Environmental Applications of Data Mining

Sašo Džeroski

Department of Knowledge Technologies, Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia

Abstract

Data mining, the central activity in the process of knowledge discovery in databases (KDD), is concerned with finding patterns in data. This paper introduces and illustrates the most common types of patterns considered by data mining approaches and gives rough outlines of the data mining algorithms that are most frequently used to look for such patterns. In this paper, we also to give an overview of KDD applications in environmental sciences, complemented with a sample of case studies. The latter are described in slightly more detail and used to illustrate KDD-related issues that arise in environmental applications. The application domains addressed mostly concern ecological modelling.

Keywords: Data Mining; Knowledge Discovery; Decision Trees; Rule Induction; Environmental Applications; Ecological Modelling; Population Dynamics; Habitat Suitability;

Introduction

Knowledge discovery in databases (KDD) was initially defined as the “non-trivial extraction of implicit, previously unknown, and potentially useful information from data” (14). A revised version of this definition states that “KDD is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data” (11). According to this definition, data mining (DM) is a step in the KDD process concerned with applying computational techniques (i.e., data mining algorithms implemented as computer programs) to actually find patterns in the data. In a sense, data mining is the central step in the KDD process. The other steps in the KDD process are concerned with preparing data for data mining, as well as evaluating the discovered patterns (the results of data mining).

The above definitions contain very imprecise notions, such as knowledge and pattern. To make these (slightly) more precise, some explanations are necessary concerning data, patterns and knowledge, as well as validity, novelty, usefulness, and understandability. For example, the discovered patterns should be valid on new data with some degree of certainty (typically prescribed by the user). The patterns should potentially lead to some actions that are useful (according to user defined utility criteria). Patterns can be treated as knowledge: according to Frawley et al. (14),“a pattern that is interesting (according to a user-imposed interest measure) and certain enough (again according to the user’s criteria) is called knowledge.”

This paper will focus on data mining and will not deal with the other aspects of the KDD process (such as data preparation). Since data mining is concerned with finding patterns in data, the notions of most direct relevance here are the notions of data and patterns. Another key notion is that of a data mining algorithm, which is applied to data to find patterns valid in the data. Different data mining algorithms address different data mining tasks, i.e., have different intended use for the discovered patterns.

Data is a set of facts, e.g., cases in a database (according to Fayyad et al. (11)). Most commonly, the input to a data mining algorithm is a single flat table comprising a number of attributes (columns) and records (rows). When data from more than one table in a database needs to be taken into account, it is left to the user to manipulate the relevant tables. Usually, this results in a single table, which is then used as input to a data mining algorithm.

The output of a data mining algorithm is typically a pattern or a set of patterns that are valid in the given data. A pattern is defined as a statement (expression) in a given language, that describes (relationships among) the facts in a subset of the given data and is (in some sense) simpler than the enumeration of all facts in the subset (14; 11). Different classes of pattern languages are considered in data mining: they depend on the data mining task at hand. Typical representatives are equations; classification and regression trees; and association, classification, and regression rules. A given data mining algorithm will typically have a built-in class of patterns that it considers: the particular language of patterns considered will depend on the given data (the attributes and their values).

Many data mining algorithms come form the fields of machine learning and statistics. A common view in machine learning is that machine learning algorithms perform a search (typically heuristic) through a space of hypotheses (patterns) that explain (are valid in) the data at hand. Similarly, we can view data mining algorithms as searching, exhaustively or heuristically, a space of patterns in order to find interesting patterns that are valid in the given data.

In this paper, we first look at the prototypical format of data and the main data mining tasks addressed in the field of data mining. We next describe the most common types of patterns that are considered by data mining algorithms, such as equations, trees and rules. We also outline some of the main data mining algorithms searching for patterns of the types mentioned above.

Environmental sciences comprise the scientific disciplines, or parts of them, that consider the physical, chemical and biological aspects of the environment (2). A typical representative of environmental sciences is ecology, which studies the relationships among members of living communities and between those communities and their abiotic (non-living) environment.

Such a broad, complex and interdisciplinary field holds much potential for application of KDD methods. However, environmental sciences also pose many challenges to existing KDD methods. In this paper, we attempt to give an overview of KDD applications in environmental sciences, complemented with a sample of case studies in which the author has been involved. The latter are described in slightly more detail and used to illustrate KDD-related issues that arise in environmental applications.

Data mining tasks

This section first gives an example of what type of data is typically considered by data mining algorithms. It then defines the main data mining tasks addressed when such data is given. These include predictive modeling (classification and regression), clustering (grouping similar objects) and summarization (as exemplified by association rule discovery).

Data

The input to a data mining algorithm is most commonly a single flat table comprising a number of fields (columns) and records (rows). In general, each row represents an object and columns represent properties of objects. A hypothetical example of such a table is given in Table 1. We will use this example in the remainder of this paper to illustrate the different data mining tasks and the different types of patterns considered by data mining algorithms.

Here rows correspond to persons that have recently (in the last month) visited a small shop and columns carry some information collected on these persons (such as their age, gender, and income). Of particular interest to the store is the amount each person has spent at the store this year (over multiple visits), stored in the field Total. One can easily imagine that data from a transaction table, where each purchase is recorded, has been aggregated over all purchases for each customer to derive the values for this field. Customers that have spent over 15000 in total are of special value to the shop. An additional field has been created (BigSpender) that has value yes if a customer has spent over 15000 and no otherwise.

In machine learning terminology, rows are called examples and columns are called attributes (or sometimes features). Attributes that have numeric (real) values are called continuous attributes: Age, YearlyIncome and Total are continuous attributes. Attributes that have nominal values (such as Gender and BigSpender) are called discrete attributes.

Classification and regression

The tasks of classification and regression are concerned with predicting the value of one field from the values of other fields. The target field is called the class (dependent variable in statistical terminology). The other fields are called attributes (independent variables in statistical terminology).

If the class is continuous, the task at hand is called regression. If the class is discrete (it has a finite set of nominal values), the task at hand is called classification. In both cases, a set of data is taken as input, and a model (a pattern or a set of patterns) is generated. This model can then be used to predict values of the class for new data. The common term predictive modeling refers to both classification and regression.

Given a set of data (a table), only a part of it is typically used to generate (induce, learn) a predictive model. This part is referred to as the training set. The remaining part is reserved for evaluating the predictive performance of the learned

Table 1: A single table with data on customers (table `Customer`).

CID	Gender	Age	Income	Total	Big Spender
c1	Male	30	214000	18800	Yes
c2	Female	19	139000	15100	Yes
c3	Male	55	50000	12400	No
c4	Female	48	26000	8600	No
c5	Male	63	191000	28100	Yes
c6	Male	63	114000	20400	Yes
c7	Male	58	38000	11800	No
c8	Male	22	39000	5700	No
c9	Male	49	102000	16400	Yes
c10	Male	19	125000	15700	Yes
c11	Male	52	38000	10600	No
c12	Female	62	64000	15200	Yes
c13	Male	37	66000	10400	No
c14	Female	61	95000	18100	Yes
c15	Male	56	44000	12000	No
c16	Male	36	102000	13800	No
c17	Female	57	215000	29300	Yes
c18	Male	33	67000	9700	No
c19	Female	26	95000	11000	No
c20	Female	55	214000	28800	Yes

model and is called the testing set. The testing set is used to estimate the performance of the model on new, unseen data, or in other words, to estimate the validity of the pattern(s) on new data.

Clustering

Clustering is concerned with grouping objects into classes of similar objects (19). A cluster is a collection of objects that are similar to each other and are dissimilar to objects in other clusters. Given a set of examples, the task of clustering is to partition these examples into subsets (clusters). The goal is to achieve high similarity between objects within individual clusters (interclass similarity) and low similarity between objects that belong to different clusters (intraclass similarity).

Clustering is known as cluster analysis in statistics, as customer segmentation in marketing and customer relationship management, and as unsupervised learning in machine learning. Conventional clustering focusses on distance-based cluster analysis. The notion of a distance (or conversely, similarity) is crucial here: objects are considered to be points in a metric space (a space with a distance measure). In conceptual clustering, a symbolic representation of the resulting clusters is produced in addition to the partition into clusters: we can thus consider each cluster to be a concept (much like a class in classification).

Association analysis

Association analysis (15) is the discovery of association rules. Market basket analysis has been a strong motivation for the development of association analysis. Association rules specify correlations between frequent itemsets (sets of items, such as bread and butter, which are often found together in a transaction, e.g., a market basket).

The task of association analysis is typically performed in two steps. First, all frequent itemsets are found, where an itemset is frequent if it appears in at least a given percentage s (called support) of all transactions. Next, association rules are found of the form $X \rightarrow Y$, where X and Y are frequent itemsets and confidence of the rule (the percentage of transactions containing X that also contain Y) passes a threshold c.

Other data mining tasks

The above three data mining tasks receive by far the most attention within the data mining field and algorithms for performing such tasks are typically included in data mining tools. While classification and regression are of predictive nature, cluster analysis and association analysis are of descriptive nature. Subgroup discovery is at the boundary between predictive and descriptive tasks. Several additional data mining tasks (15) are of descriptive nature, including data characterization and discrimination, outlier analysis and evolution analysis.

Patterns

Patterns are of central importance in data mining and knowledge discovery. Data mining algorithms search the given data for patterns. Discovered patterns that are valid, interesting and useful can be called knowledge.

Frawley et al. (14) define a pattern in a dataset as a statement that describes relationships in a subset of the dataset with some certainty, such that the statement is simpler (in some sense) than the enumeration of all facts in the dataset. A pattern thus splits the dataset, as it pertains to a part of it, and involves a spatial aspect which may be visualized.

This section introduces the most common types of patterns that are considered by data mining algorithms. Note that the same type of pattern may be used in different data mining algorithms addressing different tasks: trees can be used for classification, regression or clustering (conceptual), and so can distance-based patterns.

Equations

Statistics is one of the major scientific disciplines that data mining draws upon. A predictive model in statistics most commonly takes the form of an equation.

Linear models predict the value of a target (dependent) variable as a linear combination of the input (independent) variables. Three linear models that predict the value of the variable Total are represented by Equations 1, 2, and 3. These have been derived using linear regression on the data from Table 1.

$$\begin{aligned}
\text{Total} &= 189.5275 \times \text{Age} + 7146.89 \qquad (1)\\
\text{Total} &= 0.093 \times \text{Income} + 6119.74 \qquad (2)\\
\text{Total} &= 189.126 \times \text{Age} + 0.0932 \times \text{Income} - 2420.67
\end{aligned}$$

Linear equations involving two variables (such as Equations 1 and 2) can be depicted as straight lines in a two-dimensional space (see Fig.). Linear equations involving three variables (such as Equation 3) can be depicted as planes in a three-dimensional space. Linear equations, in general, represent hyper-planes in multidimensional spaces. Nonlinear equations are represented by curves, surfaces and hyper-surfaces.

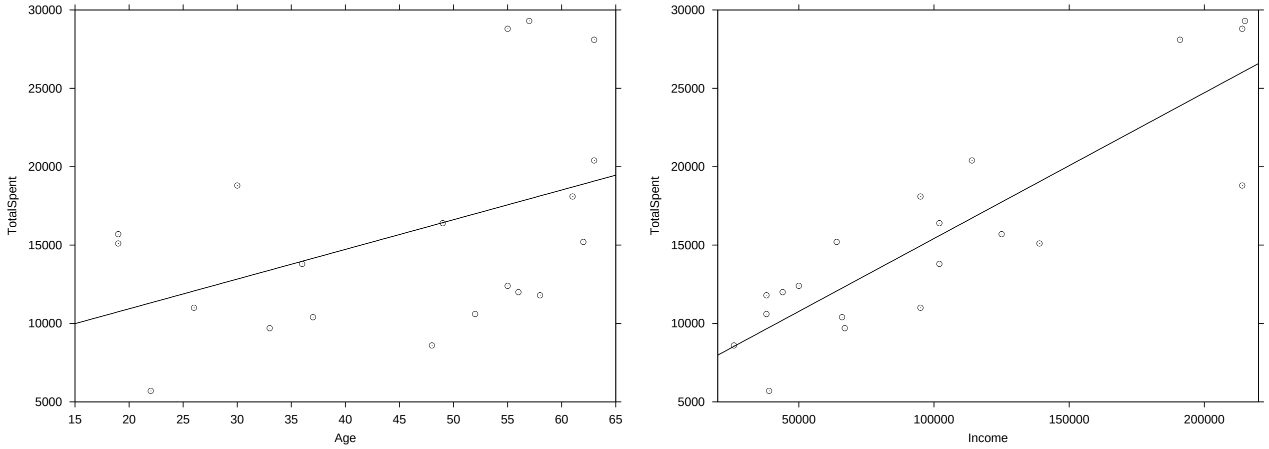

Figure 1: *Two regression lines that predict the value of variable Total from each of the variables Age and Income, respectively. The points correspond to the training examples.*

Note that equations (or rather inequalities) can be also used for classification. If the value of the expression $0.093 \times \mathrm{Income} + 6119.744$ is greater than 15000, for example, we can predict the value of the variable BigSpender to be “Yes”. Points for which “Yes” will be predicted are those above the regression line in the left-hand part of Fig. .

Decision trees

Decision trees are hierarchical structures, where each internal node contains a test on an attribute, each branch corresponds to an outcome of the test, and each leaf node gives a prediction for the value of the class variable. Depending on whether we are dealing with a classification or a regression problem, the decision tree is called a classification or a regression tree, respectively. Two classification trees derived from the dataset in Table 1 are given in Fig. 2. An example regression tree, also derived from the dataset in Table 1, is given in Fig. 3.

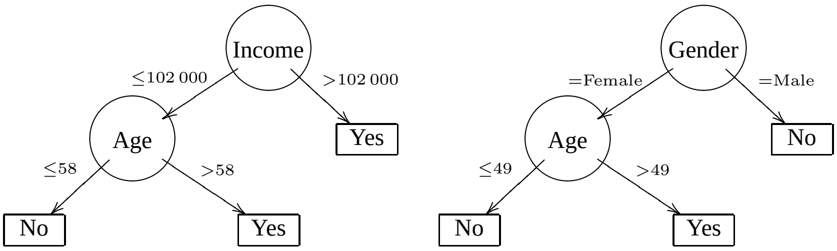

Figure 2: Two classification trees that predict the value of variable BigSpender from the variables Age and Income, and Age and Gender, respectively.

Regression tree leaves contain constant values as predictions for the class value. They thus represent piece-wise constant functions. Model trees, where leaf nodes can contain linear models predicting the class value, represent piece-wise linear functions.

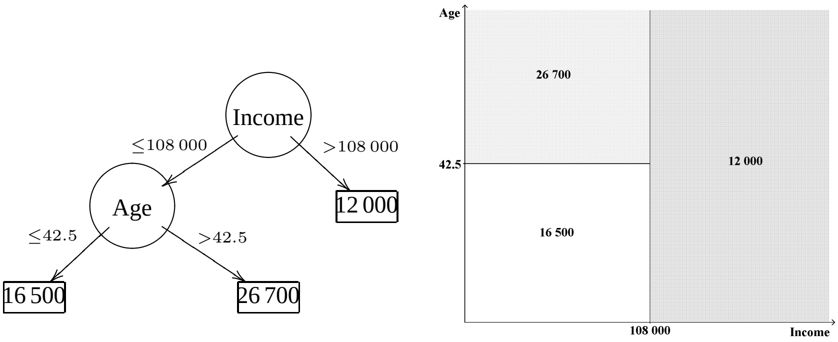

Figure 3: A regression tree and the partition of the data space induced by the tree. The tree predicts the value of the variable Total from the variables Age and Income.

Note that decision trees represent total partitions of the data space, where each test corresponds to an axis-parallel split. This is illustrated in Fig. 3. Most algorithms for decision tree induction consider such axis-parallel splits, but there are a few algorithms that consider splits along lines that need not be axis-parallel or even consider splits along non-linear curves.

Predictive rules

We will use the word rule here to denote patterns of the form “IF Conjunction of conditions THEN Conclusion.” The individual conditions in the conjunction will be tests concerning the values of individual attributes, such as “Income $\leq$ 108000” or “Gender=Male”. For predictive rules, the conclusion gives a prediction for the value of the target (class) variable.

If we are dealing with a classification problem, the conclusion assigns one of the possible discrete values to the class, e.g., “BigSpender=No”. A rule applies to an example if the conjunction of conditions on the attributes is satisfied by the particular values of the attributes in the given example. Each rule corresponds to a hyper-rectangle in the data space, as illustrated in Fig. 4.

IF Income $\leq$ 102000
AND Age $\leq$ 58
THEN BigSpender=No
ELSE
DEFAULT BigSpender=Yes.

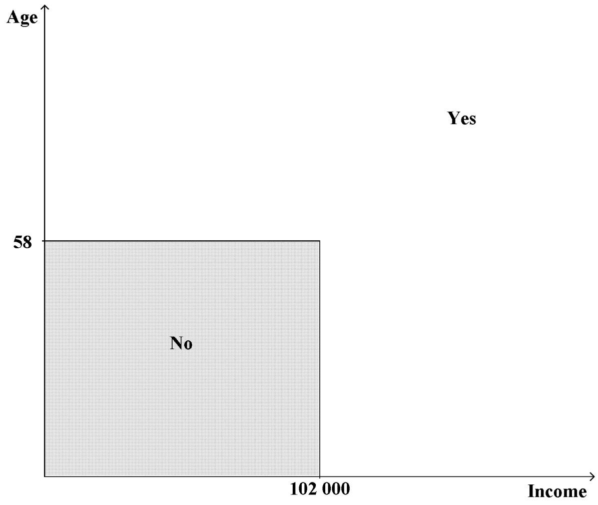

Figure 4: A partition of the data space induced by an ordered list of rules, derived from the data in Table 1. The shaded box corresponds to the first rule in the list IF Income $\leq$ 102000 AND Age $\leq$ 58 THEN BigSpender=No, while the remainder of the data space is covered by the default rule BigSpender=Yes.

Predictive rules can be ordered or unordered. Unordered rules are considered independently and several of them may apply to a new example that we need to classify. A conflict resolution mechanism is needed if two rules which recommend different classes apply to the same number of examples. A default rule typically exists, whose recommendation is taken if no other rule applies.

Ordered rules form a so-called decision list. Rules in the list are considered from the top to the bottom of the list. The first rule that applies to a given example is used to predict its class value. Again, a default rule with an empty precondition is typically found as the last rule in the decision list and is applied to an example when no other rule applies.

An ordered list and an unordered list of rules are given in Table 2. Both have been derived using a covering algorithm, described in the next section. The ordered list of rules in Fig. 4, on the other hand,has been generated from the decision tree in the left-hand side of Fig. 2. Note that each of the leaves of a classification tree corresponds to a classification rule. Although less common in practice, regression rules also exist, and can be derived, e.g., by transcribing regression trees into rules.

Table 2: An ordered (top) and an unordered (bottom) set of classification rules derived from the data in Table 1.

Ordered rules
IF Age $<$ 60 AND Income $<$ 81000 THEN BigSpender = No ELSE
IF Age $>$ 42 THEN BigSpender = Yes ELSE
IF Income $>$ 113500 THEN BigSpender = Yes ELSE
DEFAULT BigSpender=No

Unordered rules
IF Income $>$ 108000 THEN BigSpender = Yes
IF Age $\geq$ 49 AND Income $>$ 57000 THEN BigSpender = Yes
IF Age $\leq$ 56 AND Income $<$ 98500 THEN BigSpender = No
IF Income $<$ 51000 THEN BigSpender = No
IF 33 $<$ Age $\leq$ 42 THEN BigSpender = No
DEFAULT BigSpender=Yes

Data mining algorithms

The previous section described several types of patterns that can be found in data. This section outlines some basic algorithms that can be used to find such patterns in data. In most cases, this involves heuristic search through the space of possible patterns of the selected form.

Linear and multiple regression

Linear regression is the simplest form of regression (16). Bivariate linear regression assumes that the class variable can be expressed as a linear function of one attribute, i.e., $C = \alpha + \beta \times A$. Given a set of data, the coefficients α and β can be calculated using the method of least squares, which minimizes the error $\sum_i (c_i - \alpha - \beta a_i)^2$ between the measured values for C (c_i), and the values calculated from the measured values for A (a_i) using the above equation. We have

$$\beta = \sum_i (a_i - \overline{a})(c_i - \overline{c}) / \sum_i (a_i - \overline{a})^2$$

$$\alpha = \overline{c} - \beta\overline{a},$$

where $\overline{a}$ is the average of $a_1, \ldots, a_n$ and $\overline{c}$ is the average of $c_1, \ldots, c_n$.

Multiple regression extends linear regression to allow the use of more than one attribute. The class variable can thus be expressed as a linear function of a multi-dimensional attribute vector, i.e., $C = \sum_{i=1}^{n} \beta_i \times A_i$. This form assumes that the dependent variable and the independent variables have mean values of zero (which is achieved by transforming the variables - the mean value of a variable is subtracted from each measured value for that variable). The method of least squares can also be applied to find the coefficients β_i. If we write the equation $C = \sum_{i=1}^{n} \beta_i \times A_i$ in matrix form $\underline{C} = \underline{\beta}\underline{A}$, where $\underline{C} = (c_1, \ldots, c_n)$ is the vector of measured values for the dependent variable and $\underline{A}$ is the matrix of measured values for the independent variables, we can calculate the vector of coefficients β as

$$\underline{\beta} = (\underline{A}^T \underline{A})^{-1} \underline{A}^T \underline{C}$$

where the operations of matrix transposition $\bullet^T$ and matrix inversion $\bullet^{-1}$ are used. The use of non-linear transformations, such as $A_i = A^i, i = 1, ..., n$, allows non-linear models to be found by using multiple regression: such models are linear in the parameters.

Note that both for linear and multiple regression, the coefficients α, β, and β_i can be calculated directly from a formula and no search through the space of possible equations takes place. Equation discovery approaches (10), which do not assume a particular functional form, search through a space of possible functional forms and look both for an appropriate structure and coefficients of the equation.

Linear regression is normally used to predict a continuous class, but can also be used to predict a discrete class. Generalized linear models can be used for this, of which logistic regression is a typical representative. The fitting of generalized linear models is currently the most frequently applied statistical technique (32).

Top-down induction of decision trees

Finding the smallest decision tree that would fit a given data set is known to be computationally expensive (NP-hard). Heuristic search, typically greedy, is thus employed to build decision trees. The common way to induce decision trees is the so-called Top-Down Induction of Decision Trees (TDIDT) (27)). Tree construction proceeds recursively starting with the entire set of training examples (entire table). At each step, an attribute is selected as the root of the (sub)tree and the current training set is split into subsets according to the values of the selected attribute.

For discrete attributes, a branch of the tree is typically created for each possible value of the attribute. For continuous attributes, a threshold is selected and two branches are created based on that threshold. For the subsets of training examples in each branch, the tree construction algorithm is called recursively. Tree construction stops when the examples in a node are sufficiently pure (i.e., all are of the same class) or if some other stopping criterion is satisfied (there is no good attribute to add at that point). Such nodes are called leaves and are labeled with the corresponding values of the class.

Different measures can be used to select an attribute in the attribute selection step. These also depend on whether we are inducing classification or regression trees (4). For classification, Quinlan (27) uses information gain, which is the expected reduction in entropy of the class value caused by knowing the value of the given attribute. Other attribute selection measures, however, such as the Gini index or the accuracy of the majority class, can and have been used in classification tree induction. In regression tree induction, the expected reduction in variance of the class value can be used.

An important mechanism used to prevent trees from over-fitting data is tree pruning. Pruning can be employed during tree construction (pre-pruning) or after the tree has been constructed (post-pruning). Typically, a minimum number of examples in branches can be prescribed for pre-pruning and a confidence level in accuracy estimates for leaves for post-pruning.

The covering algorithm for rule induction

In the simplest case of concept learning, one of the classes is referred to as positive (examples belonging to the concept) and the other as negative. For a classification problem with several class values, a set of rules is constructed for each class. When rules for class c_i are constructed, examples of this class are referred to as positive, and examples from all the other classes as negative.

The covering algorithm works as follows. We first construct a rule that correctly classifies some examples. We then remove the positive examples covered by the rule from the training set and repeat the process until no more examples remain. The pseudo code for this algorithm is given in Table 10.2.

Within this outer loop, different approaches can be taken to find individual rules. One approach is to heuristically search the space of possible rules top-down, i.e., from general to specific (in terms of examples covered this means from rules covering many to rules covering fewer examples) (6). To construct a single rule that classifies examples into class c_i, we start with a rule with an empty antecedent (IF part) and the selected class c_i as a consequent (THEN part). The antecedent of this rule is satisfied by all examples in the training set, and not only those of the selected class. We then progressively refine the antecedent by adding conditions to it, until only examples of class c_i satisfy the antecedent. To allow for handling imperfect data, we may construct a set of rules which is imprecise, i.e., does not classify all examples in the training set correctly.

Date mining applications in ecological modelling

Ecological modelling is concerned with the development of models of the relationships among members of living communities and between those communities and their abiotic environment. These models can then be used to better understand the domain at hand or to predict the behavior of the studied communities and thus support decision making for environmental management. Typical modelling topics are population dynamics of several interacting species and habitat suitability for a given species (or higher taxonomic unit).

Modelling population dynamics

Population dynamics studies the behavior of a given community of living organisms (population) over time, usually taking into account abiotic factors and other living communities in the environment. For example, one might study the population of phytoplankton in a given lake (33) and its relation to water temperature, concentrations of nutrients/pollutants (such as nitrogen and phosphorus) and the biomass of zooplankton (which feeds on phytoplankton). The modelling formalism most often used by ecological experts is the formalism of differential equations, which describe the change of state of a dynamic system over time. A typical approach to modelling population dynamics is as follows: an ecological expert writes a set of differential equations that capture the most important relationships in the domain. These are often linear differential equations. The coefficients of these equations are then determined (calibrated) using measured data.

Relationships among living communities and their abiotic environment can be highly nonlinear. Population dynamics (and other ecological) models have to reflect this to be realistic. This has caused a surge of interest in the use of techniques such as neural networks for ecological modelling (25). Measured data are used to train a neural network which can then be used to predict future behavior of the studied population. In this fashion, population dynamics of algae (28), aquatic fauna (30), fish (5), phytoplankton (29) and zooplankton (1) - among other - have been modelled.

While regression tree induction has also been used to model population dynamics. Systems for discovery of differential equations have proved most useful in this respect (10), since differential equations are the prevailing formalism used for ecological modelling. Algal growth has been modelled for the Lagoon of Venice (21; 23) and the Slovenian Lake of Bled (22), as well as phytoplankton growth for the Danish Lake Glumsoe (33).

Case study: Modelling algal growth in the Lagoon of Venice

The beautiful and shallow Lagoon of Venice is under heavy pollution stress due to agricultural activities (use of fertilizers) on the neighboring mainland. Pollutants are food (nutrients) for algae, which have on occasions grown excessively to the point of suffocating themselves, then decayed and caused unpleasant odors (noticed also by the tourists). Models of algal growth are needed to support environmental management decisions and answer questions such as: ”Would a reduction in the use of phosphorus-rich fertilizers reduce algal growth?”

Kompare and Džeroski (21) use regression trees and equation discovery to model the growth of the dominant species of algae (*Ulva rigida*) in the lagoon of Venice in relation to water temperature, dissolved nitrogen and phosphorus and dissolved oxygen. The trees give a rough picture of the relative importance of the factors influencing algal growth (cf. Fig. 5), revealing that nitrogen is the limiting factor (and thus providing a negative answer to the question in the above paragraph). The equations discovered, on the other hand, give better prediction of the peaks and crashes of algal biomass.

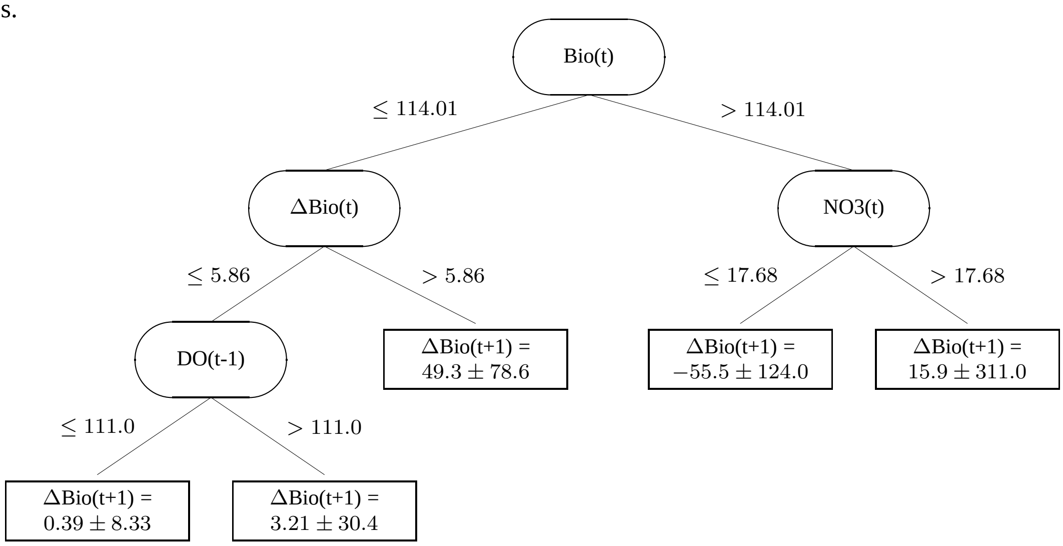

Figure 5: A regression tree for predicting algal growth, i.e., change in biomass. $Bio(t)$, $DO(t)$ and $NO_3(t)$ stand for the concentrations of biomass, dissolved oxygen and nitrates at time t. $\Delta X(t)$ stand for $X(t) - X(t-1)$.

Severe problems of data quality were encountered in this application.

1. Dissolved oxygen, for example, was measured at the water surface approximately at noon (when oxygen is produced by photosynthesis and is plentiful) and does not reveal potential anoxic conditions (which might occur at night) - which it was supposed to reveal.
2. Measurement errors of algal biomass were estimated to be quite large by the domain experts (up to 50% relative error).
3. Finally, winds were not taken into account: these might move algae away from the sampling stations and cause huge variations in the observed biomass values.

Case study: Phytoplankton growth in Lake Glumsoe

The shallow Lake Glumsoe is situated in a sub-glacial valley in Denmark. It has received mechanically-biologically treated waste water, as well as non-point source pollution due to agricultural activities in the surrounding area. High concentration of pollutants (food for phytoplankton) lead to excessive growth of phytoplankton and consequently no submerged vegetation, due to low transparency of the water and oxygen deficit (anoxia) at the bottom of the lake. It was thus important to have a good model of phytoplankton growth to support environmental management decisions.

We used KDD methods for the discovery of differential equations (10) to relate phytoplankton ($phyt$) growth to water temperature ($temp$), nutrient concentrations (nitrogen-$nitro$ and phosphorus - $phosp$) and zooplankton concentration - zoo (33). Some elementary knowledge on population dynamics modelling was taken into account during the discovery process. This domain knowledge tells us that a term called Monod’s term, which has the form $Nutrient/(Nutrient + constant)$ is a reasonable term to be expected in differential equations describing the growth of an organism that feeds on $Nutrient$. It describes the saturation of the population of organisms with the nutrient.

Table 3: The discovered model for phytoplankton growth in Lake Glumsoe.

$$\dot{phyt} = 0.553 \cdot temp \cdot phyt \cdot \frac{phosp}{0.0264 + phosp} - 4.35 \cdot phyt - 8.67 \cdot phyt \cdot zoo$$

The discovered model is given in Table 3. Here $\dot{phyt}$ denotes the rate of change of phytoplankton concentration. The model reveals that phosphorus is the limiting nutrient for phytoplankton growth, as it includes a Monod term with phosphorus as a nutrient. This model made better predictions than a linear model, which has the form
$\dot{phyt} = -5.41 - 0.0439 \cdot phyt - 13.5 \cdot nitro - 38.2 \cdot zoo + 93.9 \cdot phosp + 3.20 \cdot temp$

It was also more understandable to domain experts: the first term describes phytoplankton growth, where temperature and phosphorus are limiting factors. The last two terms describe phytoplankton death and the feeding of zooplankton on phytoplankton.

The following issues were raised in this application:

1. Data quantity and preprocessing: measurements were only made at 14 time points during two months (once weekly). Some preprocessing/interpolation was thus necessary to generate enough data for discovering differential equations.
2. Data quality: ecological experts often have poor understanding of modelling concepts, which strongly influences the way data are collected. An electrical engineer with knowledge of control theory would know much better that sampling frequency has to be increased at times when the system under study has faster dynamics (e.g., at peaks of phytoplankton growth).
3. The need for taking into account domain knowledge during the KDD process: this can compensate to a certain extent for poor data quality and quantity (as was the case in this application). This issue is of great importance, yet few KDD methods allow for the provision of domain knowledge by experts.

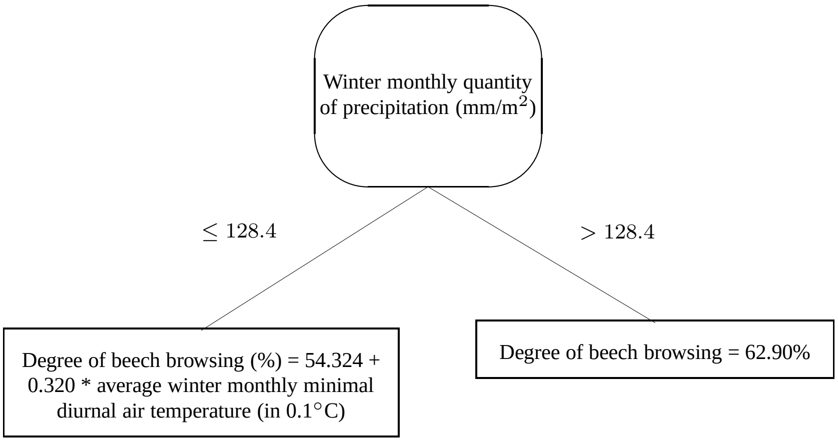
Winter monthly quantity of precipitation (mm/m^2)
≤ 128.4
> 128.4
Degree of beech browsing (%) = 54.324 + 0.320 * average winter monthly minimal diurnal air temperature (in 0.1°C)
Degree of beech browsing = 62.90%

Figure 6: A regression tree for predicting the degree of beech browsing.

Case study: Modelling the interactions of a red deer population with the new growth in a forest

Here we studied the interactions among a population of red deer and new forest growth in a natural regenerated forest in Slovenia. Ideally, foresters would like to keep in balance the size of the deer population and the rate of regeneration of the forest: if the deer population is large, so are the browsing rates of new forest growth and regeneration slows down. Understanding the relationship between the two is crucial for managing the balance. Our study has shown that meteorological parameters strongly influence this relationship and have to be taken into account.

A preliminary study using regression trees to model the interactions was performed by Stankovski et al. (31). Here we summarize the results of a follow-up study that used a slightly larger dataset, cleaner data, and more reliable methods of regression tree induction (8). The induced models show that the degree of browsing for maple (the preferred browse species of red deer) depends directly on the size of the population. The degree of beech browsing, on the other hand, was most strongly influenced by meteorological parameters, i.e., winter monthly quantity of precipitation (snow) and average monthly minimal diurnal air temperature. (cf. Fig. 6). While beech is not the preferred browse species of red deer, it is consumed yearlong; it is also elastic and snow-resistant and thus more exposed to the reach of red deer even in deeper snow.

The following issues were raised by this application:

1. Data quantity: the size of the deer population and browsing rates are only estimated once a year. Even though we were dealing with 18 years worth of data, these were still only 18 data points.
2. Data quality: some of the data collected in this domain were unreliable and had to be cleaned/corrected/removed before obtaining reasonable results.
3. Missing information: the outcome of the data analysis process suggested that measuring winter and summer browsing rates separately would greatly improve the models. This information was not measured and it couldn't be reconstructed from the currently measured data, but should be measured in the future.

Habitat-suitability modelling

Habitat-suitability modelling is closely related to population dynamics modelling. Typically, the effect of the abiotic characteristics of the habitat on the presence, abundance or diversity of a given taxonomic group of organisms is studied. For example, one might study the influence of soil characteristics, such as soil temperature, water content, and proportion of mineral soil on the abundance and species richness of *Collembola* (springtails), the most abundant insects in soil (24). The study uses neural networks to build a number of predictive models for collembolan diversity. Another study of habitat suitability modelling by neural networks is given by Ozesmi and Ozesmi (26).

Several habitat-suitability modelling applications of other data mining methods are surveyed by Fielding (13). Fielding (12) applies a number of methods, including discriminant analysis, logistic regression, neural networks and genetic algorithms, to predict nesting sites for golden eagles. Bell (3) uses decision trees to describe the winter habitat of pronghorn antelope. Jeffers (17) uses a genetic algorithm to discover rules that describe habitat preferences for aquatic species in British rivers.

The author has been involved in a number of habitat suitability studies using rule induction and decision trees. Rule induction was used to relate the presence or absence of a number of species in Slovenian rivers to physical and chemical properties of river water, such as temperature, dissolved oxygen, pollutant concentrations, chemical oxygen demand, etc. (9). Regression trees were used to study the influence of soil characteristics, such as soil texture, moisture and acidity on the abundance (total number of individuals) and diversity (number of species) of *Collembola* (springtails) (18). We have also used decision trees to model habitat suitability for red deer in Slovenian forests using GIS data, such as elevation, slope, and forest composition (7). Finally, decision trees that model habitat suitability for brown bears have been induced from GIS data and data on brown bear sightings (20). The model has then been used to identify the most suitable locations for the construction of wildlife bridges/underpasses that would enable the bears to safely cross the highway passing through the bear habitat.

Summary

This paper introduced data mining, the central activity in the process of knowledge discovery in databases (KDD), which is concerned with finding patterns in data. It also gave an overview of KDD applications in environmental sciences, complemented with a sample of case studies. The paper is based on the chapter ”Data Mining in a Nutshell” by S. Džeroski, which appears in the book *Relational Data Mining*, edited by S. Džeroski and N. Lavrač and published by Springer in 2001, as well as the article ”Applications of KDD in Environmental Sciences”, that appears in the ”Handbook of Data Mining and Knowledge Discovery”, edited by W. Kloesgen, and J. M. Zytkow, published by Oxford University Press in 2002. For more information on the topic of this paper, we refer the reader to these.

References

[1] I. Aoki, T. Komatsu,and K. Hwang. Prediction of response of zooplankton biomass to climatic and oceanic changes. *Ecological Modelling* 120(2-3): 261-270, 1999.

[2] M Allaby. *Basics of Environmental Science*. Routledge, London, 1996.

[3] J.F. Bell. Tree based methods. In A.H. Fielding, editor, *Machine Learning Methods for Ecological Applications*, pages 89–105. Kluwer Academic Publishers, Dordrecht, 1999.

[4] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth, Belmont, CA, 1984.

[5] S. Brosse, J.-F. Guegan, J.-N. Tourenq, and S. Lek. The use of artificial neural networks to assess fish abundance and spatial occupancy in the littoral zone of a mesotrophic lake. *Ecological Modelling* 120(2-3): 299-311, 1999.

[6] P. Clark and R. Boswell. Rule induction with CN2: Some recent improvements. In *Proceedings of the Fifth European Working Session on Learning*, pages 151–163. Springer, Berlin, 1991.

[7] M. Debeljak, S. Džeroski, K. Jerina, A. Kobler, and M. Adamič. Habitat suitability modelling of red deer (Cervus elaphus, L.) in South-Central Slovenia. *Ecological Modelling*. Forthcoming, 2000.

[8] M. Debeljak, S. Džeroski, and M. Adamič. Interactions among the red deer (Cervus elaphus, L.) population, meteorological parameters and new growth of the natural regenerated forest in Snežnik, Slovenia. *Ecological Modelling* 121(1): 51–61, 1999.

[9] S. Džeroski and J. Grbović. Knowledge discovery in a water quality database. In *Proc. First International Conference on Knowledge Discovery and Data Mining*, pages 81–86. AAAI Press, Menlo Park, CA, 1995.

[10] S. Džeroski, L. Todorovski, I. Bratko, B. Kompare, and V. Križman. Equation discovery with ecological applications. In A.H. Fielding, editor, *Machine Learning Methods for Ecological Applications*, pages 185–207. Kluwer, Boston, 1999.

[11] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery: An overview. In U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, pages 1–34. MIT Press, Cambridge, MA, 1996.

[12] A.H. Fielding. An introduction to machine learning methods. In A.H. Fielding, editor, *Machine Learning Methods for Ecological Applications*, pages 1–35. Kluwer Academic Publishers, Dordrecht, 1999.

[13] A.H. Fielding, editor. *Machine Learning Methods for Ecological Applications*. Kluwer Academic Publishers, Dordrecht, 1999.

[14] W. Frawley, G. Piatetsky-Shapiro, and C. Matheus. Knowledge discovery in databases: An overview. In G. Piatetsky-Shapiro and W. Frawley, editors, *Knowledge Discovery in Databases*, pages 1–27. MIT Press, Cambridge, MA, 1991.

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

[16] R.V. Hogg and A.T. Craig. *Introduction to Mathematical Statistics*, 5th edition. Prentice Hall, Englewood Cliffs, NJ, 1995.

[17] J.N.R. Jeffers. Genetic algorithms I. In A.H. Fielding, editor, *Machine Learning Methods for Ecological Applications*, pages 107–121. Kluwer Academic Publishers, Dordrecht, 1999.

[18] C. Kampichler, S. Džeroski, and R. Wieland. The application of machine learning techniques to the analysis of soil ecological data bases: relationships between habitat features and *Collembola* community characteristics. *Soil Biology and Biochemistry* 32: 197–209, 2000.

[19] L. Kaufman and P. J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley & Sons, New York, 1990.

[20] A. Kobler and M. Adamič. Brown bears in Slovenia: identifying locations for construction of wildlife bridges across highways. In *Proc. Third Intl. Conf. on Wildlife Ecology and Transportation*, pages 29–38. Florida Department of Transportation, Tallahassee, FL, 1999.

[21] B. Kompare and S. Džeroski. Getting more out of data: Automated modelling of algal growth with machine learning. In *Proc. International Conference on Coastal Ocean Space Utilization*, pages 209–220. University of Hawaii, 1995.

[22] B. Kompare, S. Džeroski, and A. Karalič. Identification of the Lake of Bled ecosystem with the artificial intelligence tools M5 and FORS. In *Proc. Fourth International Conference on Water Pollution*, pages 789–798. Computational Mechanics Publications, Southampton, 1997.

[23] B. Kompare, S. Džeroski, and V. Križman. Modelling the growth of algae in the Lagoon of Venice with the artificial intelligence tool GoldHorn. In *Proc. Fourth International Conference on Water Pollution*, pages 799–808. Computational Mechanics Publications, Southampton, 1997.

[24] S. Lek-Ang, L. Deharveng, and S. Lek. Predictive models of collembolan diversity and abundance in a riparian habitat. *Ecological Modelling* 120(2-3): 247-260, 1999.

[25] S. Lek, and J.F. Guegan, guest editors. Application of Artificial Neural Networks in Ecological Modelling. Special issue of *Ecological Modelling* 120 (2-3), 1999.

[26] S.L. Ozesmi and U. Ozesmi. An artificial neural network approach to spatial habitat modelling with interspecific interaction. *Ecological Modelling* 116 (1): 15–31, 1999.

[27] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1: 81–106, 1986.

[28] F. Recknagel, M. French, P. Harkonen, and K. Yabunaka. Artificial neural network approach for modelling and prediction of algal blooms. *Ecological Modelling* 96 (1-3): 11–28, 1997.

[29] M. Scardi and L.W. Harding. Developing an empirical model of phytoplankton primary production: a neural network case study. *Ecological Modelling* 120(2-3): 213-223, 1999.

[30] I.M. Schleiter, D. Borchardt, R. Wagner, T. Dapper, K.-D. Schmidt, H.-H. Schmidt, and H. Werner. Modelling water quality, bioindication and population dynamics in lotic ecosystems using neural networks. *Ecological Modelling* 120(2-3): 271–286, 1999.

[31] V. Stankovski, M. Debeljak, I. Bratko, and M. Adamič. Modelling the population dynamics of red deer (Cervus elaphus L.) with regard to forest development. *Ecological Modelling* 108(1-3): 145-153, 1998.

[32] P. Taylor. Statistical methods. In M. Berthold and D.J. Hand, editors, *Intelligent Data Analysis: An Introduction*, pages 67–127. Springer, Berlin, 1999.

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