model evaluation problem, discuss their advantages and disadvantages, and make a concrete proposal for improving the RAMSYS methodology. Finally, Section 6 concludes.

2. Collaborative Data Mining, RAMSYS and Zeno

Data mining is about solving problems by analysing data already present in databases (Witten & Frank, 1999). Problem solving, in general, can be codified and a procedure or methodology can be devised. For data mining, just such a methodology has been devised[1] – the CRoss Industrial Standard Process for Data Mining, CRISP-DM (Chapman et al., 2000). CRISP-DM reduces the data mining problem into the six interrelated phases of 1) *Business Understanding*; 2) *Data Understanding*; 3) *Data Preparation*; 4) *Modelling*; 5) *Evaluation*; and 6) *Deployment*. These phases, although presented in a linear manner, have many cycles and feedback loops connecting the phases. Often, effort expended in one phase highlights the need for further work in a prior, previously considered complete, phase.

The *RAMSYS* methodology is an extension to the CRISP-DM methodology for distributed teams who collaborate in a data mining project. The aim is to combine the great range of expertise available in the data miners to effect more valuable solutions to the data mining problem. The RAMSYS methodology attempts to achieve the combination of a problem solving methodology, knowledge sharing, and ease of communication. It is guided by the following principles (Jorge et al., 2002).

- **Light management**. Clarity of objectives should be paramount. Management of the project is required so that sufficient information flows within the collaborating network and that a good solution is provided. However, the management does not control directly the work of each team.

- **Start any time**. All the problem information necessary to effect a solution is available at all times. This includes problem definition, data, evaluation criteria and any problem knowledge produced by other project participants.

- **Stop any time**. Problem solving effors should be conducted by each team so that a working solution is available whenever a stop signal is issued.

- **Problem Solving Freedom**. Members of the data mining project have complementary expertise and tools. Each team is in the best position to decide which approach to follow for the given problem.

- **Knowledge sharing**. As each data miner experiments and produces new knowledge on the problem, this should be shared with all the participants.

- **Security**. Data and problem information must be kept confidential, and appropriate controls must be applied to accessing such.

So far, the RAMSYS efforts have focussed on supporting the *Data Preparation* and *Modelling* phase in a remote-collaborative setting. This paper considers how the *Evaluation* phase can also be supported effectively by RAMSYS.

Some of the basic requirements of the RAMSYS methodology is the emphasizing and availability of the *current best understanding* (Voß et al., 2001) of the data mining problem. This has been implemented using the academic groupware platform Zeno (Gordon et al., 2001), by providing *coordination*, *collaboration*, *communication*, and *awareness*. The provision of these features are achieved by utilizing (new) features in Zeno including *task management*, *resource management*, and *discussion sections*.

So far the RAMSYS methodology has been trialed (in part or in full) on the following data mining projects:

- Marketing in the Insurance domain – clustering and predicting customer's purchasing propensities.

- Web log analysis in the public sector – improving site usability.

- Resourse scheduling for health and leisure farms – identify customer groups and predict the resources such groups are likely to require.

It is this third project that serves as a case for this paper. We describe the project in more detail in the following section.

3. An Experiment with Collaborative data mining: the SPA Problem

3.1. The SPA Problem

The "SPA problem" was offered to the SolEuNet consortium by a health farm. The health farm has a number of facilities that can be used by its visitors. More specifically, upon their arrival visitors are prescribed certain procedures to follow during their stay at the spa, as well as a schedule for them. The number of

[1] Other methodologies for data mining exist. See for example (Adriaans & Zantinge, 1996).

people that can simultaneously make use of certain facilities is limited. Thus the spa is faced with a classical scheduling task: given the procedures that newly arrived visitors need to follow and the limited capacity of certain facilities, create a suitable schedule.

In practice there is insufficient information to solve this scheduling task for the following reason. Visitors stay for several weeks and a schedule for their whole period of stay is made, but during their stay new visitors will arrive. While some information about these new visitors is available in advance (such as time of arrival, age, sex, ...) the procedures they need to follow will be known only at the time of their arrival.

The best one can do is to estimate the demand for the facilities for the near future, and use these estimates for producing schedules for the current patients. It is at this point that data mining comes in: by mining a database of previous visitors and trying to link properties of these visitors to the procedures they followed, predictive models could be built that estimate the demand for certain facilities based on known properties of future visitors.

Thus the data mining task can succinctly be described as follows: given a set of visitor descriptions that will arrive during a certain week, estimate how many of these visitors will need to follow each of some 40 available procedures.

The data set was available as a relational database, which means a reasonable amount of preprocessing was needed before data mining could start.

3.2. Collaborating Groups

Four different groups worked on this data mining problem. We refer to them as CTU (Czech Technical University in Prague), BRI (University of Bristol), LIACC (University of Porto) and KUL (University of Leuven). CTU served as contact with the end user (the health farm). These groups typically consist of 2 to 4 people, at the expertise level of PhD students or postdocs.

Following the RAMSYS methodology implies following the CRISP-DM methodology, hence we here briefly describe the efforts according to the different phases.

Phase 1 (business understanding) involved becoming familiar with the data mining problem, which was done by all groups separately. During Phase 2 (data understanding) several groups explored the data using visualisation techniques, association rule discovery, etc. and published their results on Zeno. In Phase 3 (data preparation) the main effort consisted of data transformations. As the original database consisted of multiple tables, this involved to some extent computation of aggregate functions. Data transformations were performed mainly by means of the SumatraTT tool (Aubrecht & Kouba, 2001) developed by CTU.

In this paper we focus mainly on Phases 4 and 5, the modelling and evaluation phase. There is an intense feedback from 5 to 4: based on the evaluation of produced models data miners wish to change their model building approach and go through Phases 4 and 5 once more. In the collaborative data mining setting the feedback should not remain within one group but flow to all groups for which it is relevant.

The different groups used the following approaches:

- BRI: support vector machines, relevance vector machines, linear regression, multi-layer perceptrons; a comparison was also made between static and dynamic approaches for time series prediction
- LIACC: model trees, linear regression, instance based learning, neural networks
- CTU: nearest neighbour, naive Bayes, linear regression, decision trees, subgroup discovery
- KUL: linear regression as implemented in Weka (Witten & Frank, 1999); Clus, a system for induction of predictive clustering trees derived from Tilde (Blockeel et al., 1998)

Besides the different algorithms, approaches also differed in the version of the data set that was used (these versions resulting from different data transformations).

3.3. Results of the collaborative data mining process

The results of this collaborative data mining experiment are both positive and negative. A positive result is that the results obtained in the end were relatively good; the end-user found them interesting and useful (Štepánková et al., 2002). The bad news is that the added value of collaboration of different groups on this task was much smaller than hoped. The most notable collaboration was that the results of data transformations performed by one group were used for modelling by another group. This is in line with the kind of collaboration that RAMSYS promotes, but it is a minimal version of it: much more such collaboration is desirable. In the following section we analyse in more detail what went wrong.

4. What Went Wrong and How to Improve It

In this section we describe a number of problems that were encountered when attempting to follow the RAMSYS methodology for collaborative data mining. Some of these were foreseen in RAMSYS and thus confirm the need for highly specific software support; others are new.

4.1. Information Exchange

One important idea behind RAMSYS is that the different groups that work on a problem share the results they have obtained. It is important that groups understand the results produced by other groups, which means these results must be documented. During the SPA experiment we observed that information flow between groups was hampered because of two main reasons: (a) documentation may be too concise, in which case a group may have trouble understanding the results, or (b) it may be too extensive, in which case the overhead of reading the documentation demotivates people. A mixture of both can even occur: there may be extensive documentation without groups being able to find the most relevant information in there.

Thus there is an overhead in both producing documents and reading documents, which should be kept to a minimum in order for collaborative data mining to work well. Apparently this requirement was insufficiently met in the SPA project.

What is needed in this respect, is a more formalised method for information exchange, in which it is clearly specified what the most relevant information is and how it should be communicated. This should minimize both the effort in producing documentation and in understanding it. It seems that more research is needed, however, to determine what the format for information exchange should be.

4.2. Synchronisation

The knowledge discovery process as described by CRISP-DM consists of many steps and cycles. The idea of collaborative data mining is that within each of these, collaboration may be useful. Since some of these steps are iterated over, it is clear that the exchange must be very efficient. Assume group A produces a result, group B builds on this and obtains a new result which is then built upon by A. In the mode of cooperation that was used for the SPA problem, it might take several weeks before B is able to look a A's result and take a next step, and consecutively it might take a few weeks before A can use B's results. As this whole process is just part of a single step, of which there may be dozens, it is obvious that this mode of collaboration is not feasible.

There are several causes of the slowness in this process. One is that academic partners in the SolEuNet project typically are unable to be available all the time for the SPA project. A second reason is that the used version of Zeno lacks a feature which is referred to as *awareness*: when one user publishes something on Zeno, other users are not aware of this until they check out the relevant area on Zeno. There are multiple areas on Zeno, and users visit them infrequently. The result is that it may take days or even weeks before one user is aware that another user has posted something.

The first problem mentioned above is difficult to alleviate in a purely technical way. A possible solution could be to plan the work better. The current mode of operation is relatively unorganized: people devote some time on the project when it suits them. A central planning agent could provide better synchronisation of the work. The second problem could be solved by adding awareness to RAMSYS. A new version of Zeno (version 2.0) has in the mean time been developed that does support awareness.

Summarizing: exchange of results and ideas is useful at a relatively low level, therefore it should be fast and efficient, and a good synchronization is necessary for this. Awareness is clearly an important aspect; better planning of the work is a second one.

4.3. Comparative Evaluation of Models

It is obvious that in order to compare different models, they have to be evaluated according to the same criteria. The original RAMSYS methodology proposed to determine an evaluation criterion in advance so that each group can evaluate their models according to this criterion.

The SPA experiment revealed several problems with this proposal.

- It is difficult to propose a precise evaluation criterion in advance. In fact, it was unclear at the beginning of the SPA experiment exactly what the goal was. Exploratory analysis gradually developed into predictive model building. Consequently, each group reported the criterion that was most natural or easily obtainable for them (e.g. if a tool reports the RMSE of a given model, this RMSE was communicated).
- The preferred evaluation criteria may change over

time, as a result of the knowledge gained during the data mining process. For instance, in the SPA experiment visual data analysis in some cases revealed strong outliers. As it turned out, these were related to unavailability of certain procedures because the facility was under maintenance. Such outliers may dominate RMSE values, which essentially makes the whole comparison between data mining approaches or the resulting models unreliable (even if everyone has agreed on using RMSE). Note that outlier detection is not always easy in advance but may be a result of the data mining itself and discussing results with the end user.

- There may be discussion on which evaluation criterion is most relevant. In fact, it may not be the case that one evaluation criterion is sufficient. It seems more realistic to talk of a set of evaluation criteria, instead of a single criterion. Different criteria measure different properties all of which may be relevant (see e.g. (Köpf et al., 2001)).
- There may be subtle differences in the computation of certain criteria, the data set from which they are computed (including or excluding outliers), differences in the partitioning used for cross-validation, ... which make the comparison unreliable or not optimally powerful.
- When criteria evolve, some overhead is involved for the different groups in adopting the new criteria. This causes a certain slowness and reluctancy to change the evaluation criteria among the data mining groups.

The lesson learned here is that the proposed evalution methodology of having a single evaluation criterion that is determined in advance and does not change over time seems unrealistic. It is necessary to have a more flexible evaluation scheme in which criteria can be changed, new criteria can be added, and it is guaranteed (enforced) that every group uses exactly the same version of a criterion, all this without significant overhead for the different groups. This implies for instance that all groups should not be forced to implement all criteria themselves.

Note that what we are discussing here is a problem of *implementation*. Issues concerning the need for fixed evaluation criteria and which evaluation criteria to use have been raised before, and these are not the focus of this paper. The question is rather how to ensure that, in a collaborative setting, different groups can efficiently adopt the right evaluation criteria, in a setting where these criteria evolve.

The simplest way to achieve this is to have *centralized model evaluation*. Instead of having all different groups evaluate their own models, one should have a kind of model evaluation server to which groups send the models they have produced, or the predictions produced by their models. When a group decides they are interested in some specific criterion, they should be able to add the criterion to the central evaluation server and immediately see the scores of all earlier submitted models on these criteria.

While the problems mentioned in Sections 4.1 and 4.2 are perhaps more important than the comparative evaluation problem, we currently have no concrete solutions for them. We do have a concrete proposal on how to implement a centralized model evaluation procedure, and this is what we will focus on in the remainder of the paper.

5. Centralized Model Evaluation

The original RAMSYS proposal contains a so-called data master, which is a database containing the original data, results of transformations, and meta-data. The meta-data might consist among other things of information about which folds should be used for cross-validation: it is sufficient to add a single attribute to a table which for each instance identifies the fold in which it should be a test example.

The mode of operation for this model is as follows: when a group wants to evaluate a technique, they download the data with fold information, run a cross-validation consistent with the fold information, and report the result. A precise evaluation criterion should be decided upon in advance.

As mentioned, our SPA experience reveals a number of problems with this approach: the evaluation criterion may change (e.g. one may wish to leave outliers out of the evaluation), people tend to report the measures that their tools compute but are reluctant to implement measures themselves, and in the best case, assuming they do adopt some new criterion, it will at least take some time before all groups have adopted it and have produced the corresponding results.

Our proposal concerning centralised model evaluation is as follows. Data mining groups ("clients") should send predictions or even the models themselves to a "model evaluation server", which is responsible for the evaluation of the predictive model and automatically publishes the results.

Several levels of communication are possible. An inductive system typically has a number of paramet-

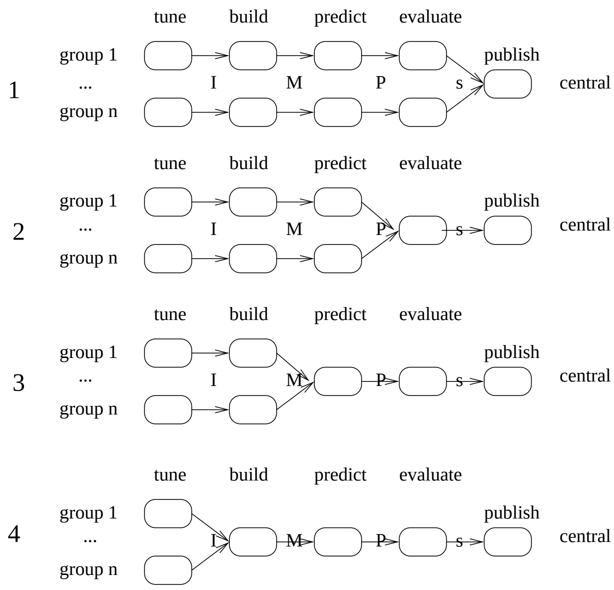

Figure 1. Overview of different options for centralizing model evaluation in collaborative data mining.

ers; for a given set of parameters values the system implements a function $I : 2^{X\times C} \rightarrow (X \rightarrow C)$ that maps a dataset (a subset of the universe of labelled instances $X \times C$ with X the instance universe and C the set of target values) onto a function M (a predictive model) that in turn maps single instances onto some target value. One has the option to submit the inductive function I; the model M learnt from a given data set T; or a set of predictions for some data set S, $P = \{(e, M(e)) | e \in S\}$. In all cases the server should be able to derive from the submission a score on one or more evaluation criteria, which we assume to be a function $c(M, P)$ (i.e. evaluation can be based on the model itself as well as on the predictions it makes). The current procedure in fact corresponds to a fourth option where $s = c(M, P)$ is communicated.

A schematic overview of these four different options (in reverse order compared to above) is given in Figure 1. It is assumed that I consists of a combination of a machine learning tool and parameter settings, so I is the result of tuning the tool with the parameters. Using I a model M is built from a training set, this M is used to predicted labels for a test set S, from these predictions a score s is computed using the evaluation criterion c. Note that in the case of a cross-validation the process becomes more complicated but the same basic scheme is valid: different models M_i are then built from different training sets, to produce one single set of predictions P.

Table 1 summarizes a number of characteristics of the different options. In the table H, M and L refer to High, Medium and Low respectively. Language complexity refers to the language that is needed for communication. Options 3 and 4 impose the challenge of developing relatively complex languages and interpreters for them (e.g., when submitting a model M the server needs to be able to compute the predictions M makes on some test set). Communication cost is lowest when just a score needs to be communicated, may be high when a set of predictions needs to be communicated (assuming the data set can be large), and is medium when function descriptions are communicated. Result availability refers to how fast the scores of different models, according to a new criterion, are made available for other data miners to study. This is low for Option 1 because here the different groups need to implement the computation of the new criterion themselves; for the other options recomputation happens automatically as soon as an implementation of the new criterion becomes available. Comparability reflects the trust in the comparability of the results, which is higher when a single implementation is used (avoiding possible differences in local implementations). User overhead refers to the overhead for the data mining groups when some option is adopted. In Option 1 it is highest, in Option 4 lowest because the user need only submit I (induction system + parameters) and all testing is then done automatically. In Options 2 and 3 the user needs to implement e.g. cross-validation according to given folds. Finally flexibility of evaluation criterion is lowest for Option 2 because with this option the criterion cannot involve the model itself (complexity, interpretability, prediction times) but only its predictions. This still supports a wide range of criteria as long as predictive accuracy is the most important element. For purely descriptive induction it is less suitable.

	1	2	3	4
language complexity	L	L	M	H
communication cost	L	H	M	M
result availability	L	H	H	H
comparability	M	H	H	H
user overhead	H	M	M	L
flexibility of evaluation	H	M	H	H

Table 1. Characteristics of different options.

Option 1 is the current mode of operation within SolEuNet. Option 2 provides significant advantages over Option 1 with respect to automatic availability of all evaluation scores for all models, and is still easy to implement. Option 3 imposes the challenge that a good model description language and an interpreter for it need to be available. A reasonable choice for such a language would be PMML (Wettschereck & Müller, 2001). PMML is already being proposed as a common language in which models should be represented; it handles a reasonable amount of different types of models and there exist visualisers for them. Under the assumption that PMML is going to be used anyway in a collaborative data mining system, an interpreter for PMML models would be sufficient to cater for a wide range of different model evaluation criteria. Cross-validation would still require cross-validation folds to be known and used by the client, but testset evaluation works fine.

Option 4 is the most powerful one but seems least feasible. There are different suboptions: (4a) all model building systems are translated into a single common language; (4b) the central model evaluation server has the necessary interpreters for the different languages in which inductive systems, data preprocessing systems, etc. are programmed; (4c) the server has its own versions of the inductive systems, and all that is actually submitted is an identifier of the system to be used and a list of parameters. Option (4c) seems the most feasible among these, but has the disadvantage that only the systems and versions available at the server can be used.

Note that Option 4c is somewhat similar in spirit to the option taken in the European Metal Project (`http://www.metal-kdd.org/`) where samples of data mining data are uploaded to a central server which executes numerous data mining algorothms and presents the results with respect to a range of criteria, in an attempt to suggest the most appropriate method.

In the short term, we believe the most realistic improvement to the RAMSYS model corresponds to Option 2 (submission of predictions). This option is easy to implement and already presents a significant improvement over the current mode of operation. In the longer run, under the assumption that PMML is general enough to describe any kind of model that could be submitted and that interpreters are available, it seems desirable to shift to Option 3.

Summarizing, a centralised model evaluation:

- reduces the workload of collaborators
- increases the confidence in the comparisons made between systems
- guarantees availability of all criteria for all models
- reduces the time needed to obtain scores on new criteria
- adds flexibility w.r.t. defining new criteria

We believe that all of these contribute significantly to the added value that a collaborative data mining process may have over the non-collaborative approach. (Although they are obviously not sufficient: as mentioned other problems exist that we do not solve in this paper.)

6. Conclusions

Collaborative data mining, as promoted by and used within the SolEuNet project, is not a trivial enterprise. In order for it to work well, a highly tuned supporting environment is needed. This was recognized early on in the project, which led to the RAMSYS proposal.

An experiment with collaborative data mining, following the RAMSYS methodology as much as possible, failed in the sense that while good results have been obtained in the end, the added value of the collaboration of different groups seems to have been small. Several reasons have been identified: cumbersome information exchange, difficult synchronization, and the lack of an environment supporting flexible comparative model evaluation. The last of these problems is most easy to solve, if a centralized model evaluation procedure is implemented. A proposal to move in this direction has been made in this paper.

The other problems mentioned are not less important, but we currently do not have a solution for them; further research seems necessary. Another point is that while we here focus mainly on technological aspects, there are also psychological aspects to collaboration, which we have not investigated here. Most data mining people are used to a competitive setting to such an extent that even when the necessary environment is present, it is not obvious that it will be used optimally. This, too, is a possible subject for further investigation.

Acknowledgements

This work was supported by the European IST project SolEuNet. H.B is a post-doctoral fellow of the Fund for Scientific Research of Flanders (FWO-Vlaanderen). The authors thank their colleagues at the universities of Prague (CTU), Bristol, Porto, and Leuven, and the Fraunhofer Institute, for the collaboration on the SPA problem. They thank Olga Stepankova for comments on this text, as well as the anonymous reviewers for their useful suggestions. The SPA data set was provided by Lauryn v.o.s.

References

Adriaans, P., & Zantinge, D. (1996). *Data mining*. Addison-Wesley.

Aubrecht, P., & Kouba, Z. (2001). Metadata driven data transformation. *Proceedings of the 5th World Multi-Conference on Systemics, Cybernetics and Informatics*.

Blockeel, H., De Raedt, L., & Ramon, J. (1998). Top-down induction of clustering trees. *Proceedings of the 15th International Conference on Machine Learning* (pp. 55–63). `http://www.cs.kuleuven.-ac.be/~ml/PS/ML98-56.ps`.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). CRISP-DM 1.0: Step-by-step data mining guide.

Gordon, T., Voß, A., Richter, G., & Märker, O. (2001). Zeno: Groupware for discourses on the internet. *Künstliche Intelligenz*, *15*, 43–45.

Jorge, A., Moyle, S., Voß, A., & Richter, G. (2002). Remote collaborative data mining through online knowledge sharing. *PRO-VE'02, 3rd IFIP Working Conference on Infrastructures for Virtual Enterprises*. Portugal.

Köpf, C., Taylor, C., & Keller, J. (2001). Multi-criteria meta-learning in regression. *Proceedings of the ECML/PKDD Workshop on Integration of Data Mining, Decision Support and Meta-Learning*.

Štepánková, O., Lauryn, S., Aubrecht, P., Klema, J., Mikšovský, P., Nováková, L., & Palouš, J. (2002). Data mining for resource allocation: A case study. *Intelligent Methods for Quality Improvement in Industrial Practice*, *1*.

Voß, A., Gärtner, T., & Moyle, S. (2001). Zeno for rapid collaboration in data mining projects. *Proceedings of the ECML/PKDD Workshop on Integration of Data Mining, Decision Support and Meta-Learning* (pp. 43–54). ECML/PKDD'01 workshop notes.

Wettschereck, D., & Müller, S. (2001). Exchanging data mining models with the predictive modelling markup language. *Proceedings of the ECML/PKDD Workshop on Integration of Data Mining, Decision Support and Meta-Learning* (pp. 55–66). ECML/PKDD'01 workshop notes.

Witten, I., & Frank, E. (1999). *Data mining: Practical machine learning tools and techniques with java implementations*. Morgan Kaufmann.