The Knowledge Engineering Review, Vol. 20:4, 363–402. © 2006, Cambridge University Press
doi:10.1017/S0269888906000762 Printed in the United Kingdom

Applications of machine learning: matching problems to tasks and methods

MAARTEN VAN SOMEREN[1] and TANJA URBANČIČ[2,3]

[1]*Human-Computer Studies, University of Amsterdam, Kruislaan 419, 1098 VA Amsterdam, The Netherlands;*
e-mail: maarten@science.uva.nl
[2]*Nova Gorica Polytechnic, Vipavska 13, PO 301, SI-5001 Nova Gorica, Slovenia;*
e-mail: tanja.urbancic@p-ng.si
[3]*Jozef Stefan Institute, Jamova 39, SI-1000 Ljubljana, Slovenia;*
e-mail: tanja.urbancic@ijs.si

Abstract

The terminology of Machine Learning and Data Mining methods does not always allow a simple match between practical problems and methods. While some problems look similar from the user's point of view, but require different methods to be solved, some others look very different, yet they can be solved by applying the same methods and tools. Choosing appropriate Machine Learning methods for problem solving in practice is therefore largely a matter of experience and it is not realistic to expect a simple look-up table with matches between problems and methods. However, some guidelines can be given and a collection that summarizes other people's experience can also be helpful. A small number of definitions characterize the tasks that are performed by a large proportion of methods. Most of the variation in methods is concerned with differences in data types and algorithmic aspects of methods. In this paper, we summarize the main task types and illustrate how a wide variety of practical problems are formulated in terms of these tasks. The match between problems and tasks is illustrated with a collection of example applications with the aim of helping to express new practical problems as Machine Learning tasks. Some tasks can be decomposed into subtasks, allowing a wider variety of matches between practical problems and (combinations of) methods. We review the main principles for choosing between alternatives and illustrate this with a large collection of applications. We believe that this provides some guidelines.

1 Introduction

Recent years have seen a large increase in the use of methods from Machine Learning (ML) in industrial applications. Overviews are given in, for example, Mitchell (1997), Hand *et al*. (2001) Han & Kamber (2001) and Fayyad *et al*. (1996). Under the name 'Data Mining' (DM), ML methods were combined with methods from the areas of databases and statistics and introduced into many application areas. This has led to a large increase in the use of ML methods. DM plays a key role in 'data warehousing' and 'business intelligence': the collection of data about business processes and the analysis of these data is integrated into the infrastructure and processes of many companies. Data are stored and accessed by standard querying languages and database systems, but this is extended with queries for more general information or knowledge that is derived by induction.

The terms Data Mining and Machine Learning are used for a wide variety of tasks and methods and show substantial overlap with statistical analysis methods and neural networks. In this paper we focus on methods that are traditionally known as 'symbolic Machine Learning'. These methods produce models of data that are non-numerical, but some methods include limited concepts for numerical modelling (e.g. conditions on intervals of values or simple numerical functions). In many

Table 1 The US DM market in millions of dollars. Source: IDC

Year	DM tools
1997	117
1998	152
1999	197
2000	256
2001	332
2002	486

Table 2 Distribution of DM applications over sectors of industry (percentages)

Sector	2004 (216 votes)	2003 (270 votes)	2002 (608 votes)
Banking	13	13	13
Biology/genetics/proteomics	8	10	5
Direct marketing/fundraising	9	10	7
eCommerce/Web	6	5	9
Entertainment	0	1	2
Fraud detection	9	9	8
Insurance	7	8	6
Investment/stocks	4	3	3
Manufacturing	4	2	5
Medical/pharmaceuticals	7	6	5
Retail	4	6	6
Scientific data	9	9	8
Security	4	2	2
Supply chain analysis	—	1	3
Telecommunications	6	8	9
Other	11	6	7

applications, neural networks are a direct alternative for symbolic learning methods in the sense that they address the same type of learning tasks.

Some statistical analysis methods are aimed at the same tasks as DM and ML. Some methods are studied in both areas, for example various regression methods and Bayesian networks. Traditionally the emphasis in statistics is more on hypothesis testing than on model estimation and in the context of estimation the focus is on methods that allow analysis of the variance of estimators. In DM there is more focus on exploratory analysis and on computational issues. It is likely that in the future the relations between these areas will be strengthened. Table 1 gives the estimated size of the US market for DM tools.

The technology for DM is currently well advanced, but it seems that its potential is still not fully realized. One indication for this is the difference between sectors of industry. A recent poll (from the KDNuggets Website, `www.kdnuggets.com`, September 2003) shows the following distribution of DM projects over sectors of industry, see Table 2.

Consider, for example, the situation in the Netherlands. The total expenditure on software in the Netherlands is around €32 billion in 2000, of which 34% is spent on telecommunications and 66% on information technology. The distribution of expenses (both internal in the company and external) over branches is approximately as in Table 3.

Table 3 Expenditure on computing by application field in the Netherlands in 2000

Branch	Expenditure (billion €)
Financial services	4.4
Government and education	2.9
Trade and retail	2.1
Process industry	2.0
Business services	1.9
Computing services	1.2
Transportation and distribution	0.9
Discrete industry	0.7
Other	0.7
Health care	0.5
Telecommunication	0.4
Media	0.3

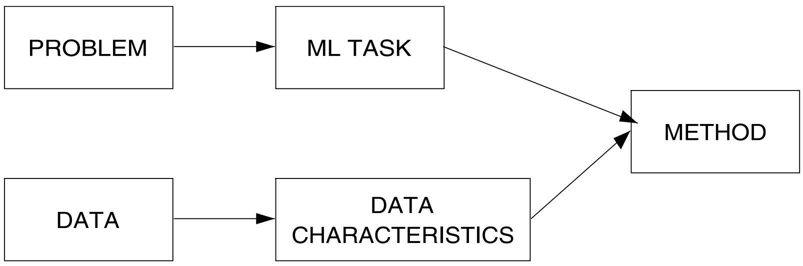

Figure 1 Process overview

Although the number of applications is relatively small, we can see several discrepancies between the overall expenditures on computing and those for DM. In the Dutch economy, financial and commercial services are relatively important, as is agriculture; the production industry makes up a somewhat smaller part than in the worldwide situation on which the data in Table 2 are based. The proportion of ML applications in the health care sector (including medical biology, genetics and pharmaceuticals) is much higher than its share in computing expenditure, and the opposite is true for financial services. Some sectors show very few ML applications. Notable examples are government and education and the computing industry. These patterns are also visible in an earlier study of the Dutch situation (Verdenius & Van Someren, 1997). A reason for the lack of penetration in government and administration may be the legal responsibility of the organization for its actions. If a government body takes an administrative decision then this in many cases should not be based on an inductive argument but on regulations. Yet there seems to be room for ML in optimizing administrative procedures. An example could be the optimization of workload by analogy with optimization of computer networks. Another opportunity seems to be the fine-grained optimization of procedures. ML methods can be used to support procedures by learning from examples to produce a draft decision or document, which also reduces workload.

One of the main reasons for the slow adoption of DM and ML technology is simply the lack of awareness and training, but another factor is the difficulty of matching practical problems with methods for solving them. DM and ML are basically methodologies for automatically analysing data and in this sense they differ from methods for knowledge acquisition that rely on manual model construction (e.g. Schreiber *et al.*, 2000). However, in practice the application of learning methods almost always involves manual modelling. Methods often cannot be applied directly to a problem as it is formulated by the ‘problem owner’. It is unusual for a problem to be presented directly as a ML problem. In practice, the application of a ML system is only one step in a complex process that involves dealing with a number of issues, such as acquiring and cleaning observational data, selecting and tuning a learning system, interpreting and presenting or implementing the results of learning. For a comprehensive methodology for the application of ML in this broader sense see, for example, Chapman *et al.* (2000).

In this paper, we address an issue that is part of existing methodologies but for which little conceptual and technical guidance is available: the conceptualization of the problem as a ML problem. Problems that occur in practice often need to be reformulated to see whether and how ML can be applied. ML experts are usually able to make such transformations, but for non-experts it is not always clear how ML could be applied. Moreover, many practical problems require more than a single ML application. For example, optimizing a production process can sometimes be achieved by first constructing a simulation model of the process and then using this to optimize control settings. This requires two different types of methods. To distinguish different types of methods, researchers use technical terms that are uninformative for 'problem owners'. For example, for non-experts it is hard to understand what the terms 'supervised/unsupervised learning', 'reinforcement learning' or 'evolutionary learning' precisely mean. Publications in the scientific literature prevailingly take the technical innovation as their focus and minimize information about the application problem. Matching practical problems to ML methods is taught typically by example. Consultants collect experience and new 'problem owners' hire a consultant or try to find examples of similar problems with solutions. In this paper we take a more abstract approach and define 'application paradigms': typical ways of using ML formulated not in technical terms but in application terms. Each paradigm is illustrated with examples. We then associate these application paradigms with generic ML tasks and these with methods and tools. Applications may well be virtually identical from a technical viewpoint, but are entirely different from the viewpoint of the users and *vice versa*.

Methodologies for applying ML methods (e.g. Chapman *et al.*, 2000; Saitta & Neri, 1998) include a step that is called the 'learning problem definition' (Saitta and Neri), identification of the 'target function' (Mitchell, 1997) or 'business understanding' (Chapman *et al.*, 2000). In this step, the problem as presented by the client is transformed into a form in which it can be approached by ML methods. This step is generally seen as outside the scope of ML research and more of an art than a science. This step precedes collection, selection and pre-processing of data and selection and application of methods (as studied, for example, in the MiningMart project; see Morik & Scholz (2003)). In this paper, we present a number of examples of such transformations. The picture that emerges is that of a small set of types of learning tasks that are technically different, requiring different types of methods, and a set of transformations that can be applied to adapt a problem to the technology such that constraints imposed by the methods are satisfied. As this paper is based on our experience with European projects, we focus on European applications.

This paper is organized as follows. First we characterize learning tasks in technical terms. Next we describe a number of generic applications, organized by application field and the task that is addressed and we show how a practical problem is formulated as a ML task. The applications were taken from a number of applications described briefly in Appendix A.

2 ML task types

In this section we give an overview of learning tasks in the mathematical terms that underlie the methods and algorithms. We can classify ML methods by the types and structures of the input data and of the model that is constructed by the learning method, and by the relation between the input data and the model.

2.1 Learning data

The basic type of input is always a set of cases. A single case is described in a format that can, in general, have any possible type of elements and structures. A structure for a case can be a vector or a tuple, relational description, sequence, set or tree. Elements can be numbers, rank ordered or nominal identifiers. The components of structured types (dimensions of vector, objects in relational descriptions, elements of a sequence or a set, etc.) can themselves again have different types. The choice of the type for cases is of great importance because, in combination with the type of the learned model, it determines the models that can be found and the order in which a method will consider them. Another key decision concerns the information that is included in the case descriptions. It is often claimed (e.g. Langley & Simon, 1997) that the selection and structure of information in the cases is the key to success, specifically that it is more important than the choice of the type of the model or the method (e.g. Van der Putten & Van Someren, 2004).

In addition to cases, some learning methods can take as input explicit general prior knowledge, but because most methods take only cases as input, we shall focus on this type of learning data.

2.2 *Learned model*

The models that are learned are general patterns over cases. They usually involve relations between parts of the case descriptions. For example, one of the numbers in a case description may be a value that needs to be predicted from the rest of the case description. This means that to characterize learning tasks and learned models we must distinguish between the roles that parts of the case descriptions have (and that appear in the learned models). Given this, we can specify the purpose of a learning method in terms of the cases: which property does the method optimize? When all cases have the same structure, the corresponding parts can be viewed as variables and we can distinguish between the following classes of models.

- *Prediction models*: a model that expresses one designated predicted variable as a function of other predictor variables. Here we use 'prediction', without implying that the prediction concerns the future. Instead it means that it may be unknown and that we will then use the model to predict its value. The model should maximize the accuracy of the predictions, measured as the expected number of prediction errors, or if the size of the error is included, as the expected average prediction error. A useful distinction is that between single-value prediction and distribution prediction. Distribution prediction methods predict a probability distribution over possible values. Prediction models can be constructed for pre-assigned predictors ('supervised') or between arbitrary predicted variables and arbitrary sets of predictors. The predicted and predictor variables can be assigned by the user or the system may search for combinations of predictable and predicting variables (as in the search for association rules).
- *Clustering models*: divide the cases into case clusters or the variables into variable clusters, such that elements of a cluster are similar and different from elements in other clusters. Clustering can be applied to cases or to variables. There are no special roles for variables. Clustering can be combined with models of clusters. The purpose of clustering is usually the same as that of association models: find models that optimize predictive or characterizing relations between multiple variables. Like predictions, clusters can be deterministic or distributional. In deterministic clustering methods, each object is assigned to the single cluster to which it most belongs. In distributional clustering, objects belong to a cluster with a certain probability or other relative measure.

Some tasks differ from the tasks above in subtle ways and, as we shall see, are usually addressed with methods that were designed for other tasks as follows.

- *Association models*: specify relations between (subsets of) variables, such that the model concisely describes the relations between variables but, unlike prediction models, association models do not maximize predictive accuracy in one direction, but the total distribution in all directions. To see the difference with prediction models, consider what happens with strongly correlated variables. For prediction one of these might be enough. For association they will most likely both be

included because adding the correlated variable will increase the 'backwards' predictive power. This can even go at the expense of the predictive accuracy of the model. An intermediate model type between prediction and association models are models with predictors and a predicted variable in which the predictive accuracy of the predicted variable and the predicted accuracy of the predictors from the predicted variable are maximized, but not the relations between the predictors. There are few specialized methods for this, however, and for this type of learning task prediction models or association models are usually learned.
- *Characterizing models* (Michalski, 1983): a characterizing model is a model that optimizes predictive accuracy of the characterized variable and at the same time predictions of characterizing variables from the characterized variable.

Few methods have been developed for association and characterizing models and, in practice, prediction methods are used for these tasks, although this does not give optimal results.

In addition to the basic tasks, there are several derived tasks as follows.

- *Explaining*: which combination of variables 'causes' another variable. Many applications simply formulate an explanation as a prediction, apply corresponding learning methods and take the prediction model as a causal relation. One can argue about whether explaining and predicting are the same. For example, a variable may be predictable from several sets of predicting variables. A prediction model will produce only one such model and miss alternative explanations. Explanations are not used for making predictions but for human understanding of the domain. Therefore, comprehensibility of the model is an important criterion. A model that predicts well but is incomprehensible has little use as an explanation.
- *Finding exceptions*: for some problems exceptions from the general pattern are more useful than general patterns. These are found by first finding a prediction model and, from this, individual cases that deviate from the predictions.
- *Subgroup discovery*: find a subgroup that is large enough and has a substantially different distribution from the entire population. Subgroup discovery is different from clustering because the goal is not to partition the entire domain.
- *Control*: a type of learning task that happens to be studied less often in ML and DM is control (or optimization). From data about conditions, control settings (or actions) and payoff, a model is constructed that given conditions, finds control settings (or an action) that optimize the payoff. There are the following three approaches to optimization tasks.
  – *Cloning expert behaviour*: data are conditions of the process (including input, environmental conditions, control settings) and actions selected by a human expert. The learning task is to learn a prediction model for the experts action.
  – *Learning a model that predicts rewards*: data are process input, conditions, control settings and payoff. The learning task is to learn to predict the payoff and use this information in a system that selects and performs actions or to derive control knowledge.
  – *Reinforcement learning*: the most direct way is to use payoff direct to learn a 'policy', the relation between condition and (optimal) action or control setting. Although Reinforcement Learning provides a range of methods for this, we found few applications.

We can formalize the different learning tasks as follows, assuming data that consist of cases represented as variable value pairs. Variables can be partitioned into subsets that play different roles.

Given a set of cases (and corresponding variables) and a partition of the variables we can define a learning task as finding a program that implements one or more function on the cases.

Control or reinforcement learning is also studied under the heading control theory. A learning problem can often be decomposed into subproblems that correspond to different task types. For example, a prediction task can be decomposed into (a) clustering objects and (b) prediction within clusters. Prediction for sequences can be decomposed into predictions for single members of a sequence. Whether decomposition is effective depends in complicated ways on the characteristics of the data and the learning method.

3 Mapping application tasks to ML task types and methods

Although it may seem that there exists a simple mapping between application tasks and ML methods, this is not the case. This can be seen in several applications where the original problem was transformed into a different problem before ML methods were applied. Given 'raw' case descriptions and an informally expressed goal, modelling choices must be made for the following: a learning task, a data type, a type for the resulting model and a method. In some application projects the data have not yet been collected and there is still a choice.

These choices are interdependent because a method performs a task and requires a certain type of input. As we discussed above, most learning problems have more than one solution. We therefore need a way to make the best choices. Some choices amount to assessing which type of model matches best the original goal. For example, is the application goal to predict whether a customer will buy a certain product or to associate buying with other customer properties, and for a prediction task single prediction may be enough or distribution predicted is needed. These choices are a matter of judgement.

A second decision is the choice of data and model type, given the choice of a task. Fortunately, the choice of a task is more or less independent of the choice of data and model type because methods have been proposed for most combinations of tasks and data and model types. In practice the available tools limit the possibilities. The choice of data type can be very important and is the subject of current research (e.g. Morik & Scholz, 2003). Here we restrict the discussion to summarizing the goal of the optimization of data and model type.

Many practical learning tasks can be solved with different methods and combinations of methods. This is because 'raw' data can be represented as different types and because a learned model can be used to add information to a case description that can then be used for further learning. For example, the raw values of a single variable can be re-scaled or combined with values of a different variable into a new variable. Membership of a cluster can be viewed as a (nominal) variable and be added to a case description, and predictions for cases that are based on models using one type of case description can be added to a case description and used by another method.

3.1 Example: the COIL challenge 2001

Consider the following example, a cross-selling problem developed for the COIL competition (Van der Putten & Van Someren, 2004). In this case, data were already available. These were the properties of customers of an insurance company. The data included information on the insurance policies of the customers. The goal was to improve the response to direct-mail campaigns using information about customers that was present in the data. This problem can be formulated as a ML problem in different ways. To illustrate this we give three different formulations of the problem:

Profit optimization

Each mailing brings with it costs and benefits, most with some probability. One approach is to express the costs and benefits as a function of (the properties of) the mailing and then find the values of the properties for which the net benefits are maximal. The costs can be estimated, although they may depend on the number of mailings (as some costs are marginal and others are fixed). If a customer reacts positively, there is a probability that they will actually buy some insurance. In this case, we get a process that brings benefits and risks to the insurance company. This approach further involves an estimate of the probability that a (particular) customer will react positively to the mailing. This probability may depend on the properties of the customer that are in the data. To make this approach feasible, we may need to make some simplifying assumptions, for example that the costs of a single mailing and the expected benefits of an insurance policy are independent of the information in our data.

The data contain information about customers that already have one or more insurance policies. We may consider using ML to learn to predict (for new customers) whether they are likely to buy a particular insurance policy. However, the marketing department may think that among the current customers there may be some who could be interested in additional insurance services. If this would be true, how could these be identified?

Buying behaviour prediction
A simpler model is to only take the probability of buying insurance into account and not the costs. In this case, the main factor is the probability that a customer will react positively.

Customer segmentation with prediction
This approach starts from customer segmentation. If customers can be divided into segments of similar customers then it may be possible to predict the probability of a positive reaction to a mailing for each segment. This resembles the previous approach, but is subtly different because in the segmentation approach an attempt will be made to find segments independent of buying behaviour: the prediction approach may produce such segments as a by-product of prediction, within buyers or non-buyers.

Which approach is chosen will depend on an estimate, based on experience and prior knowledge, of the relevance of marginal costs, the dependency between customer properties and expected gain of a policy and of the nature and strength of the relation between customer data and buying behaviour.

3.2 Example: optimizing control settings

In spite of well-established control theory, there are cases where alternative approaches are desired, e.g. when a process to be controlled is highly nonlinear, changeable or even not sufficiently known. A frequently occurring example where ML offers such an alternative approach is the optimization of control settings. The data consist of control settings, measures on the process and/or the product and some payoff measure (amount or quality of product, value of product, etc.). A frequently occurring variant occurs when the effect of control settings depends on other features such as environmental conditions or features of the process ingredients. This problem can be formulated as a learning problem in different ways as follows.

- *Direct*: collect data with varying control settings and find settings that maximize the payoff. If the effect of settings depends on other features then a relation must be learned that gives the optimal settings as a function of the other features.
- *Via simulation*: however, in practice, the direct method is often very hard because the relation is uncertain and can be complex. A possibility is then to search for the inverse relation from settings and other features to the properties of the product, possibly including payoff. This does not give the optimal settings, but the relation may well be simpler and easier to discover. This can then be used as a simulation of the process and the optimal settings can be derived from the simulation by analysis or experimentation.

4 Selecting the best mapping of application problem to ML task

Some application problems are not translated into tasks that are the obvious candidates. In particular, prediction is often used for applications that seem to match 'characterization' tasks more closely. There can be several reasons for preferring a formulation of a problem in a ML task that differs somewhat from the original applications problem. Important factors are the availability of data and of methods and tools. For example, in the COIL problem the most direct approach would be a profit maximization approach, in which the expected (net) benefit of mailing is predicted and used for a decision on mailing. Neither data nor methods for solving this problem as an optimization task were available, however.

For some problems, several mappings to ML tasks are available. How can we choose between alternatives? Without going into technical details, the reasons for failure are, in general: (1) there may be nothing to learn because there is no systematic pattern or only a very weak pattern that cannot be found from a small set of cases and that is also not very useful; or (2) the model type for the result prevents the actual pattern from being discovered. Comparing mappings to different ML tasks involves a comparison of the uncertainty in the domain, the available data and prior knowledge that allows the choice of an adequate type for the target model.

5 Data characteristics

ML and DM tasks involve aspects other than the task, such as the use of prior knowledge, how cases are provided by the environment (one by one, all at once, on demand by the learner). The other main aspect along which methods vary are characteristics of the data. The most important is the scale type of the variables of the training data and the models. Prediction, optimization, clustering and association models can be learned for different types of variables playing different roles (e.g. nominal, numerical variables, relational variables or a mixture) and the form of the model is a factor: decision trees, rules, logical formulae, various types of numerical functions, etc. Table 5 gives an overview of the main data types.

Tables 6 and 7 list popular methods with their ML task and input and model type. Short descriptions of methods can be found in textbooks such as, for example, Mitchell (1997), Hand *et al.* (2001), which also give references to more detailed descriptions. Tools can be found, for example, via Websites of KDNET (`www.kdnet.org`) or KD Nuggets (`www.kd-nuggets.com`).

In the literature, methods are given for many combinations of ML tasks and types for input data and output model. Other characteristics of the data determine the accuracy of the models that are found by a method. The relation between these characteristics and the effects of methods is a major class of research questions.

A type of data that is increasingly important is textual data, documents and parts of documents, including documents on the Internet. As this area is very recent and the number of applications is relatively small, we do not include it here. Some basic information can be found in Mladenić & Grobelnik (1993).

6 Application tasks

In this section we describe the application tasks that were abstracted from approximately 40 European applications (see Appendix A). The main sources of these applications are a collection of the Jozef Stefan Institute in Ljubljana and a collection compiled for MLNET, a network of excellence in ML. The applications are summarized in Appendix A. An ‘application task’ is a generalized application that is characterized by input data and result of the learning process that are described with the same name in the domain.

We use quite broad domains, corresponding roughly to ‘sectors of industry’: marketing, banking/ insurance/finance, scientific research, production industry, information technology, social and medical care, human resource planning, administration and government, culture and entertainment. Obviously these sectors do not correspond completely to companies. Many companies operate in more than one sector or have internal activities in different sectors. For example, a company that produces parts of aircraft may internally make financial predictions and will need human resource management. The tasks listed at each ‘application field’ do not exhaust the possibilities. Instead, each task corresponds to one or more existing applications.We do not

Table 4 Overview of learning tasks

Task	Variable partition	Find *f* such that
Prediction	Predicted; predictor	*Predicted=f(Predictors), value(Case, Predicted) − f(Predictors)* is minimal over all cases
Characterization	Characterized; characterizor	Prediction of characterized from characterizors and prediction of characterizors from characterized over the domain
Explanation	Explanadum; explanans	*explanandum=f(explanans)* in contrast to prediction, here we want to find all *f*
Prediction exception	Predictor; predicted	Prediction function and now find cases for which *f(predictors) − value(Case, Predicted)* is large
Exception subgroups		
Reinforcement or control learning	No partition Observed; Control; Reward	Find the set of exception cases that are similar *control= f(Observed)* that maximizes expected *reward*
Clustering	No partition	Split cases into subsets such that within subsets average prediction and characterization are maximized and between clusters these are minimized. Clustering can be accompanied by characterization or prediction of cluster membership
Discovery	No partition	Prediction and explanation without variable partition

Table 5 Types of variables

Scale type	Definition
Nominal	Values are different
Ordered	Transitivity
Interval	Distances between neighbouring values are equal; metric
Sequential	Sequence of values instead of one value
Relational	Relation between objects instead of one value

claim that this list exhausts the possible types of applications. Our goal is rather to present applications at a level that is less abstract than the technical formulation of ML tasks and also more general than single applications. We expect that the number of possible generic applications is much larger and that many new generic applications will emerge in the near future.

For each application problem we give its ML task formulation. This consists of the name of the ML task (as defined above) and the role of data from the domain. For example, the application problem *customer segmentation* corresponds to the ML task *clustering* with customer properties as input variables and clusters of customers as model. Some tasks are solved using compound ML tasks. For example, one formulation of *customer segmentation* is to first apply clustering using

Table 6 Examples of prediction methods

Tool	Method		
	Predicted type	Predictors type	Prediction model type
Decision tree learning	Nominal	Mixed	Decision tree
Linear regression	Numerical	Numerical	Linear function
Linear discriminant analysis	Nominal	Numerical	Linear discriminant function
Naive Bayes classifier	Nominal	Nominal	Distribution
Backpropagation	Nominal	Mixed	Neural network
Probabilistic decision trees	Nominal	Mixed	Probabilistic decision tree

Table 7 Examples of clustering methods

Method	Variable type	Model class
K-means clustering	Mixed	Cluster structure
Loglinear model estimation	Nominal	Log-linear models

customer descriptions as cases and then *prediction* to the resulting clusters. For each generic application we also summarize one specific application and we list references to additional summaries in Appendix A.

6.1 Marketing

Customer segmentation

Identifying groups of customers ('market segments') that have common characteristics. These characteristics can be of different types such as demographic properties, consumption behaviour, geographical properties. The goal is to obtain comprehensible knowledge of subgroups of customers to be used by marketing experts. The learning methods search for groups of customers that have many characteristics in common and that differ from other customers. This is a form of *clustering*. An additional possibility is to construct descriptions of groups, for example in the form of average or characteristics values, removing characteristics that do not distinguish between segments. This amounts to characterization or *prediction* of cluster membership. A human domain expert is usually needed to interpret the cluster descriptions in terms of variables that are not in the data. Segments can also be hierarchically structured such that small segments form larger segments.

An example of market segmentation is described in Van der Putten (1999). A publishing company was interested in the structure of the market for media. Survey data were available with a number of properties that included media consumption. The data included both numerical and non-numerical properties. Clustering the data revealed a number of subgroups that were characterized by a combination of media use and social, educational and economic variables.

Cross-marketing

Predicting the probability that a customer will buy a product or service from data about the customer. A similar problem is predicting which product a customer is most likely to buy or predicting, for each product, whether a customer is likely to buy it or not. The result can be a system that directly *predicts* which customers are likely to buy a product or it can be a comprehensible model of the market to be used by a human marketing expert. A more subtle version of the problem takes information about the expected costs of approaching a customer and the expected benefits of selling an item into account. An example of a cross-marketing application concerned insurance policies. Data were properties of customers of an insurance company. The data included information on the insurance policies of the customers. The goal was to improve the response to a direct-mail campaign aimed at selling caravan insurance policies. This was done by using customer data to learn a model that predicts if a customer would buy a caravan insurance. Customers with a probability above a certain threshold were mailed (Van der Putten & Van Someren, 2004). Another example of prediction of buying behaviour is Van der Putten (1999).

Explaining and predicting customer loyalty

A more general version of cross-marketing is to predict whether a customer will stay or go to another brand/supplier. The relation between properties of customers (including earlier transactions) is used to predict customer loyalty. This is to be used by the marketing experts and must be comprehensible. An example of a customer loyalty application is given by Vanhoof *et al.* (1997). They used customer data about car sellers to *predict* whether customers would change seller or brand. The study identified the main factors that determine customer loyalty and discovered that these factors vary between brands. One can argue about whether the use of prediction for explaining is appropriate. For example, data may allow explanations from different subsets of predictor variables. A prediction model will produce only one such model.

Adaptive recommendation

Generating recommendations for individual users by predicting whether a customer is likely to buy a product or service. The technology can be used for many types of applications, such as recommending documents on the Web, books, CDs and other products, television programmes, films, etc. This is very similar to cross-marketing but in an on-line individual setting: data about the customer may include traces of requests for Webpages and preferences indicated by the user. Recommender systems are typically adaptive systems that are embedded in a browser or eCommerce Website and that learn sustained from data that are collected continuously. Recommenders use descriptions of the content of pages and usage patterns to *predict* items in which a user is interested. Recommending is closely related to cross-selling and also to advertising.

An example of a recommender system is given by Middleton *et al.* (2004) who combined machine learning with prior knowledge in a system that learns to recommend books and CDs. Adaptive recommending can also be viewed as an optimization task in which the recommendations are optimized for benefit or user satisfaction (Hollink *et al.*, 2004).

Predicting market development

A different type of application is the *prediction* of the development of markets. Cyclic developments and the effect of external factors can be studied by using data to construct a predictive model for parameters of a market. An example is the prediction of gas consumption. For example, Klema & Kout (1999) applied ML to the prediction of the local consumption of gas from features such as weather forecasts, temperature, holidays, etc. Gas consumption is predicted 24 hours ahead, which enables the gas production company to optimize production and storage of gas.

6.2 Banking, insurance, finance

Predicting financial risk

The task is to *predict* the financial state of a company, department, a person or other entity after some time from properties of its current state and economic conditions. The result usually must be comprehensible knowledge because the prediction is used by a person to make a decision. Input for the application are records with information of comparable companies, departments, persons, etc., about which the later status is known. An example is a credit-worthiness application by Paass & Kindermann (1998). The expected profit on loans was predicted from properties of clients. Thomaides *et al.* (1999) predicted bankruptcy of companies.

Recognizing fraudulent transactions
Recognizing fraudulent financial transactions, e.g. credit-card payments, from information about the transaction, money laundering. If data can be collected about fraudulent transactions then *prediction* is appropriate. If this is not known then *finding exceptions* is appropriate, assuming that fraud is exceptional. One example is described by Fawcett & Provost (1997). They learned a model for recognizing fraud with cell phones from record of calls that were classified as fraudulent.

6.3 Scientific research

The area of scientific research is very wide and also very diverse. Most applications involve the exploration of models to predict or explain certain observations, not to test hypotheses. The purpose of the selection below is to show both the diversity and the commonalities between applications in different subareas.

Constructing a model of a process or structure
A model is constructed that finds the relation between measured variables. The goal is comprehensible knowledge of the process or structure. This corresponds to exploratory analysis of empirical data. A small number of very successful applications have shown the potential of ML technology here. In particular, inductive logic programming methods applied to domains that involve structured objects such as organic chemicals have shown the potential of ML for discovery. ML methods compete with statistical analysis and methods from applied mathematics. The strength of ML methods is in non-numerical modelling and nonlinear structures and relational modelling.

An example of the use of ML for scientific research is the construction of a model for *predicting* the toxicity of chemicals (Srinivasan & King, 1999). For research and development of new medicine and also for other chemicals it is important to know in advance whether the chemical will be toxic. One of the purposes of these studies is to construct a model that *predicts* from their chemical composition and structure if new chemicals will be toxic. This can be used to avoid expensive testing or even chemical engineering to construct it. In this problem, not only are the presence of chemical components and their properties relevant, but also how the components are structured. Therefore, relational learning methods proved to be very effective. Similar studies are reported in King *et al.* (1995) and Bratko *et al.* (1998). Learning *prediction* models for various biological problems are reported in Debeljak *et al.* (2001) for modelling the population of red deer, Blockeel *et al.* (2004) for the prediction of biodegradation of chemicals, Kraakman (1998) for the prediction of characteristics of offspring of vegetables, Dalaka *et al.* (2000) for the prediction of photosynthesis. Constructing and extending a classification of plants by *clustering* plant descriptions is presented in Alberdi & Sleeman (1997). Džeroski *et al.* (2003) address modelling soil radon concentration for earthquake prediction. The radon concentration is predicted in normal environmental conditions. If the observed concentrations deviate from the predicted values, this deviation predicts earthquakes about a week in advance, an example of *finding exceptions*.

6.4 Production industry

A simple example is given by Moczulski (1999) for selecting parts for a device. A system was trained with examples of correct and incorrect configurations to advise which parts to use in constructing a device.

Constructing a simulation model of a process

For many production processes there is no complete and accurate model that relates characteristics of the input and control settings of the process to properties of the product. ML and case-based reasoning (CBR) are used to construct 'flat' models of processes that directly relate input to output of the process. These models are used as knowledge for human experts who obtain a better understanding of the process or to optimize the process. The model can be knowledge that is inspected and analysed by humans or a system that is embedded in an optimization system.

For many processes, a model is available for part of the process or for underlying, unobservable processes. By collecting data about the remaining part of the process the model is complemented with a partial model of the missing part or about the relation between observable properties and underlying properties. This is a variation of the task of constructing a model from scratch, but it requires methods and tools that support acquisition and use of human knowledge about a domain.

An example is learning a model to predict the effect of adding chemicals to a steel production process (Gams *et al*., 1997). Data were collected that consisted of control settings (which chemicals were added, the quantity and at which time) and materials input into the process and properties of the steel that was produced. A general *prediction* model was learned that enabled analysis of the effect of changing parameters. Learning prediction models along the same line is reported in Knowledge Process Solutions (2002) (for potash production), Vasconcelos & Pecas Lopes (1999) and Matos *et al*. (1999) for wind turbines.

Optimization of control parameters

An example of this type of application is the minimization of chemical waste output (Sanchez-Marre *et al*., 1996). Controls of production processes affect the output of chemicals such as nitrogen into air or water. Due to regulations it is important to minimize this output. In several applications, data were collected on different control settings and other parameters of the process and chemical waste output. Several types of ML and CBR methods were used to learn a prediction model for chemical waste output. This model was then used to set controls such that waste output was minimized.

Examples of 'behavioural cloning' are Filipič & Junkar (2000) for grinding, Gams *et al*. (1997) for a steel rolling mill, Bratko *et al*. (1998) for an electric discharge machine, Urbančič & Bratko (1994) for controlling cranes. In Bratko & Suc (2002), behavioural cloning for crane control is upgraded with the task of explaining, which is done through learning of the operator's subconscious subgoals and the use of qualitative control strategies.

Explanation of production failure

ML methods are used to identify the conditions under which the process fails to meet quality standards. The resulting knowledge can be used to improve the control parameters or other aspects of the process. Input for the learning application are records of 'runs' of the process described by control settings, inputs to the process and other conditions in the environment with failure versus success of the process. This is typically done once because the goal is to modify the process so that the failure is removed. The approach is similar to that of constructing a model that simulates the production process. An example is Manago & Auriol (1997a, 1997b) who applied this approach to the diagnosis of aircarft engines and robot arms. Both learned *prediction* models.

Automating interpretation of measurements

The interpretation of sets of measurements to underlying variables is a frequently occurring task in industry. By collecting logs of measurements and associating these with interpretations obtained from an expert or from deeper measurements obtained at a later stage or under controlled conditions, input for a learning system is collected and a general model can be constructed automatically. The resulting model *predicts* the interpretations.

An example of this is an application in security management of wind power plants (Vasconcelos & Pecas Lopes, 1999). Varying weather conditions were simulated and the effect on measurements from the power plant were measured. This information was used to learn how weather conditions disturb measurements, which in turn is used to detect security problems. Another example is Džeroski *et al.* (1998).

Learning mesh design

In engineering analyses of stresses and deformations in physical structures, numerical methods such as the finite element method (FEM) are used extensively. Before these methods can be applied, a structure under consideration has to be approximated with a mesh model which consists of finite elements interconnected in the nodal points. The task in mesh design is to determine an appropriate geometric model which corresponds to the shape of the structure, ensures low approximation errors by fine mesh in areas where high deformations are expected, and avoids unnecessary computational overheads by coarse mesh in other areas. Determining where the mesh should be fine and where it should be coarse is a difficult task which requires a lot of knowledge and experience. Dolšak *et al.* (1998) report about learning mesh design from examples of known good meshes.

6.5 *Information technology*

Few applications are known that address problems in information technology itself. There are three main types as follows.

Detection of illegal operations

Detecting operations in a computer system or network that are 'illegal' in the sense that they are caused by a user who is not entitled to this. Examples are 'hackers' who have illegal access to a system or a user who performs or initiates illegal operations. Data are records of computer operations that are classified as legal or illegal. They include parameters of the actions and of the context of the action. The resulting models are used to recognize illegal operations. Methods used are symbolic learning methods and genetic algorithms.

An example is an experiment by Mischiati & Neri (2000). A collection of TCP/IP transactions was classified as legal/illegal. Illegal were intruders trying to get access to the network or to perform illegal operations. The system learned to classify transactions as legal/illegal. A similar example is Kephart *et al.* (1995) who learned to recognize files as computer viruses. Both used learning *prediction*.

Explaining load patterns

Similar to chemical processes and other industrial processes, performance data about a system are used to construct a simulation model of the system that explains and predicts when components of the system are overloaded. This model is then used to explain problems. The input for learning include parameters of user input (number of users and jobs, etc.), but also the structure and function of the network. The simulation can be used to optimize the architecture, control settings or use of the system. For example, Knobbe *et al.* (1998) describe a system that discovers rules that explain the occurrence of network problems from properties of the network, its structure and inputs. The prediction model was then used to optimize network performance.

Adaptive filters

Descriptions of documents with rating of documents from users is used to learn to recognize documents that the user finds useful or interesting. Documents on the Web are described by the occurrence of words or other patterns. The result is a system that is embedded in, for example, a browser, search engine or mail handler. One type of application is a spam filter that removes undesired mail. Another type directs e-mail to the appropriate person in an organization. In a browser a model of the user's interest is used to recommend documents. In each case, the properties of the mail, Web document and user are collected together with evaluations, for a large number of e-mails. The learning system extracts a pattern (model) and this is used to filter or recommend objects.

There are many examples of this type of application. One type is adaptive spam filters that learn to recognize spam from classified e-mail messages. Spam is a significant practical problem. A system can be trained to recognize spam messages and delete them or store them at a special place for later scanning. Examples are described in Androtsopoulos *et al.* (2000) and Armstrong *et al.* (1997).

6.6 Social and medical care

Diagnosis

It relates symptoms and other properties of patients to their disease. In some cases this is a comprehensible model in other cases it is a system that is either embedded or stand alone. Learning is typically done once from patient data with the (probably) correct diagnosis. If this is done continuously (including detection of deviations) it is called *monitoring*. There are a number of variations on this theme. Data consist of patients with correct *treatment*, from which the learner can learn to select the best treatment, or patients with later conditions, with or without treatment. This makes it possible to learn to predict the future condition of a patient with/without treatment.

A typical example of diagnosis is described by Andrews *et al.* (2002). Patients with head/neck injury were given two forms of treatment. The purpose of the study was to learn to predict in an early stage the type of injury which can only be established with high certainty by later examination of the patient. Although a precise diagnosis can be given after examination, early recognition is important because the type of treatment is different for different types of injuries. The result was a decision tree that predicted the type of injury with enough accuracy to be practically useful. A large number of similar applications have been described, for example Kononeko (2001), Matsumoto (1995, 1999). An elaborate example of monitoring including modelling processes over time is described in Morik *et al.* (2000).

Discovering subgroups of patients

An example is coronary heart disease risk group detection Gamberger *et al.* (2003) show that subgroups of patients with coronary heart disease can be detected that vary in risk. As in the discovery and characterization of risk subgroups, the main risk factors are made explicit, the proposed methodology has high potential for patient screening and early detection of patient groups at risk. This application combined *prediction* of coronary heart disease with subgroup discovery.

Gene expression

A growing number of applications is in the area of gene expression. Features of genes obtained by microarrays are used to predict the properties of organisms. A practical purpose is to identify, for example, disease markers from gene expression data (Gamberger *et al.*, 2004).

6.7 Human resource planning

Two examples involve the prediction of the personnel of an organization from the properties of the personnel and of the organization (career tracks, available and required education). Adriaans & Zantinge (1996) describe a system that learns to predict career choices (e.g. applying for a new job or training programme) by airline personnel. This is important to enable planning of personnel intake and training capacity. Data consisted of career information and choices of personnel. The resulting system did allow prediction of career choices and can be used to optimize human resource management decisions by simulating the number of qualified personnel and trainees. This was used to control the acquisition of new personnel and the planning of courses. The methods used were rule learning and genetic algorithms. Learning was done once. In one case the result was embedded in a larger human resource planning simulation.

6.8 Culture and entertainment

Learning to compose or perform music

The production (or reproduction) of art can be supported by a system that applies general design principles to input provided by the user. Although this type of application is mostly motivated by a scientific interest in music and learning methods, some systems provide prototypes of applications. Several systems (Widmer *et al*., 2003; Widmer & Tobudic, 2003; Arcos & Lopez de Mantaras, 2002) learn to play existing (written) music in ways that correspond to different expressions.

Personalized entertainment advice

These applications provide the user with advice for the choice of some form of entertainment, a TV programme, a movie or a theatre performance. Input for the learning process is information about choices that the user made in earlier situations and patterns in the choices of other users to select or rank attractive possibilities. This amounts to a prediction task. The resulting model is embedded into a system with general information about these choices. An example is a system designed by Smyth & Cotter (2000). This stores a person's selected TV programmes and learns to recognize and recommend new programmes on the basis of features such as start/end time, title, actor and channel. The system was designed to be part of a handheld system that includes remote control and Internet access. This can be done with various methods. The resulting system can, for example, indicate to a user which of the available TV programmes he is likely to find interesting. Another typical example is personalization of newspapers on the basis of user-selected articles (e.g. Van der Putten, 1999).

6.9 Logistics/transport

Controlling vehicles

More direct control of vehicles is demonstrated in for example Sammut *et al*. (1992) and Camacho (1998) who used records of expert pilots to learn to predict their actions as a function of conditions in simulated flying. Training data were information from the control panel of the simulator and pilot actions for specific manoeuvres.

Detecting risky traffic situations

A relatively small proportion of traffic situations gives rise to an accident or to a (registered) dangerous situation. Records of this can be used to learn to detect general conditions by which risky situations can be recognized. Examples of this type of application were described, for example, for road situations (using information about the road situation and traffic flow) in Džeroski *et al*. (1998) and Flach *et al*. (2003) and air traffic control (using positions and movements of airplanes) in Kodratoff & Vrain (1997).

Transport security

Michalski *et al*. (1998) describe an application for predicting whether an X-ray image contains a picture of a blasting cap. This is aimed at automating the task of scanning luggage and cargo for aircraft and ships.

6.10 Government and public administration

Performance evaluation

In the government and public administration sector, most of the applications deal with evaluation of performance. The examples of this task include examination of socio-economic development of countries (Krisper *et al.*, 1989), performance evaluation of enterprizes (Olave *et al.,* 1988) and the evaluation of performance in government application units (Pe'ear, 2002). Possible factors that influence the performance are examined and rules (usually in a form of decision trees) for classification are induced. In this way, machine learning can help in discovering which of the examined factors substantially influence the performance. This can be very useful for planning improvements, as some analyzes find surprisingly large differences in efficiency: among 58 administrative units analysed by Pe'ear (2002) the ratio was even 1:10.

6.11 Security

The Dutch police used DM to discover patterns in the features of burglaries (Lopez & Van der Wal, 2002). Data about offenders, victims and the situation are analysed to predict the risk of burglaries and to predict the features of burglars in the form of offender profiles.

7 Discussion

The application tasks cover a wide range of application fields and of tasks within these fields. There is no reason to think that the tasks above exhaust possibilities. In fact, if we look at our collection and we consider the years in which publications were published then it is clear that both the number and the range of tasks increase.

The total number of documented applications is not very large. From informal contacts we suspect that the actual number of applications in Europe will be at least ten times higher than the 40 European applications that we review in this paper. This number is still quite small. However, it seems that there is much more potential. The application tasks that we described above are mostly based on small numbers of documented applications, but these tasks are quite general. This suggests a large number of possible applications.

7.1 Distribution of applications over application fields

The overview of applications shows a lack of balance between different application fields. Many applications are in medicine and health, production industry and marketing. Few applications are in government/ administration and information technology. Most applications here are adaptive spam filters and document classification. One of the possible reasons for the lack of penetration in government and administration may be legal responsibility. Another reason might be differences in background. ML applications require a special combination of expertise in ML, databases, statistics and business processes. However, this does not explain the situation with few applications of DM in information technology. This can be due to the lack of a tradition of empirical research in information technology.

In this paper we have given an overview of the application of ML to a wide range of generic problems from different application fields. We have defined the main ML tasks and showed how practical problems were mapped to ML tasks to allow methods and tools to be applied. Given a mapping to a ML task, an appropriate type for the data and the model is selected. Given these decisions and the type of the data, several methods are usually applicable from which one (or a combination) must be selected. Some methods then also involve optimization of parameter settings.

Not all application fields are equally represented. We expect that this is due to two factors. One is that many applications are not publicly available and the other is that indeed the technology has not penetrated well in some areas. One of the goals of this paper is to clarify which tasks are addressed by ML methods by making these tasks explicit.

7.2 Mapping problems to ML methods

Deciding whether a practical problem can be solved with ML/DM needs an answer to the following main questions. (1) Which ML task matches the problem? (2) Which is the right type for the data? (3) Which is the right type for the model? (4) Are there enough data available, relative to the uncertainty in the domain, to find a model? In this paper we focus on the first question. Initially, matching a problem to a ML task is a matter of judgement. Preliminary analysis may point out that the most direct match is not likely to lead to a successful solution. Adequate data may not be available at all or experts expect that the amount of uncertainty will not allow a model to be found. To assess the probability of success, the content of the domain is relevant. For example, it is likely that in different areas, similar factors will determine customer loyalty and existing models can give an indication of what we may expect from a new area. For the choice of a ML task *per se* the content of the domain is not relevant. If the initial match is likely to fail, a possibility is to redefine the problem as a different ML task.

7.3 Implications for ML and DM

From an application viewpoint, one of the tasks for ML researchers and developers is to construct methods and tools for all combinations of ML tasks, data types and model types, or even to automate the selection of data and model types and the transformation of the 'raw' problem into the optimal data and model types. The range of model types and data types is not closed and new types are being proposed in each conference. It is remarkable, though, that relatively little emphasis has been placed on methods for characterization and for optimization. In the applications we reviewed, tasks that fit characterization and optimization tasks were often solved by prediction methods.

Note: This term and the idea for this concept was suggest by Pieter Adriaans who formulated the need for innovative application paradigms rather than more technology.

Appendix A: Selected applications of ML

Predicting customer loyalty

The purpose of this application is to understand, explain and predict customer behaviour, in particular loyalty to and satisfaction with a brand of cars and with a dealer. This understanding has a number of applications in marketing and general management of car sales. Decision tree learning was used instead of statistical methods, to construct a model of customer loyalty from data describing properties of customers, including their buying records. The resulting model showed better results than the regression models.

Reference	Vanhoof *et al.* (1997)
Client	Car company
Application field	Marketing, car industry
Learning task	Prediction
Input of learning	Data describing properties of customers, including their buying records
Form of input	Discretised numerical variables
Form of the result	Rules
Method	Decision tree learning
Status	Reported to client

Automating control of a grinding process

The optimal settings of control parameters for grinding depend on a number of properties of the objects to be ground, such as the material, thickness and shape. Setting the controls is normally done by humans on the basis of experience. The purpose of this application was to learn how to set the controls and possibly use this to automate control.

Reference	Filipič & Junkar (2000)
Client	Research Laboratory at the Faculty of Mechanical Engineering, University of Ljubljana
Application field	Process industry
Learning task	Control
Input of learning	Control parameters and properties of: objects that are to be ground; of success and failure
Form of input	Numerical and nominal
Form of the result	Decision tree with intervals on numerical variables
Method	Decision tree learning
Status	Demo

Credit-worthiness scoring

Evaluating the credit worthiness of a company as client for a bank from data that describe the financial state of the company and general information about its activities and markets.

Reference	Paass & Kindermann (1998)
Client	Unknown
Application field	Banking/insurance/finance
Learning task	Prediction
Input of learning	Properties of clients requesting a loan with (positive versus negative) expected profit
Form of input	Mixed numerical and nominal
Form of the result	Hybrid models based on Bayesian model
Method	Bayesian
Status	Unknown

Modelling Website navigation

Identifying user request patterns to improve Website design. Frequent patterns in Websites can be used to organize the pages and navigation information such that users can avoid redundant accesses to pages.

Reference	Spiliopoulou & Pohle (2001)
Client	German government
Application field	Administration and government
Learning task	Frequent groups (cases are sequences)
Input of learning	Records (logs) of page requests of users of a Website with information about education
Form of input	User logfiles
Form of the result	Weighted links
Method	*Ad hoc*
Status	Used to manually improve navigation Website

Explaining energy use and other parameters of the potash mining process

The purpose of this application was to reduce the energy consumption of potash drying by a rotating kiln. The data included exit product temperature, exit gas temperature, oil firing position (energy use) and centrifuge current. Analyses allowed a better understanding of the relation between process parameters and energy use.

Reference	Knowledge Process Solutions (2002)
Client	Cleveland Potash
Application field	Production industry, mining
Learning task	Characterisation
Input of learning	Records of control settings and input of process and parameters of output
Form of input	Unknown
Form of the result	Rules
Method	Rule learning
Status	Completed analysis of performance of rotary kiln driers in an ore refinery; reduction of energy consumption. Knowledge discovery was audited and drier operating conditions were improved.

Assessment of public enterprises

Learn a decision tree that will reproduce the weighted classification formula usually used for this purpose with the objective to increase understandability.

Reference	Olave *et al.* (1988)
Client	International Centre for Public Enterprises (ICPE)
Application field	Administration and government
Learning task	Prediction
Input of learning	Records of evaluated enterprises
Form of input	Numerical
Form of the result	Decision tree
Method	Top-down induction of decision trees
Status	Reported to the client

Explanation of customer loyalty/defection

This information is then used for various decisions in marketing, pricing and product development of insurances.

Reference	Koenig (1998)
Client	Centraal Beheer insurance
Application field	Marketing, finance
Learning task	Explanation
Input of learning	Client data and claim history
Form of input	Mixed numerical and nominal
Form of the result	Rules derived from decision trees
Method	Decision tree learning
Status	Presented to clients marketing department

Discovering unknown properties of chemicals

For research and development of new medicine and also for other chemicals it is important to know in advance if the chemical will be toxic. One of the purposes of these studies is to construct a model that can predict from their chemical composition and structure if new chemicals will be toxic. This can be used to avoid expensive testing or even chemical engineering to construct it.

Reference	Srinivasan & King (1999)
Client	—
Application field	Science; pharmacy
Learning task	Prediction
Input of learning	Descriptions of the chemical structures of chemicals and their toxicity

Form of input	Numerical, symbolic and relational
Form of the result	Rules derived from decision trees
Method	Inductive logic programming
Status	Demo

Predicting properties of offspring, to find good crossings

This is used to select promising crossing experiments embedded in an interactive system for planning experiments and storing the results and also comprehensible knowledge for predicting properties of offspring.

Reference	Kraakman (1998)
Client	Department of Plant Breeding, Agricultural University Wageningen
Application field	Production industry (agriculture)
Learning task	Prediction
Input of learning	Descriptions of plants with descriptions of their offspring
Form of input	Numerical/nominal
Form of the result	Decision tree
Method	Decision tree learning
Status	Demo

Modelling population dynamics of red deer

Predicting the development of a population of red deer from population size, number of off-shoots, habitat observations and properties.

Reference	Debeljak *et al.* (2001)
Client	Science/biology
Application field	Science
Learning task	Prediction
Input of learning	Population size, number of off-shoots, habitat observations and properties
Form of input	Mixed numerical/nominal values
Form of the result	Regression tree
Method	Regression tree learning
Status	Demo

Modelling the effects of environmental conditions on apparent photosynthesis of Stipa bromoidesf-stipa

For the scientific purposes a model was constructed of the effect of the number of factors on the rate of photosynthesis of grass.

Reference	Dalaka *et al.* (2000)
Client	Science/biology
Application field	Science/biology
Learning task	Prediction (or explanation?)
Input of learning	Measured properties of plants and environmental conditions
Form of input	Numerical
Form of the result	Regression tree
Method	Regression tree with RReliefF as an attribute estimator
Status	Evaluated by experts

Modelling plankton growth in lakes

Predicting the growth of algae in lakes from weather conditions, water conditions including flow.

Reference	Todorovski *et al.* (1998)
Client	Science/biology

Application field	Science/biology
Learning task	Prediction (or explanation)
Input of learning	Properties of water and growth of plankton in lakes
Form of input	Numerical and nominal; background knowledge
Form of the result	Regression tree and equations
Method	Equation discovery (Lagrange)
Status	Demo

Hierarchical structure of classes of plants

Constructing a hierarchical category structure of plants from their descriptions. The resulting structure was quite acceptable to human experts.

Reference	Alberdi & Sleeman (1997)
Client	Science/biology/botany
Application field	Science/biology
Learning task	Clustering
Input of learning	Descriptions of plants
Form of input	Nominal
Form of the result	Cluster hierarchy
Method	Clustering
Status	Results were evaluated by experts

Predicting fresh concrete properties

Predicting properties of concrete from mix proportions and data about input materials. Formalizing knowledge in order to reduce the amount of expensive and time-consuming laboratory experiments.

Reference	Urbančič *et al.* (1998)
Client	Institute for Testing and Research in Materials and Structures, Ljubljana
Application field	Process industry
Learning task	Prediction
Input of learning	Mix proportions and data about input materials
Form of input	Numerical and nominal
Form of the result	Decision trees
Method	Decision tree and Bayesian learning
Status	Reported to the client

Diterpene structure Elucidation from ^{13}C NMR spectra

Diterpenes are organic compounds of low molecular weight with a skeleton of 20 carbon atoms. They are of significant chemical and commercial interest because of their use as lead compounds in the search for new pharmaceutical effectors. The interpretation of diterpene ^{13}C NMR spectra normally requires specialists. Associations between peak patterns and chemical structure are discovered from a database.

Reference	Džeroski *et al.* (1998)
Client	Unknown
Application field	Production industry, chemistry
Learning task	Prediction
Input of learning	Diterpene spectra with molecule structures, background knowledge on structures
Form of input	Numerical and nominal
Form of the result	Inductive logic programming rules

Method	Decision tree learning and Bayesian learning, first-order inductive learning, relational instance-based learning, induction of logical decision trees and inductive constraint logic programming
Status	Performance close to the one of domain experts is achieved, sufficient for practical use.

Improving energy consumption of potash-drying

Parameters of a potash-drying process (materials, quantities, temperature, humidity and others) were related to energy consumption (heating). The patterns that were detected enabled lower temperatures thereby saving energy.

Reference	Knowledge Process Software (2004b)
Client	Cleveland Potash
Application field	Process industry; power/energy production
Learning task	Prediction (or explanation)
Input for learning	Conditions in environment and control settings of power plant
Form of input	Numerical and nominal
Form of the result	Rules
Method	Rule learning
Status	Applied and proven to produce large benefits.

Improving energy consumption of brewing

Parameters of brewing process (temperature, load and compression of brewing machine) were related to energy consumption (heating). The patterns that were detected enabled lower temperatures thereby saving energy.

Reference	Knowledge Process Software (2004a)
Client	Bass Burton Brewery
Application field	Process industry; power/energy production
Learning task	Prediction/control
Input of learning	Conditions in environment and control settings of brewing plant
Form of input	Numerical and nominal
Form of the result	Rules
Method	Rule learning
Status	Applied and proven to produce large benefits.

Modelling network load patterns

Rules were discovered that explain the occurrence of network problems from properties of the network and its structure, to optimize network performance.

Reference	Knobbe *et al.* (1998)
Client	Syllogic/Perot Systems
Application field	Services; information technology
Learning task	Explanation
Input of learning	Description of computer network structure, input and system performance records
Form of input	Numerical and nominal
Form of the result	(First-order) logical rules
Method	Rule learning; inductive logic programming
Status	Demonstrator; lead to followup projects.

Constructing spam filters

Spam is a significant practical problem. A system can be trained to recognize spam messages and delete them or store them at a special place for later scanning.

Reference	Androtsopoulos *et al.* (2000)
Client	Unknown
Application field	Information technology
Learning task	Prediction
Input of learning	E-mails transformed into word vectors and classified as spam/non-spam
Form of input	Word vectors with labels
Form of the result	Bayesian model based on bag-of-words
Method	Cost-sensitive Bayesian method
Status	Demonstrator

Recognize intruders in computer networks

A collection of TCP/IP transactions was classified as legal/illegal. Illegal were intruders trying to get access to the network or to perform illegal operations. The system learned to classify transactions as legal/illegal.

Reference	Mischiati & Neri (2000)
Client	—
Application field	Information technology
Learning task	Prediction
Input of learning	TCP/IP transactions in network classified as intruders/non-intruders
Form of input	Mixed numerical/nominal
Form of the result	Rules
Method	Genetic algorithm for rule learning
Status	Demonstrator

Prediction of treatment for ischaemic pain

Construct a model to predict the effect of different treatments of ischaemic pain. The result can then be used to support treatment of ischaemic pain.

Reference	Kukar *et al.* (1999)
Client	—
Application field	Social and medical care
Learning task	Prediction (or control)
Input of learning	Descriptions of patients with ischaemic pain, treatment and effect
Form of input	Mixed numerical/nominal
Form of the result	Decision tree
Method	Decision tree learning
Status	Demo with positive evaluation by experts

Prognosis of prostate cancer

Learning prognostic models for prostate cancer recurrence, where the sole prediction of probability of event (and not its probability dependency on time) is of interest.

Reference	Zupan *et al.* (2000)
Client	Memorial Sloan-Kettering Cancer Center, New York
Application field	Social and medical care
Learning task	Prediction
Input of learning	Descriptions of patients with prostate cancer, treatment and process
Form of input	Mixed numerical/nominal
Form of the result	Decision tree
Method	Decision tree learning
Status	Implemented on Palm top

Prognosis of femoral neck fracture recovery

A model is learned for prediction of recovery from features derived from physical examination and the conditions under which trauma occurred.

Reference	Kukar *et al.* (1996)
Client	—
Application field	Social and medical care
Learning task	Prediction
Input of learning	Descriptions of patients with femoral neck fracture, treatment and recovery
Form of input	Mixed numerical/nominal
Form of the result	Decision tree
Method	Decision tree learning
Status	Demo

Diagnosis of head injury

Subgroups of patients and critical values in variables were discovered and discussed with experts. Verified values falling outside threshold limits were analysed by insult grade and duration. using logistic regression. The most significant predictors of mortality in this patient set were duration of hypotensive, pyrexic and hypoxemic insults.

Reference	Andrews *et al.* (2002)
Client	—
Application field	Social and medical care
Learning task	Explanation
Input of learning	Patient data with later diagnosis of head injury with outcome after 12 months
Form of input	Mixed numerical/nominal
Form of the result	Decision tree and logistic regression
Method	Decision tree learning
Status	Published in medical literature and used by physicians.

Prediction of chemical carcinogenicity

For the development of new chemicals it is important to know in advance whether they can cause cancer. It is very time consuming and expensive to perform experiments for the development of a Structure Activity Relationship (SAR) model.

Reference	King *et al.* (1995).
Client	—
Application field	Social and medical care
Learning task	Prediction
Input of learning	Properties of chemicals and carcinogenity (from Carcinogenic Potency Database)
Form of input	Mixed numerical/nominal
Form of the result	Inductive logic programming rules
Method	Several inductive logic programming methods
Status	Demo

Prediction of biodegradation pathways of compound chemicals

The majority of organic matter released into the environment is degraded by microorganisms. This does not happen in a single step, but in a sequence of reactions, which form a biodegradation pathway. The purpose of this study was to explain the various steps in the degradation process.

Reference	Blockeel *et al.* (2004)
Client	—
Application field	Environmental science

Learning task	Prediction/explanation
Input of learning	Structures of chemicals, numerical and non-numerical data of degradation paths
Form of input	Numerical/nominal
Form of the result	Inductive logic programming rules
Method	Several inductive logic programming methods
Status	—

Prediction of career decisions

From data on career choices of airline personnel a model was learned to enable prediction of future career choices and thereby enable prediction of needs for new personnel and training.

Reference	Adriaans & Zantinge (1996)
Client	KLM Royal Dutch Airlines
Application field	Transport, human resources
Learning task	Prediction
Input of learning	Data about personnel (age, education, experience, income)
Form of input	Mixed numerical/nominal
Form of the result	Rules
Method	Evolutionary computing
Status	Used in practice

Detecting dangerous road sections

Where a problem has occurred (critical sections) from sensor data and records of road accidents.

Reference	Džeroski *et al.* (1998)
Client	City of Barcelona
Application field	Transport
Learning task	Prediction
Input of learning	Descriptions of road sections and traffic flow data (from sensors) with and without problems; relational background knowledge on the road network
Form of input	Nominal, numerical, structural
Form of the result	Horn clauses
Method	Inductive logic programming
Status	Demo

Intensive care monitoring

A number of measures are taken with varying intervals of patients in the intensive care units of hospitals. Alarming conditions are recognized by computer systems that then send a warning to nurses on guard. Current systems are not able to recognize a number of important conditions or rather, they recognize them later than necessary. A critical condition can often be recognized early by noting changes in variables that cannot be explained from day–night rhythm, medication, etc.

Reference	Morik *et al.* (2000)
Client	Surgical Department, City Hospital Dortmund
Application field	Medical care
Learning task	Prediction
Input of learning	Descriptions of histories of intensive care patients with data on patient and treatment
Form of input	Nominal, numerical, structural, prior knowledge
Form of the result	Hybrid form, horn clauses
Method	Several methods from ML; inductive logic programming
Status	Demo

Estimation of socio-economic development of countries

Find and examine the possible factors that influence the socio-economic development of a country. Develop a criterion for estimation of socio-economic development of a country.

Reference	Krisper *et al.* (1989)
Client	Institute for Economic Research, Ljubljana
Application field	Administration and government, political science
Learning task	Explanation
Input of learning	Economic and cultural data about countries with indicator of socio-economic development
Form of input	Mixed numerical and nominal
Form of the result	Decision tree
Method	Top-down induction of decision trees
Status	Reported to the client

Recommending TV programmes

From user choices of TV programmes and other features (weekday, time, season) a model is constructed of user interests. This model is used to recommend TV channels and programmes.

Reference	Smyth & Cotter (2000)
Client	General public
Application field	Culture and entertainment
Learning task	Prediction
Input of learning	Available and preferred TV programs
Form of input	Mixed numerical and nominal
Form of the result	Decision tree
Method	Top-down induction of decision trees
Status	Demo under development into product

Training an advertisement filter

Many Webpages contain advertisements that users may not want to download and watch. One approach is to train a system by pointing out advertisements so that the system can learn to recognize and filter advertisements. A team at University College Dublin realized this in a system, using ML methods.

Reference	`http://www.cs.ucd.ie/smi/`
Client	—
Application field	Services
Learning task	Prediction
Input of learning	Advertisements on the Web
Form of input	Mixed numerical and nominal
Form of the result	Decision tree
Method	Decision tree learning
Status	Available via the Internet

Expressive music

A system was trained to modify music to make it sound expressive.

Reference	Arcos & Lopez de Mantaras (2001)
Client	—
Application field	Culture and entertainment
Learning task	Prediction (or control)
Input of learning	Written and played music with expression label

Form of input	Mixed numerical and nominal
Form of the result	Case base
Method	CBR
Status	—

Learning to control the quality of the rolling emulsion in a steel rolling mill (the Sendzimir rolling mill)

The quality of the cold strip or cold sheet is usually acceptable if the quality of the rolling emulsion is high. For this purpose, the quality of the rolling emulsion is systematically controlled and actions are taken according to the observed parameters.

Reference	Gams *et al.* (1997)
Client	Iron and Steel Works Jesenice
Application field	Process industry
Learning task	Control
Input of learning	Mitigating number of oil, presence of Fe in oil, presence of Fe in sediment, concentration of oil, addition of oil, addition of basin oil and presence of inactive oil
Form of input	Numerical and nominal
Form of the result	Decision tree with intervals on numerical variables
Method	Decision tree learning
Status	Regularly used in Iron and Steel Works Jesenice

Controlling an airplane

The conditions of the airplane, the information on the control panel and the actions were recorded of pilots who were flying an airplane in a single flight plan. These were divided into situation–action pairs. ML was used to discover which action in general was appropriate for a given condition. The resulting system was able to follow the same flight plan even though conditions vary. The study by Sammut *et al*. was done with a Cessna simulation and that by Camacho with an F-16 jet fighter.

Reference	Sammut *et al.* (1992); Camacho (1998)
Client	—
Application field	Transport
Learning task	Control
Input of learning	Traces of positions of airplane with actions of pilot
Form of input	Mixed
Form of the result	Decision tree with intervals on numerical variables
Method	Decision tree learning
Status	Demo

Predicting customer reaction to direct mail

Reactions of customers to direct mail were predicted from data about the customer.

Reference	Van der Putten (1999)
Client	Dutch Insurance company
Application field	Finance
Learning task	Prediction
Input of learning	Customer data and accept/reject offer
Client	Dutch insurance company
Form of input	Mixed
Form of the result	Rough set rules
Method	Rough set learning (ProbRough)
Status	Delivered and paid

Security management of wind power plants

A detailed simulation of supply and consumption of electric power on Crete was used to generate data on the environment, the state of the power system and its transition to an insecure state. ML produced a model that can predict under which conditions the system enters an insecure state.

Reference	Vasconcelos & Pecas Lopes (1999)
Client	—
Application field	Process industry
Learning task	Prediction
Input of learning	Simulated disturbances under varying weather conditions
Form of input	Mixed
Form of the result	Regression trees; fuzzy rules
Method	Kernel regression trees; fuzzy modelling; decision tree learning
Status	Model is finished; will be integrated in electric power control system in Crete.

Monitoring and predictive maintenance of power plants

Measurements from the power plant are interpreted to detect conditions under which security is at risk.

Reference	Vasconcelos & Lopes (1999); Matos *et al.* (1999)
Client	—
Application field	Process industry
Learning task	Prediction
Input of learning	Sensor measurements on equipment and deviation from optimal condition
Form of input	Mixed
Form of the result	—
Method	Reinforcement learning
Status	—

Finite mesh design

Designing a finite mesh that enables analysis of an object is used for estimating properties of objects such as behaviour under stress, temperature changes, etc. The granularity and the components of the mesh must be estimated and the mesh configured.

Reference	Dolšak *et al.* (1998)
Client	—
Application field	Process industry
Learning task	Prediction (not variable value)
Input of learning	Objects with mesh designs
Form of input	Relational descriptions
Form of the result	Relational rules
Method	Inductive logic programming (several)
Status	Evaluated

Customer profiling

Characterize profiles of market segments.

Reference	Van der Putten (1999)
Client	De Telegraaf (publishing company)
Application field	Services
Learning task	Characterisation
Input of learning	Properties of persons, including the media they consume
Form of input	Mixed

Form of the result	Type of clustering
Method	Mixture
Status	Used

Predicting gas consumption

The local consumption of gas is predicted 24 hours ahead, from features such as weather forecasts, temperature, holidays, etc.

Reference	Klema & Kout (1999)
Client	German local gas production companies
Application field	Government/energy
Learning task	Prediction
Input of learning	—
Form of input	Mixed
Form of the result	Hybrid of neural networks, CBR, regression
Method	Hybrid of neural networks, CBR, regression
Status	Used

Predicting the mutagenicity of chemical compounds

Components and structural properties of chemicals determine whether the chemical can generate mutations in cells. Linear regression of derived properties works, but the nonlinear structural models are more comprehensible and relates molecular structure directly to mutagenicity.

Reference	Bratko *et al.* (1998)
Client	—
Application field	Biochemistry
Learning task	Prediction
Input of learning	Molecular structures and mutagenicity of a number of chemicals
Form of input	Structured descriptions
Form of the result	Relational model
Method	Inductive logic programming, Progol
Status	Finished

Controlling electrical discharge machining

Classification of dielectric fluids used in electrical discharge machining.

Reference	Bratko *et al.* (1998)
Client	Department of Mechanical Engineering, University of Ljubljana
Application field	Production industry
Learning task	Prediction
Input of learning	Control variables and result
Form of input	Structured descriptions
Form of the result	Relational model
Method	Inductive Logic Programming, Progol
Status	Used

Supporting troubleshooting of engines

Finding the cause of faults in the engines of airplanes is improved by learning from past cases. The diagnostic system is integrated with systems that display information about the structure and function of the engine. Reduces the time needed for diagnosis to 50%.

Reference	Manago & Auriol (1997)
Client	Cfm-international

Application field	Transport
Learning task	Prediction
Input of learning	Descriptions of engines and complaints with cause
Form of input	Attribute–value pairs
Form of the result	Decision trees combined with cases for case-based reasoning
Method	Decision tree learning
Status	Used

Supporting diagnosis of robot axis

Finding the cause of faults in robots is improved by learning from past cases. The diagnostic system is integrated with systems that display information about the structure and function of the engine. Reduces the time needed for highly specialized engineers.

Reference	Manago & Auriol (1997)
Client	SEPRO Robotique
Application field	Production industry
Learning task	Prediction
Input of learning	Descriptions of engines and complaints with cause
Form of input	Mixed numerical/nominal
Form of the result	Decision trees combined with cases for case-based reasoning
Method	Decision tree learning
Status	Used

Designing anti-friction bearing arrangements in drives

Completing a partial design for a derive by selecting components (axis, rings, seals, lubricant) and specifying parameter values.

Reference	Moczulski (1999)
Client	Department of mechanical engineering, Technical University of Silesia, Poland
Application field	Production industry; mechanical engineering
Learning task	Prediction
Input of learning	Successful designs
Form of input	Mixed linear and nominal
Form of the result	Rules
Method	*Ad hoc* method and tool
Status	Unknown

Recognizing blasting caps in X-ray images of luggage

X-ray images were abstracted into geometric features derived from a model of X-ray impact on blasting caps. Derived features were used for learning.

Reference	Michalski *et al.* (1997)
Client	—
Application field	Transportation, Government
Learning task	Prediction
Input of learning	Descriptions of X-ray images with blasting cap
Form of input	Mixed numeric/non-numeric
Form of the result	Rules
Method	AQ algorithm
Status	Demo

Recognizing actions in video image sequences

Objects and actors are recognized in images. Motion sequences are used to detect motion of objects and the relative position of actors. This is combined to detect movements.

Reference	Michalski *et al.* (1997)
Client	—
Application field	Transportation, Government
Learning task	Prediction
Input of learning	Descriptions of objects with motion
Form of input	Mixed numeric/non-numeric features
Form of the result	Rules
Method	AQ algorithm
Status	Demo

Learning to play music

How a note is played is learned from examples of notes and performance on piano. Prior knowledge about play is refined into rules for playing.

Reference	Widmer & Tobudic *et al.* (2003); Widmer, Dixon *et al.* (2003)
Client	—
Application field	Culture and entertainment
Learning task	Prediction/control
Input of learning	Notes with play
Form of input	Mixed numeric/non-numeric features
Form of the result	Rules
Method	*Ad hoc* method
Status	Demo

Learning to find interesting information on the Internet

Discovering characteristics of interesting documents and using these to modify user queries and to actively search for documents.

Reference	Armstrong *et al.* (1997)
Client	—
Application field	Information technology
Learning task	Classification
Input of learning	Documents rated as interesting
Form of input	Documents
Form of the result	Naive Bayesian classifier
Method	Naive Bayes
Status	Demo
Comment	Uses anomaly detection

Recommending documents

From documents that are labelled as relevant or irrelevant, user profiles are constructed that predict relevance of new documents for users. This application involves the use of prior knowledge about relevance.

Reference	Middleton *et al.* (2004)
Client	—
Application field	Knowledge management
Learning task	Prediction
Input of learning	Documents
Form of input	Nominal
Form of the result	Probabilistic model
Method	Nearest-neighbour based
Status	Demo

Learning to control cranes

Learns to control crane manoeuvres from records of manoeuvres by human experts.

Reference	Bratko *et al.* (1997); Bratko & Šuc (2002)
Client	—
Application field	Transportation
Learning task	Control
Input of learning	Expert traces of crane control
Form of input	Documents
Form of the result	Decision tree
Method	Decision tree learning
Status	Demo

Learning air traffic control

Learns to prevent dangerous situations in air traffic control.

Reference	Kodratoff & Vrain (1997)
Client	—
Application field	Transportation
Learning task	Prediction
Input of learning	Descriptions of air plane configurations with label dangerous/safe
Form of input	Relations and features
Form of the result	Relational rules
Method	Inductive logic programming
Status	Demo

Localization of primary tumor

Medical treatment of patients with metastases is much more successful if the location of the primary tumor in the body of the patient is known. The task is to determine one of 22 possible locations on the basis of age, sex, histological type of carcinoma, degree of differentiation and 13 possible locations of discovered metastases.

Reference	Kononenko (2001)
Client	Institute of Oncology, Ljubljana
Application field	Medicine
Learning task	Prediction
Input of learning	Descriptions of patients: age, sex, histological type of carcinoma, degree of differentiation and 13 possible locations of discovered metastases.
Form of input	Feature-value
Form of the result	Several, naive Bayes
Method	Naive Bayes
Status	Demo

Prediction of recurrence of breast cancer

Among patients with removed breast cancer the disease recurs in five years after the operation in about 20% of the cases. For better treatment it is important to be able to predict the possibility of recurrence.

Reference	Kononenko (2001)
Client	Institute of Oncology, Ljubljana
Application field	Medicine
Learning task	Prediction

Input of learning	Descriptions of patients: age, sex, size and location of the tumor and data bout lymphatic nodes
Form of input	Feature-value
Form of the result	Several, naive Bayes
Method	Naive Bayes
Status	Demo

Classifying meningoencephalitis

Patients with certain symptoms may have meningoencephalitis, which can have a viral or a bacterial cause. From a collection of patient data, rules were discovered to classify patients.

Reference	Tsumoto (1995, 1999)
Client	—
Application field	Medicine
Learning task	Prediction
Input of learning	Sex, age, white blood count and other medical tests
Form of input	Nominal
Form of the result	Rules
Method	Rule learning
Status	Demo; discovered new domain knowledge

Explaining traffic accident rates

A large set of data about roads, locations and traffic accidents was analysed in several ways with the main goal of finding the conditions under which accidents are likely to occur.

Reference	Flach *et al.* (2003)
Client	Hampshire County Council (UK)
Application field	Transportation
Learning task	Explanation/prediction
Input of learning	Data about locations with frequency of accidents over time (years)
Form of input	Mixture
Form of the result	Rules, clusters
Method	Rule learning, clustering, association rules, subgroup discovery, clusters of time series
Status	Proof of concept; motivated successor project

Predicting chemical waste output

The purpose was to optimize control of the activated sludge process, the main biological technology applied to waste water treatment plants (WWTPs). This paper, proposes an integrated and distributed supervisory multi-level architecture for the supervision of WWTPs, that overcomes some of the main troubles of classical control techniques and those of knowledge-based systems applied to real-world systems.

Reference	Sanchez-Marre *et al.* (1996)
Client	Catalonian government
Application field	Production industry
Learning task	Prediction/explanation
Input of learning	Control setting and measurements of sludge processes
Form of input	Mixture
Form of the result	Case base
Method	CBR
Status	Demo

Modelling the radon concentration in soil gas for earthquake prediction

Predicting the concentration of radon in soil gas in dependence of meteorological parameters. Discrepancies between measured and predicted values of radon indicate likely seismic activity.

Reference	Džeroski *et al.* (2003)
Client	Science/seismology
Application field	Science
Learning task	Prediction
Input of learning	The daily radon concentration average with: average daily barometric pressure, average daily air temperature, average daily soil temperature, and daily amount of rainfall and derived variables
Form of input	Numerical values
Form of the result	Regression tree
Method	Regression tree learning
Status	Demo

Coronary heart disease risk group detection

Active mining of patient records aimed at discovering patient groups at high risk for coronary heart disease.

Reference	Gamberger *et al.* (2003)
Client	Institute for Cardiovascular Prevention and Rehabilitation, Zagreb
Application field	Health care
Learning task	Explanation
Input of learning	Patient records including anamnesis attributes, laboratory and ECG at rest test results
Form of input	Mixed
Form of the result	Rules describing subgroups
Method	Subgroup discovery
Status	Demo

Modelling the population dynamics of brown bears in Slovenia

Predicting the development of a population of brown bears in terms of the size of the population, as well as its habitat.

Reference	Jerina *et al.* (2003)
Client	—
Application field	Science; biology
Learning task	Prediction
Input of learning	Presence/absence data, data from GIS (land cover data, forest inventory data, settlements map, road map, digital elevation model)
Form of input	Mixed
Form of the result	Decision tree
Method	Regression tree learning
Status	Demo

Patterns in burglaries

Data about burglaries were used to discover patterns in features of offenders (personal, socio-economic background, procedure of intrusion, etc.), victims (personal, socio-economic background, lifestyle, goods available) and situation (place, time, building type).

Reference	Lopez & van der Wal (2002)
Client	Dutch Police Institute

Application field	Government
Learning task	Prediction/explanation
Input of learning	Features of burglaries, offenders, victims, situations
Form of input	Mixed
Form of the result	Decision tree, discrimination function, regression
Method	Decision tree learning, discriminant analysis, regression analysis
Status	Demo

References

Adriaans, P and Zantinge, D, 1996, *Data Mining*. Reading, MA: Addison-Wesley.

Alberdi, E and Sleeman, D, 1997, ReTAX: a step in the automation of taxonomic revision. *Artificial Intelligence* **9**, 257–279.

Andrews, PJD, Sleeman, D, Satham, PFX, McQuatt, AM, Corruble, V, Jones, PA, Howells, T and Macmillan, CSA, 2002, Forecasting recovery after traumatic brain injury using intelligent data analysis of admission variables and time series physiological data—a comparison with logistic regression. *Journal of Neurosurgery* **97**, 326–336.

Androtsopoulos, I, Koutsias, J, Chandrinos, V, Pailiouras, G and Spyropoulos, CD, 2000, An evaluation of naïve Bayesian anti-spam filtering. In *Proceedings of the Conference on Machine Learning in the New Information Age*.

Arcos, JL and Lopez de Mantaras, R, 2001, An interactive case-based reasoning approach for generating expressive music. *Applied Intelligence* **14**, 115–129.

Blockeel, H, Džeroski, S, Kompare, B, Kramer, S, Pfahringer, B and van Laer, W, 2004, *Applied Artificial Intelligence* **18**, 157–181.

Bratko, I, 1992, Applications of Machine Learning: towards knowledge synthesis. In *Proceedings of the International Conference on 5th Generation Computer Systems*. ICOT.

Bratko, I and Šuc, D, 2002, Using machine learning to understand operator's skill. In Hendtlass, T and Ali, M (eds) *Developments in Applied Artificial Intelligence*. Berlin: Springer, pp. 812–823.

Bratko, I, Muggleton, S and Karalič, A, 1998, Applications of inductive logic programming. In Michalski, RS, Bratko, I and Kubat, M (eds) *Machine Learning and Data Mining: Methods and Applications*. New York: Wiley, pp. 131–143.

Bulitko, VV and Wilkins, DC, 1999, Learning to envision: an intelligent agent for ship damage control. In *Proceedings of the Workshop on Machine Learning and Intelligent Agents, ACAI 1999*, Chania.

Camacho, R, 1998, Inducing models of human control skills. In N Edellec, C and Rouveirol, C (eds) *Proceedings of the 10th European Conference on Machine Learning (ECML 98)*. London: Springer.

Chapman, F, Clinton, J, Kerber, R, Khabaza, T, Reinartz, T, Shearer, C and Wirth, R, 2000, *CRISP-DM 1.0 Step-bystep Data Mining Guide*, `http://www.crisp-dm.org`.

Dalaka, A, Kompare, B, Robnik-Šikonja, M and Sgardelis, S, 2000, Modeling the effects of environmental conditions on apparent photosynthesis of Stipa bromoides by machine learning tools. *Ecological Modelling* **129**, 245–257.

Debeljak, M, Džeroski, S, Jerina, K, Kobler, A and Adamič, M, 2001, Habitat suitability modelling for red deer (*Cervus elaphus* L.) in South-Western Slovenia with classification trees. *Ecological Modelling* **138**, 321–330.

Dolšak, B, Bratko, I and Jezernik, A, 1998, Applications of machine learning in finite element computation. In Michalski, RS, Bratko, I and Kubat, M (eds) *Machine Learning and Data Mining: Methods and Applications*. New York: Wiley, pp. 147–171.

Džeroski, S, Jacobs, N, Molina, M and Moure, C, 1998, ILP experiments in detecting traffic problems. In N Edellec, C and Rouveirol, C. (eds.), *Proceedings of the 10th European Conference on Machine Learning (ECML'98)*. Berlin: Springer, pp. 61–66.

Džeroski, S, Schulze-Kremer, S, Heidtke, K, Siems, K, Wettschereck, D and Blockeel, H, 1998, Diterpene Structure Elucidation from 13C NMR Spectra with Inductive Logic Programming. *Applied Artificial Intelligence* **12**, 363–384.

Džeroski, S, 2002, KDD Applications in Environmental sciences. In Klösgen, W and Zytkow, JM (eds) *Handbook of Data Mining and Knowledge Discovery*. Oxford: Oxford University Press, pp. 817–830.

Džeroski, S, Todorovski, L, Zmazek, B, Vaupotič, J and Kobal, I, 2003, Modelling soil radon concentration for earthquake prediction. In *Proceedings of the 6th International Conference on Discovery Science*. Berlin: Springer, pp. 87–99.

Fawcett, T and Provost, F, 1997, Data mining for adaptive fraud detection. *Data Mining and Knowledge Discovery* **1**, 291–316.

Fayyad, UM, Piatetsky-Shapiro, G, Smyth, P and Uthurusamy, R (eds), 1996, *Advances in Knowledge Discovery and Data Mining*. Boston, MA: AAAI Press/MIT Press.

Filipič, B and Junkar, M, 2000, Using inductive machine learning to support decision making in machining processes. *Computers in Industry* **43**, 31–41.

Flach, P *et al*., 2003, On the road to knowledge. In Mladenić, D, Lavrač, N, Bohanec, M and Moyle, S (eds) *Data Mining and Decision Support*. Dordrecht: Kluwer Academic Publishers, pp. 143–155.

Gamberger, D, Krstačić, G and Lavrač, N, 2003, Active subgroup mining: a case study in coronary heart disease risk group detection. *Artificial Intelligence in Medicine* **28**, 27–57.

Gamberger, D, Lavrač, N, Zelezny, F and Tolar, J, 2004, Induction of comprehensible models for gene expression datasets by subgroup discovery methodology. *Journal of Biomedical Informatics* **37**, 269–284.

Gams, M, Karba, N and Drobnič, M, 1997, Integration of multiple reasoning systems for process control. *Engineering Applications of Artificial Intelligence* **10**, 41–46.

Han, J and Kamber, M, 2001, *Data Mining: Concepts and Techniques*. San Francisco, CA: Morgan Kaufmann.

Hand, D, Mannila, H and Smyth, P, 2001, *Principles of Data Mining*. Boston, MA: MIT Press.

IDC, 2000, *ICT en Nederland*, Dutch Ministry of Economic Affairs.

Jerina, K, Debeljak, M, Džeroski, S, Kobler, A and Adamič, M, 2003, Modeling the brown bear population in Slovenia: a tool in the conservation management of a threatened species. *Ecological Modelling* **170**, 453–469.

King, RD, Srinivasan, A and Sternberg, MJE, 1995, Relating chemical activity to structure: an examination of ILP successes. *New Generation Computing* **13**, 411–433 (`http://www.informatik.uni-freiburg.de/~ml/carc.html`).

Knowledge Process Software, 2004a, Data Mining to improve process and energy performance. Future Practice Profile 67, Report Knowledge Process Software.

Knowledge Process Software, 2004b, Reducing energy costs in industry with modern control techniques. Good Practice Guide 215, Report Knowledge Process Software.

Knowledge Process Software, 2002. Using Advanced Data Mining to improve process control and energy management, New Practice Case Study 128, Report Knowledge Process Software.

Klema, J and Kout, J, 1999, Prediction of gas consumption. In *Systems Integration 1999*. Prague: Prague University of Economy, pp. 119–128.

Knobbe, A, Marseille, B, Moerbeek, O and Van der Wallen, D, 1998, Results in data mining for adaptive system management. *Proceedings Benelearn 1998*, pp. 31–38.

Kononenko, I, 2001, Machine learning for medical diagnosis: history, state of the art and perspective. *Artificial Intelligence in Medicine* **23**, 89–109.

Kononenko, I, 1993, Inductive and Bayesian learning in medical diagnosis. *Applied Artificial Intelligence* **7**, 317–337.

Koenig, I, 1998, *Uitstroom goed bekeken*, University of Amsterdam (in Dutch).

Kraakman, A, 1998, Application of machine learning in plant breeding: the process of putting a learning technique into practice. *Proceedings Benelearn 1998*. Wageningen: ATO-DLO, pp. 148–156.

Krisper, M, Cestnik, B, Karalič, A and Sočan, L, 1989, Constructing heuristic models with inductive learning: experiments in social science. In *Proceedings of the IASTED Expert Systems: Theory and Applications*. Zurich, Switzerland, pp. 85–89.

Kukar, M, Kononenko, I, Grošelj, C, Kralj, K and Fettich, J, 1999, Analysing and improving the diagnosis of ischaemic heart disease with machine learning. *Artificial Intelligence in Medicine* **16**, 25–50.

Kukar, M, Kononenko, I and Silvester, T, 1996, Machine learning in prognosis of the femoral neck fracture recovery. *Artificial Intelligence in Medicine* **5**, 431–451.

Langley, P and Simon, HA, 1995, Applications of machine learning and rule induction. *Communications of the ACM* **38**, 54–64.

Langley, P, 2001, The computational support of scientific discovery. In Paliouras, G, Karkaletsis, V and Spyropoulos, C (eds) *Machine Learning and its Applications*. Berlin: Springer.

Lopez, MJJ and Van der Wal, J, 2002, Crime analysis on residential burglary data. In van der Mey, J (ed.) *Dealing with the Data Flood*. The Hague: STT.

Manago, M and Auriol, E, 1997, Application of inductive learning and case-based reasoning for trouble-shooting industrial machines. In Kubat, M, Bratko, I and Michalski, RS (eds) *Machine Learning and Data Mining—Methods and Applications*. New York: Wiley.

Meij, J (ed.), 2002, *Dealing with the Data Flood*. The Hague: STT.

Michie, D, 1995, Problem decomposition and the learning of skills. In Lavrač, N and Wrobel, S (eds) *Proceedings European Conference on Machine Learning ECML-95*. Berlin: Springer, pp. 17–31.

Michalski, RS, 1983, A theory and methodology of inductive learning. In Michalski, R, Carbonell, J and Mitchell, T (eds) *Machine Learning. An Artificial Intelligence Approach*. Berlin: Springer, pp. 83–134.

Michalski, RS, Bratko, I and Kubat, M (eds), 1998, *Machine Learning and Data Mining: Methods and Applications*. Chichester: Wiley.

Middleton, SE, Shadbolt, NR and De Roure, DC, 2004, Ontological user profiling in recommender systems. *ACM Transactions on Information Systems* **22**, 54–88.

Mischiati, M and Neri, F, 2000, Applying local search and genetic evolution in concept learning systems to detect intrusion in computer networks. In Lopez de Mantaras, R and Plaza, E (eds) *Proceedings ECML-2000*. Berlin: Springer.

Mladenić, D and Grobelnik, M, 1993, Text and Web mining. In Mladenić, D, Lavrač, N, Bohanec, M and Moyle, S (eds) *Data Mining and Decision Support: Integration and Collaboration*. Dordrecht: Kluwer Academic, pp. 15–22.

Moczulski, W, 1999, Inductive learning in design: a method and case study concerning design of antifriction bearing systems. In Michalski, RS, Bratko, I and Kubat, M (eds) *Machine Learning and Data Mining: Methods and Applications*. New York: Wiley, pp. 131–143.

Morik, K and Scholz, M, 2003, The MiningMart Approach to Knowledge Discovery in Databases. In Zhong, N and Liu, J (eds) *Intelligent Technologies for Information Analysis*. Berlin: Springer, pp. 47–66.

Morik, K, Imboff, M, Brockhausen, P, Joachims, T and Gather, U, 2000, Knowledge discovery and knowledge validation in intensive care. *Artificial Intelligence in Medicine* **19**, 225–249 (`http://www-ai.cs.unidortmund.de/FORSCHUNG/PROJEKTE/SFB475C4/`).

Olave, M, Lavrač, N and Cestnik, B, 1988, Applications of expert systems to management control systems in public enterprises of developing countries. *OMEGA The International Journal of Management Science* **16**, 353–362.

Paass, G and Kindermann, J, 1998, Bayesian classification trees with overlapping leaves applied to credit-scoring. In Wu, X, Kotagiri, R and Korb, KB (eds) *Research and Development in Knowledge Discovery and Data Mining*. Berlin: Springer, pp. 234–245.

Paliouras, G, Karkaletsis, V and Spyropoulos, CD (eds), 2001, *Machine Learning and its Applications*. Heidelberg: Springer.

Peèar, Z, 2002, A model for evaluating performance of work processes in public administration by using the methods of artificial intelligence. Ph.D. Thesis, Faculty of Organizational Sciences, University of Maribor, 2002 (In Slovenian).

Saitta, L and Neri, F, 1998, Learning in the 'Real-World'. *Machine Learning* **30**, 133–164.

Sammut, C, Hurst, S, Kedzier, D and Michie, D, 1992, Learning to fly. In Sleeman, D and Edwards, P (eds) *Proceedings of the 9th International Conference on Machine Learning*, ICML-92. San Francisco, CA: Morgan Kaufmann.

Sanchez-Marre, M, Cortes, U, Lafuente, J, Roda, IR and Poch, M, 1996, DAI-DEPUR: an integrated and distributed architecture for wastewater treatment plants supervision. *Artificial Intelligence in Engineering* **10**, 275–285.

Schreiber, ATh, Akkermans, JM, Anjewierden, AA, de Hoog, R, Shadbolt, NR, Van de Velde, W and Wielinga, BJ, 2000, *Knowledge Engineering and Management: The CommonKADS Methodology*. Boston, MA: MIT Press.

Smyth, B and Cotter, P, 2000, A personalised TV listings service for the digital TV Age. *Journal of Knowledge-Based Systems* **13**, 53–59 (`http://www.cs.ucd.ie/smi/` and `http://kermit.ucd.ie/casper/`).

Spiliopoulou, M and Pohle, C, 2001, Data mining to measure and improve the success of Web sites. *Data Mining and Knowledge Discovery* **5**, 85–114.

Srinivasan, A and King, RD, 1999, Feature construction with inductive logic programming: a study of quantitative predictions of biological activity aided by structural attributes. *Data Mining and Knowledge Discovery* **3**, 37–57.

Todorovski, L, Džeroski, S and Kompare, B, 1998, Modeling and prediction of phytoplankton growth with equation discovery, *Ecological Modelling* **113**, 71–81.

Urbančič, T and Bratko, I, 1994, Reconstructing human skill with machine learning. In Cohn, A (ed.) *Proceedings of the 11th European Conference on Artificial Intelligence*. New York: Wiley, pp. 498–502.

Urbančič, T, Križman, V and Kononenko, I (eds), 1998, Review of AI applications by Department of Intelligent Systems of Jozef Stefan Institute and Artificial Intelligence Laboratory of the Faculty of Computer Science and Informatics. IJS Working Report IJS DP–7806, University of Ljubljana.

Van der Putten, P, 1999a, Data Mining in direct marketing databases. In Baets, W (ed.) *Complexity and Management: A Collection of Essays*. Singapore: World Scientific.

Van der Putten, P, 1999b, Datamining in Bedrijf. *Informatie en Informatiebeleid* **17**, 15–27 (in Dutch).

Van der Putten, P and Van Someren, M (eds), 2000, COIL Challenge 2000: The Insurance Company Case. Technical Report 2000–09, Leiden Institute of Advanced Computer Science.

Van der Putten, P and Van Someren, M, 2004, A bias-variance analysis of a real world learning problem: the CoIL challenge 2000. *Machine Learning* **57**, 177–195.

Vanhoof, K, Bloemer, J and Pauwels, K, 1997, A case study in loyalty and satisfaction research. In Van Someren, M and Widmer, G (eds) *Proceedings ECML*, pp. 290–297.

Vasconcelos, MH and Pecas Lopes, JA, 1999, Pruning kernel regression trees for security assessment of the Crete network. *Workshop on Machine Learning Applications to Power Systems ACAI*.

Verdenius, F and Van Someren, MW, 1997, Applications of inductive learning techniques: a survey in the Netherlands. *AI Communications* **10**(1), 3–20.

Widmer, G, Dixon, S, Goebl, W, Pampalk, E and Tobudic, A, 2003, In search of the Horowitz factor. *AI Magazine* **24**, 111–130.

Widmer, G and Tobudic, A, 2003, Playing Mozart by analogy: learning multi-level timing and dynamics strategies. *Journal of New Music Research* **32**, 259–268.

Zupan, B, Demšar, J, Kattan, MW, Beck, JR and Bratko, I, 2000, Machine learning for survival analysis: a case study on recurrence of prostate cancer. *Artificial Intelligence in Medicine* **20**, 59–75.