Chapter 7
Disaggregation Analysis and Statistical Learning: An Integrated Framework for Multicriteria Decision Support

Michael Doumpos and Constantin Zopounidis

Abstract Disaggregation methods have become popular in multicriteria decision aiding (MCDA) for eliciting preferential information and constructing decision models from decision examples. From a statistical point of view, data mining and machine learning are also involved with similar problems, mainly with regard to identifying patterns and extracting knowledge from data. Recent research has also focused on the introduction of specific domain knowledge in machine learning algorithms. Thus, the connections between disaggregation methods in MCDA and traditional machine learning tools are becoming stronger. In this chapter the relationships between the two fields are explored. The differences and similarities between the two approaches are identified and a review is given regarding the integration of the two fields.

7.1 Introduction

Decision-making under multiple criteria or uncertainty is a subjective task that depends on the system of preferences of the decision-maker (DM). Multicriteria decision aid (MCDA) provides a broad set of methodologies suitable for such situations, where conflicting criteria, goals, objectives, and points of view, have to be taken into consideration. Among others, MCDA is involved with problem structuring, preference modeling, the construction and characterization of different forms of criteria aggregation models, as well as the design of interactive solution and decision aid/support procedures.

In many cases, the decision situation involves a finite set of actions or alternatives that need to be evaluated following a choice, ranking, sorting or description decision problematic [116]. Within this context, the evaluation process is based on a combination of all the criteria describing the performance of the alternatives. Such a

Department of Production Engineering and Management, Technical University of Crete, University Campus, 73100 Chania, Greece e-mail: mdoumpos; kostas@dpem.tuc.gr

C. Zopounidis and P.M. Pardalos (eds.), *Handbook of Multicriteria Analysis*, Applied Optimization 103, DOI 10.1007/978-3-540-92828-7_7, © Springer-Verlag Berlin Heidelberg 2010

combination, however, cannot be meaningful within a given decision context, unless it is able to represent (with some acceptable accuracy) the DM's judgment policy. This can be achieved in two quite different ways.

The first, is a "forward" approach based on interactive, structured communication sessions between the analyst and the DM, during which the analyst elicits specific information about the DM's preferences (e.g., weights, trade-offs, aspiration levels, etc.). The success of this approach is heavily based on the willingness of the DM to participate actively in the process, as well as the ability of the analyst to guide the interactive process in order to address the DM's cognitive limitations. This kind of approach is widely used in situations involving decisions of strategic character.

However, depending on the selected criteria aggregation model, a considerable amount of information may be needed by the DM. In "repetitive" decisions, where time limitations exist, the above direct approach may not be applicable. Disaggregation methods [72] are very helpful in this context. Disaggregation methods use regression-like techniques to infer a decision model from a set of decision examples on some reference alternatives, so that the model is as consistent as possible with the actual evaluation of the alternatives by the DM. This model inference approach provides a starting basis for the decision-aiding process. If the obtained model's parameters are in accordance with the actual preferential system of the DM, then the model can be directly applied to new decision instances. On the other hand, if the model is consistent with the sample decisions, but its parameters are inconsistent with the DM's preferential system (which may happen if, for example, the decision examples are inadequate), then the DM has a starting basis upon which he/she can provide recommendations to the analyst about the calibration of the model in the form of constraints about the parameters of the model. Thus, starting with a model that is consistent with a set of reference examples, an interactive model calibration process is invoked.

Similarly to disaggregation analysis, statistical learning and data mining are also involved with learning from examples [61, 62]. Many advances have been made within these fields for regression, classification and clustering problems. Recently there has been a growing interesting among machine learning researchers towards preference modeling and decision-making. Some interest has also been developed by MCDA researchers on exploiting the advances in machine learning.

Given the growing interest on the integration of the two fields, the objective of this chapter is to explore their connections, to highlight their similarities and differences and analyze the potential from their integration towards providing improved decision support.

The rest of the chapter is organized as follows: We begin with an introduction to disaggregation paradigm of MCDA in section 7.2, followed by an introduction to statistical learning and data mining (section 7.3). Then, section 7.4 discusses the differences and the similarities between the two fields, whereas section 7.5 provides a literature review on the interactions between them. Finally, section 7.6 concludes the chapter and discusses some future research directions.

7.2 The Disaggregation Approach in MCDA

7.2.1 General Framework

Disaggregation analysis (DA) provides a general methodological framework for the analysis of the actual decisions taken by a DM so that an appropriate model can be constructed representing the DM's system of preferences, as consistently as possible. The main input used in this process is a reference set of alternatives evaluated by the DM. The reference set may consist of past decisions, a subset of the alternatives under consideration, or a set of fictitious alternatives which can be easily judged by the DM [72]. Depending on the decision problematic, the evaluation of the reference alternatives may be expressed by defining an order structure (total, weak, partial, etc. [106]) or by classifying them into appropriate classes.

Formally, let $\mathscr{D}(X)$ denote the DM's evaluation of a set X consisting of m reference alternatives described over n criteria (the description of alternative i on criterion j will henceforth be denoted by x_{ij}). The DM's evaluation is assumed to be based (implicitly) on a decision model f_β defined by some parameters β, which represents the actual preferential system of the DM. Different classes of models can be considered. Typical examples include:

- Value functions defined such that $V(\mathbf{x}) > V(\mathbf{y})$ iff alternative $\mathbf{x}$ is preferred over alternative $\mathbf{y}$ and $V(\mathbf{x}) = V(\mathbf{y})$ in cases of indifference [77]. The parameters of a value function model involve the criteria tradeoffs and the form of the marginal value functions.
- Outranking relations defined such that $\mathbf{x} S \mathbf{y}$ iff alternative $\mathbf{x}$ is at least as good as alternative $\mathbf{y}$ [115]. Depending on the specific method used, the parameters of an outranking model, may involve the weights of the criteria, as well as preference, indifference and veto thresholds, etc.
- "If ... then ..." decision rules [53]. In this case the parameters of the model involve the conditions and the conclusions associated to each rule.

The objective of DA is to infer the "optimal" parameters $\widehat{\beta}^*$ that approximate, as accurately as possible, the actual preferential system of the DM as represented in the unknown set of parameters β, i.e.:

$$\widehat{\beta}^* = \arg\min_{\widehat{\beta} \in \mathscr{A}} \|\widehat{\beta} - \beta\| \tag{7.1}$$

where $\mathscr{A}$ is a set of feasible values for the parameters $\widehat{\beta}$. With the obtained parameters, the evaluations performed with the corresponding decision model $f_{\widehat{\beta}^*}$ will be consistent with the evaluations actually performed by the DM for any set of alternatives.

However, problem (7.1) cannot be solved explicitly because β is unknown. Instead, an empirical estimation approach is employed using the DM's evaluation of the reference alternatives to proxy β. Thus, the general form of the optimization problem is now expressed as follows:

$$\widehat{\beta}^* = \arg\min_{\widehat{\beta}\in\mathscr{A}} L[\mathscr{D}(X), \widehat{\mathscr{D}}(X, f_{\widehat{\beta}})] \tag{7.2}$$

where $\widehat{\mathscr{D}}(X, f_{\widehat{\beta}})$ denotes the recommendations of the model $f_{\widehat{\beta}}$ for the alternatives in X and $L(\cdot)$ is a function that measures the differences between $\mathscr{D}(X)$ and $\widehat{\mathscr{D}}(X, f_{\widehat{\beta}})$.

Through the solution of (7.2), it is implicitly assumed that the decision model's estimated parameters $\widehat{\beta}^*$ represent the actual preferential system of the DM within some acceptable error threshold $\varepsilon > 0$, i.e., $\|\widehat{\beta}^* - \beta\| < \varepsilon$. This, however, may not be true for a number of reasons related to the quality of the reference set (e.g., too small, noisy, etc.). Thus, problems (7.1) and (7.2) are not necessarily equivalent in a realistic setting.

7.2.2 Methods and Implementations

The general framework of DA is materialized in several MCDA methods that enable the development of decision models in different forms. This section focus on two popular paradigms, which involve functional and relational models. Symbolic models have also become quite popular recently. However, given their close connections with machine learning methods, the discussion of this modeling form is given later in section 7.5.1.2.

7.2.2.1 Functional Models

Value functions are the most widely used type of functional models in MCDA. A value function aggregates all the criteria into an overall performance measure V defined such that:

$$\begin{aligned} V(\mathbf{x}) > V(\mathbf{y}) &\Leftrightarrow \mathbf{x} \succ \mathbf{y} \\ V(\mathbf{x}) = V(\mathbf{y}) &\Leftrightarrow \mathbf{x} \sim \mathbf{y} \end{aligned} \tag{7.3}$$

where $\succ$ and $\sim$ denote the preference and indifference relations, respectively. A value function may expressed in different forms, depending on the criteria independence conditions that describe the DM's preferences [77]. Due to its simplicity, the most widely used form of value function is the additive one:

$$V(\mathbf{x}) = \sum_{j=1}^{n} w_j v_j(x_j) \tag{7.4}$$

where $w_1, \ldots, w_n$ are non-negative constants representing the criteria tradeoffs ($w_1 + \cdots + w_n = 1$) and $v_1(x_1), \ldots, v_n(x_n)$ are the marginal value functions of the criteria, usually scaled such that $v_j(x_{j*}) = 0$ and $v_j(x_j^*) = 1$, where x_{j*} and x_j^* are the least and the most preferred level of criterion j, respectively.

Such a model can be used to rank a set of alternatives or to classify them in predefined groups. In the ranking case, the relationships (7.3) provide a straightforward way to compare the alternatives. In the classification case, the simplest approach is to define groups $G_1, G_2, \dots, G_q$ in the value scale with the following rule:

$$t_k \leq V(\mathbf{x}) < t_{k-1} \Leftrightarrow \mathbf{x} \in G_k \tag{7.5}$$

where $0 = t_q < t_{q-1} < \cdots < t_1 < t_0 = 1$ are thresholds that distinguish the groups.

The construction of a value function from a set of reference examples can be performed using mathematical programming techniques. For example, in an ordinal regression setting, the DM's defines a weak-order of the alternatives in the reference set, by ranking them from the best one (alternative $\mathbf{x}_1$) to the worst one (alternative $\mathbf{x}_m$). Then, the general form of the optimization problem can be expressed as in the case of the UTA method [71] as follows:

$$\begin{aligned} \min \ & \sum_{i=1}^{m} \sigma_i \\ \text{s.t.} \ & \sum_{j=1}^{n} [v_j(x_{ij}) - v_j(x_{i+1,j})] + \sigma_i - \sigma_{i+1} \geq \delta, \ \forall \mathbf{x}_i \succ \mathbf{x}_{i+1} \\ & \sum_{j=1}^{n} [v_j(x_{ij}) - v_j(x_{i+1,j})] + \sigma_i - \sigma_{i+1} = 0, \ \forall \mathbf{x}_i \sim \mathbf{x}_{i+1} \\ & v_j(x_j) \text{ non-decreasing, with } v_j(x_{j*}) = 0 \text{ and } \sum_{j=1}^{n} v_j(x_j^*) = 1 \\ & \sigma_i \geq 0, \quad \forall i \end{aligned} \tag{7.6}$$

The solution of this optimization problem provides an additive value function that reproduces the DM's ranking of the reference alternatives as accurately as possible. The differences between the model's recommendations and the DM's weak-order are measured by the error variables $\sigma_1, \dots, \sigma_m$. In this case the value function is expressed in pure additive form as:

$$V(\mathbf{x}) = v_1(x_1) + \cdots + v_n(x_n) \tag{7.7}$$

where the marginal value functions are now scaled such that $v_j(x_{j*}) = 0$ and $v_j(x_j^*) = w_j$. By modeling the marginal values as piecewise linear functions, the above optimization problem can be re-expressed in linear programming form (for the details see [71]).

Several variants of the UTA method for ordinal regression problems have been presented. Siskos et al. [125] provide a detailed review of different formulations, whereas Beuthe and Scannella [13] present a comparative analysis. Some recent extensions are presented by Figueira et al. [41] and Greco et al. [56]. Formulations for classification problems have also been developed, such as the UTADIS method and its variants [34, 37, 71, 80], the MHDIS method [150], and other similar approaches [19, 28, 81].

The optimization processes most often used involve linear and integer programming formulations (LP, IP). LP models are usually used to minimize some predefined norm (L_1 or L_∞) of real-valued error variables representing the violations of (7.3) or (7.5). The optimization problem (7.6) is an example using the L_1 norm of the error variables $\sigma_1, \dots, \sigma_m$ for the reference alternatives. IP formulations on the other hand, consider more direct measures of the number of disagreements between the recommendations of the estimated decision model and the actual evaluation of the reference alternatives by the DM. The Kendall's τ rank correlation coefficient is a typical example of such a measure.

It is also worth mentioning the considerable recent research on extending this modeling framework, which is based on simple form of value functions, towards more general preference modeling forms that allow the consideration of interaction between the criteria. The use of the Choquet integral as an aggregation function has proved quite useful and convenient towards this direction. Marichal and Roubens [93] first introduced a methodology implementing this approach in a disaggregation context. Some works on this topic can be found in the works of Angilella et al. [5] and Kojadinovic [78, 79], while a review of this topic has been presented by Grabisch et al. [51].

7.2.2.2 Relational Models

The evaluations performed on the basis of value functions are transitive and complete. In several cases, however, preferences do not satisfy these properties. Intransitivity is often observed and furthermore the alternatives can be incomparable. Relational models enable the modeling of such situations. The outranking relations theory of MCDA [115] describes such models, with close connections to social choice theory.

Typically, an outranking relation S between a pair of alternatives $\mathbf{x}$ and $\mathbf{y}$ is defined as:

$$\mathbf{x} S \mathbf{y} \Leftrightarrow \mathbf{x} \text{ is at least as good as } \mathbf{y} \tag{7.8}$$

Outranking techniques operate in two stages. The first stage involves the pairwise comparison of the alternatives. Then, an algorithmic procedure is used in the second stage to derive the evaluation results from the pairwise comparisons of the first stage.

There are several outranking methods that implement the above framework in different ways. The most widely used include the families of ELECTRE [115] and PROMETHEE methods [10, 14]. Martel and Matarazzo [94] provide a comprehensive review of other outranking approaches. Other non-outranking relational models based on distances have been presented by Chen et al. [23, 24].

In contrast to a value function approach, outranking models usually require too many parameters, which define the decision model in a complex non-linear way. This fact poses a significant computational burden in eliciting the preferential parameters from decision examples. With some simplifying assumptions this issue can be resolved. For instance, Mousseau et al. [98], Dias et al. [29], Ngo The and Mousseau [101], and Dias and Mousseau [30] developed several LP simplifications and heuristics to infer some of the parameters of pessimistic ELECTRE TRI models, while assuming the others fixed. Conventional optimization approaches (LP and quadratic programming) are generally applicable for simpler forms of outranking/relational models that implement a compensatory approach (see for instance [23, 24, 36].

A first attempt to develop an "holistic" approach for more complex outranking models was presented by Mousseau and Slowinski [99] for the ELECTRE TRI method. Similarly to the previous studies, they assumed the pessimistic assignment rule, and developed a non-linear, non-convex optimization formulation to infer all the parameters of a classification decision model from a set of assignment examples. Later, Doumpos and Zopounidis [35] presented an alternative approach combining heuristic rules with LP formulations. Recently, metaheuristics and evolutionary approaches have been used. Goletsis et al. [49] used a genetic algorithm for the development of an outranking model in a two-group problem involving ischemic beat classification. Fernandez et al. [40] used a multiobjective genetic optimization approach for constructing an outranking classification model, whereas Belacel et al. [11] used the reduced variable neighborhood search metaheuristic to infer the parameters of the PROAFTN method from a set of reference examples, and Doumpos et al. [33] used the differential evolution algorithm to develop classification models based on the ELECTRE TRI.

7.3 Statistical Learning and Data Mining

7.3.1 General Framework

Hand et al. [61] define data mining as "*the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner*".

Statistical learning plays an important role in the data mining process, by describing the theory that underlies the identification of such relationships and providing the necessary analysis techniques. According to Vapnik [135, 136] the process of learning from examples includes three main components:

1. A set X of data vectors $\mathbf{x}$ drawn independently from a probability distribution $P(\mathbf{x})$. This distribution is assumed to be unknown, thus implying that there is no control on how the data are observed [128].
2. An output y from a set Y, which is defined for every input $\mathbf{x}$ according to an unknown conditional distribution function $P(y \mid \mathbf{x})$. This implies that the relationship between the input data and the outputs is unknown.

3. A learning method (machine), which is able to assign a function $f_\beta : X \rightarrow Y$, where β are some parameters of the unknown function.

The best function f_β is the one that best approximates the actual outputs, i.e., the one that minimizes:

$$\int L[y, f_\beta(\mathbf{x})] dP(\mathbf{x}, y) \tag{7.9}$$

where $L[y, f_\beta(\mathbf{x})]$ is a function of the differences between the actual output y and the estimate $f_\beta(\mathbf{x})$,[1] and $P(\mathbf{x}, y) = P(\mathbf{x})P(y \mid \mathbf{x})$ is the joint probability distribution of $\mathbf{x}$ and y. However, this joint distribution is unknown and the only available information is contained in a training set of m objects $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$, which are assumed to be generated independently from this unknown distribution. Thus, the objective (7.9) is substituted by its empirical estimate:

$$\frac{1}{m} \sum_{i=1}^{m} L[y_i, f_\beta(\mathbf{x}_i)] \tag{7.10}$$

For a class of functions f_β of a given complexity, the minimization of (7.10) leads to the minimization of an upper bound for (7.9).

7.3.2 Methods

One of the main research directions in statistical learning involves the study of the theoretical properties of the optimization problem (7.10) in order to derive data independent bounds of the generalization performance of learning machines (for a complete coverage of the theoretical aspects of statistical learning, see [136]). The other main direction involves the development of efficient learning methods and algorithms. While a full review of all methods and algorithms is out of the scope of this chapter (detailed presentations are available in [61, 62]), a brief outline of some popular schemes is given below.

7.3.3 Neural Networks

Neural networks have been one of the most popular approaches in statistical learning and data mining. Neural networks are based on an "artificial" representation of the human brain, through a directed acyclic graph with nodes (neurons) organized into layers. In a typical feed-forward architecture, there is an layer of input nodes, a layer of output nodes, and a series of intermediate layers. The input nodes correspond to

[1] The specification of the loss function L depends on the problem under consideration. For instance, in a regression setting it may correspond to the mean squared error, whereas in a classification context it may represent the accuracy rate.

the information that is available for every input vector (the attributes/independent variables), whereas the output nodes provide the recommendations of the network. The nodes in the intermediate (hidden) layers are parallel processing units that define the input-output relationship. Every neuron at a given layer receives as input the weighted average of the outputs of the neurons at the preceding layer and maps it to an output signal through a predefined transformation function. Depending on the topology of the network and the selection of the neurons' transformation functions, a neural network can model real functions of arbitrary complexity. This flexibility has made neural networks a very popular modeling approach in addressing complex real-world problems in engineering and management.

Training a neural network involves the optimization of the connections' weights. In a supervised learning context, the optimization is based on a training set, in accordance with the general framework of statistical learning. Unconstrained non-linear optimization algorithms are commonly used in this context [63]. Evolutionary techniques have also been recently used [1].

7.3.4 Decision Trees and Rule-Based Models

Symbolic models expressed as decision trees and rule sets are quite popular among machine learning researchers and practitioners, mainly due to their interpretability. Typically, the nodes of a decision tree represent a series of (usually) binary splits defined on the independent variables, while the recommendations of the model are given at the terminal nodes of the tree. Decision tree models can also be expressed in a rule-based format of the form of "If ... then ..." decision rules. The first part of a given rule examines the necessary conditions required for the conclusion part to be valid. The conclusion provides the recommendation (output) of the rule. Except for the easy interpretation and use of such models by DMs and analysts, other advantages also include their ability to handle different types of data (quantitative or qualitative), the handling of missing data, as well as their applications in discovering interesting relations between variables in large databases (e.g., through the development of association rules [4]).

Some typical examples of algorithms used to build decision trees and rule-based models include, among others, ID3 [112] and its successor C4.5 [113], as well as CART [18], CHAID [76], and rough sets [108].

7.3.5 Support Vector Machines

Support vector machines (SVMs) have become increasingly popular among the statistical learning community. SVMs implement the structural risk minimization principle taking into consideration the empirical loss (7.10), while controlling the complexity of the model through a Tikhonov regularization approach. The empirical error can be minimized with a highly complex model, but in such cases the model is usually unstable to the selection of the training set, and consequently its generalizing ability is poor. SVMs introduce this tradeoff in the analysis providing a unified framework for both linear and non-linear models.

SVMs are usually realized in a binary classification setting, but they are also applicable in multi-group classification, regression, and clustering problems. In the simplest case involving a binary classification task, the two groups are defined by the canonical hyperplanes $a+\mathbf{wx}=\pm 1$ (where a is a constant term and $\mathbf{w}$ is the normal vector for the hyperplane), such that $a+\mathbf{wx}\geq 1$ for the positive examples and $a+\mathbf{wx}\leq -1$ for the negative ones. The distance (separating margin) between the two hyperplane is then $2/\|\mathbf{w}\|$ and it is related to an upper bound of the probability that an observation will be misclassified (the higher the margin the lower the misclassification probability [120]). Extending, this reasoning by introducing the classification errors, leads to the following convex quadratic programming formulation:

$$\begin{array}{ll} \min & \frac{1}{2}\mathbf{w}^{\top}\mathbf{w}+C\sum_{i=1}^{m}\sigma_i \\ \text{s.t.} & y_i(a+\mathbf{wx}_i)+\sigma_i\geq 1,\ \forall i \\ & \sigma_i\geq 0, \qquad\qquad\qquad \forall i \\ & a,\mathbf{w}\in\mathbb{R} \end{array} \tag{7.11}$$

where $y_i=\pm 1$ denotes the class label for observation i, σ_i is the corresponding slack variable defined such that $\sigma_i>0$ iff $y_i(a+\mathbf{wx}_i)<1$, and $C>0$ is a user-defined constant that defines the trade-off between the two conflicting objectives (margin maximization and error minimization).

The generalization to the nonlinear case is achieved by mapping the problem data to a higher dimensional space H (feature space) through a transformation of the form $\mathbf{x}_i\mathbf{x}_j^{\top}=\phi(\mathbf{x}_i)\phi^{\top}(\mathbf{x}_j)$. The mapping function ϕ is implicitly defined through a symmetric positive definite kernel function $K(\mathbf{x}_i,\mathbf{x}_j)=\phi(\mathbf{x}_i)\phi^{\top}(\mathbf{x}_j)$. The representation of the data using the kernel function enables the development of a linear model in the feature space H.

For large training sets several computational procedures have been proposed to enable the fast training of SVM models. Most of these procedures are based on a decomposition scheme, where the optimization problem is split into a series of smaller subproblems. Other algorithms are based on reformulations of the optimization problem that enable the development of the model through the solution of a set of linear equations. Linear programming formulations have also been used.

A detailed presentation of SVMs, the theory of kernel methods, the existing optimization tools, and applications, can be found in the book of Schölkopf and Smola [120] as well as the review paper of Campbell [20].

7.3.6 Ensembles

Individual learning algorithms often exhibit instability to the training data. This may lead to high bias/variance and poor generalizing performance. A popular approach to address this issue is to combine multiple models, thus forming ensemble predictors. This is not a new issue, since the first works on the combination of regression forecasts can be traced back to the work of Bates and Granger [9]. Recently, theoretical evidences have been presented (no free lunch theorems [146, 147, 148]) showing that there is no method that is universally better than others in terms of its predictive performance. On the basis of such findings, it is natural to investigate the potential of a method/model combination framework. Of course, combined models are also subject to the no free lunch theorem. However, the development of combined models aims at the reduction of the bias and/or variance of the individual models, which is expected to be useful in improving the results in real-world situations. The combination is most useful when the predictions of the models which are combined have low correlations to each other.

Since the 1990s there has been a considerable growth in the research on model combination approaches and several algorithm-independent approaches have been proposed that exploit the instability of statistical learning models and the differences between methods. Some approaches combine multiple models of the same learning method, each developed using different perturbations of the training set [15, 16, 43], while other approaches enable the combination of models from multiple methods [45, 145]. A review of different ensemble approaches can be found in [31].

7.4 Similarities and Differences

Disaggregation analysis and statistical learning have evolved significantly over the past two decades, as two separate fields. Nevertheless, the similarities between the two fields are obvious, since both consider the problem of learning a decision/prediction model from data. Within this context, it is worth noting that the minimization of the empirical loss (7.10) in statistical learning methods is actually identical to the optimization problem (7.2), which is commonly used in disaggregation methods. This may lead to the conclusion that both fields actually address the same problem. But there are a series of noticeable differences.

1. *Model interpretability*: The interpretability of MCDA models is of outmost importance. Interpretable and easy to understand decision models enable the DM's active participation in the decision-aiding process, they provide insights on the characteristics of the alternatives, and the DMs often feel more confident with them in their daily practice. On the other hand, statistical learning theory has mostly focused on the development of models of high predicting ability. In most cases these models are too complex to interpret (e.g., neural networks, non-linear

SVMs, ensembles, etc.), and consequently their operation has often been described as a "black box".

2. *Data dimensionality*: Real world applications of statistical learning methods involve large (often massive) data sets and considerable research has been devoted to the development of computationally efficient algorithms that scale up well with the size of the training data. DA methods, on the other, usually assume that only a small reference set is available, since it is difficult for the DMs to express their global preferences on too many alternatives.
3. *Inconsistencies*: In DA, data inconsistencies are usually explicitly treated during the model development process [50, 88, 96, 97]. This is done through interactive procedures whose objective is to reveal the inconsistencies to the DM, to support their resolution, and to enhance the DM's understanding of the problem data. Contrary to this approach, data mining treats inconsistencies in the training data as "hard cases", i.e., observations which are simply difficult to learn and predict. Only outliers are treated as real inconsistencies.
4. *Model validation*: While model validation is used in both DA and statistical learning to check the quality of the model, the implementation of the validation process differs. In DA it is usually assumed that the analyst cooperates with the DM and the validation is an interactive process, during which the DM checks the validity of the parameters' estimation results. The generalizing ability of the model, is on the other hand, the core issue in the validation stage of all statistical learning models. This is tested using additional data sets, outside the training sample, or through resampling methods (e.g., cross-validation, bootstrap, etc.).

The differences between statistical learning and disaggregation methods in MCDA have also been discussed by Waegeman et al. [138]. In addition to the above points, the authors have also noted some other interesting issues:

1. *The role of the DM*: In MCDA, the DM participates actively in the decision modeling process and interacts with the analyst in order to achieve the best calibration of the decision model. This interactive process is dynamic in nature, in that the DM's preferences may change as he/she gains insight to the problem data and its characteristics. Statistical learning and data mining on the other hand, assume that only a statistical sample, whereas specific inputs from the DM are not.
2. *Regularization*: The traditional MCDA disaggregation methods usually do not take into account the trade-off between model complexity and model performance. On the other hand, regularization has become a crucial issue in statistical learning processes.
3. *Data type*: The data considered in an MCDA setting involve the description of the alternatives over a number of criteria. In addition to this type of data setting, statistical learning is also concerned with more complex structures, which are often encountered in areas such as text mining, image analysis, and signal processing.

Overall, these differences really seem quite fundamental; an indeed they are. But the consideration of these differences, should take into account the crucial aspect of the scope of the application of DA methods as opposed to the common uses of data mining.

DA methods are used in a MCDA context to facilitate the decision support process. In particular, the main objective of eliciting preferential information through decision examples is to facilitate the DM in gaining insight into: (1) the characteristics of the problem data (alternatives and criteria), (2) the implications of the judgments that (s)he implicitly makes, (3) the characteristics and limitations of the modeling process, (4) the interpretations of the results, and ultimately (4) the actions that need to be taken in order to obtain good decisions through a practical model.

On the other hand, modern statistical learning and data mining adopt an *algorithmic modeling culture* as described by Breiman [17], in which the focus is shifted from data models to the characteristics and predictive performance of learning algorithms. In this framework, the data generation process in a real problem is a "black box whose insides are complex, mysterious and, at least, partly unknowable" [17], thus leading to the important issue of developing efficient algorithms that provide accurate predictions of the observed outcome from some given input data.

7.5 Interactions

Despite the development of MCDA and statistical learning/data mining as separate fields, and the differences outlined in the previous section, there have been several attempts to integrate concepts and methods from the two fields. This section reviews this emerging research stream and its potential towards the development of improved decision support methodologies. The interactions of the two fields are examined in two opposite directions. The first involves the use of statistical learning and data mining techniques in a decision aiding context through disaggregation analysis and preference learning. The second direction involves the implementation of MCDA concepts in a statistical learning framework and the development of hybrid methodologies.

7.5.1 Using Statistical Learning Methods for Disaggregation Analysis and MCDA

7.5.1.1 Neural Networks

One of the main advantages of neural network (NN) models is their ability to model highly complex problems, with an unknown underlying structure. This characteristic has important implications for MCDA, mainly with respect to modeling general preference structures.

Within this context, NNs have been successfully used for learning generalized MCDA models from decision examples. Wang and Malakooti [141], and Malakooti and Zhou [91] used feedforward NNs to learn an arbitrary value function for ranking a set of alternatives, as well as to learn a relational multicriteria model based on pairwise comparisons (binary relations) among the alternatives. The main advantage of this NN-based approach, is that the resulting decision models are free of the various independence assumptions, which are often implied by commonly used value function models (see [77]). Thus, the model is independent of functional form and quite stable to parameter perturbations. The authors examined the conditions that characterize the monotonicity of the NN model, as well as its convexity/concavity properties. The monotonicity condition of the form (7.3) is a fundamental property for any rational decision model. On the other hand, the convexity/concavity properties are very useful for calibrating the model development (training) process in order to ensure that the final model complies with the DM's preference policy. Experimental simulation results showed that NN trained models performed very well in representing various forms of decision models, outperforming other popular model development techniques based on linear programming formulations. Wang et al. [142] applied a similar NN model to a job shop production system problem.

In a different framework compared to the aforementioned studies, Stam et al. [127] used NNs within the context of the analytic hierarchy process (AHP) [117]. AHP is based on a hierarchical structuring of the decision problem, with the overall goal on the top of the hierarchy and the alternatives at the bottom. With this hierarchical structure, the DM is asked to perform pairwise comparisons of the elements at each level of the hierarchy with respect to the elements of the preceding (higher) level. The principal eigenvalues and the corresponding normalized eigenvectors of the resulting reciprocal pairwise comparison matrices are then used to obtain preference ratings for the alternatives. This eigenvector-based approach has received much criticism (see for example [8]). Stam at el. investigated two different NN structures for accurately approximating the preferences ratings of the alternatives, within the context of imprecise preference judgments by the DM. They showed that a modified Hopfield network has very close connections to the mechanics of the AHP, but found that this network formulation cannot provide good results in estimating the mapping from a positive reciprocal pairwise comparison matrix to its preference rating vector. On the other hand, a feed-forward NN model was found to provide very good approximations of the preference ratings in the presence of impreciseness. This NN model was actually superior to the standard principal eigenvector method.

Similar NN-based methodologies have also be used to address dynamic MCDA problems (where the DM's preferences change over time) [90], to learn fuzzy preferences [139, 140, 143] and outranking relations [67], to provide support in group decision making problems [143], as well as in multicriteria clustering [89].

NNs have also been employed for preference representation and learning in multiobjective optimization (MOP). The main goal in a MOP problem is to identify the set of non-dominated solutions (Pareto optimal solutions) and then to select the most appropriate one that best fits the DM's preferences. Within this context, Sun et al. [130] proposed a feed-forward NN model, which is trained to represent the DM's preference structure. The training of the model is performed using a representative sample of non-dominated solutions, which are evaluated by the DM. The flexibility of NNs enables them to model complex preference structures, even highly nonlinear ones. Thus, the trained NN model is used to formulate the objective function of a nonlinear programming problem, which is solved in order to find a solution that maximizes the output of the trained NN. A similar NN optimization formulation has also been proposed by Chen and Lin [21], while Shimizu et al. [122] presented a web-based implementation integrating a NN model with AHP. Such approaches are generally similar to techniques proposed in the MOP literature based on traditional value function models (see for example [124]). Despite the good results obtained with this approach and its robustness to the NN's architecture, the solution of the nonlinear optimization problem having the NN's output as the objective is often cumbersome. To overcome this difficulty, Sun et al. [131] presented a hybrid methodology combining the feed-forward NN model with the interactive weighted Tchebycheff procedure (IWTP) [129]. In this case a trained NN model is used to evaluate a set of nondominated solutions and to select the ones that are most likely to be of interest to the DM, thus supporting the interactive search procedure. Results on various test problems characterized with different underlying value functions (linear and nonlinear) indicated that the use of the NN-based approach provided improved results compared to IWTP and a NN-based optimization model. Other NN architectures have also been used as optimizers in MOP problems [48, 87, 95, 144] and hybrid evaluation systems [6, 111, 114, 121].

7.5.1.2 Rule-based Models

Rule-based and decision tree models are very popular within the machine learning research community. The symbolic nature of such models makes them easy to understand, which usually is a very important characteristic in decision aiding problems. During the last decade significant research has been devoted on the use of such approaches as preference modeling tools in MCDA and disaggregation analysis.

Within this framework there has been proposed a complete and well-axiomatized methodology for constructing decision rule preference models from decision examples, based on the rough sets theory [108, 109]. Rough sets have been initially introduced as a methodology to describe dependencies between attributes, to evaluate the significance of attributes and to deal with inconsistent data in multicriteria decision problems. However, over the past decade significant research has been conducted on the use of the rough set approach as a methodology for preference modeling in multicriteria decision problems [52, 53]. The main novelty of this approach concerns the possibility of handling criteria, i.e. attributes with preference ordered domains, and preference ordered classes in the analysis of sorting examples and the induction of decision rules. The rough approximations of decision classes involve the dominance relation, instead of the indiscernibility relation considered in the basic rough sets approach. They are build of reference alternatives given in the decision examples. Decision rules derived from these approximations constitute a preference model. Each “if ... then ...” decision rule is composed of a condition part specifying a partial profile on a subset of criteria to which an alternative is compared using the dominance relation, and a decision part suggesting an assignment of the alternative to “at least” or “at most” a given class.

The decision rule preference model has also been considered in terms of conjoint measurement [55] and Bayesian decision theory [57]. A representation theorem [55] for multicriteria sorting states an equivalence of simple cancellation property, a general discriminant (sorting) function and a specific outranking relation, on the one hand, and the decision rule model on the other hand. It is also shown that the decision rule model resulting from the dominance-based rough set approach has an advantage over the usual functional and relational models because it permits handling inconsistent sorting examples. The inconsistency in sorting examples is not unusual due to instability of preference, incomplete determination of criteria and hesitation of the DM.

An important feature of this methodology is that its applicability is not restricted to multicriteria classification problems, but is also extended to ranking and choice decision problems [42, 53], as well as to MOP problems [56]. It also provides the ability to work with missing data and to handle cases that involved both criteria and attributes (whose domains are not preference ordered; see [54]).

A similar approach that implements symbolic models has also been presented by Dombi and Zsiros [32], while Hammer et al. [60] modified the LAD method (logical analysis of data), which is based on the theory of boolean functions, to address multicriteria classification problems through the introduction of Pareto-optimal patterns.

7.5.1.3 Kernel Methods and Margin-Based Approaches

Kernel methods become an important research direction in statistical learning and they are now widely used for classification and regression models, as well as for density estimation. Kernel methods map the problem data to a high dimensional space (feature space), thus enabling the development of complex nonlinear decision and prediction models, using linear estimation methods [65, 119]. This kind of representation is based on a positive definite kernel function, which corresponds to a dot product in the feature space. The main novelty of the introduction of the kernel function is that it makes the explicit computation of the feature space unnecessary.

SVMs are one of the most common implementations of the theory of kernel methods, with numerous application in pattern recognition problems. Recently, they have also been used within the context of preference learning for approximating arbitrary utility/value functions and preference aggregation.

Herbrich et al. [64] illustrated the use of kernel approaches, within the context of SVM formulations, for representing value/ranking functions of the generalized form $V(\mathbf{x}) = \mathbf{w}\phi(\mathbf{x})$, where ϕ is a possibly infinite-dimensional and in general unknown feature mapping. The authors derived bounds on the generalizing performance of the estimated ranking models, based on the margin separating objects in consecutive ranks. A similar approach was also explored by Joachims [75] in the RankSVM algorithm, which has been used to improve the retrieval quality of search engines.

Waegeman et al. [138] extend this approach to relational models. In this case, the preference model of the form $f(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{w}\phi(\mathbf{x}_i, \mathbf{x}_j)$ is developed to represent the preference of alternative i compared to alternative j. This framework is general enough to accommodate special modeling forms. For instance, it includes value models as a special case, and similar techniques can also be used to kernelize Choquet integrals. As an example, Waegeman et al. illustrated the potential of this framework in the case of valued concordance relations, which are used in the ELECTRE methods.

Together with the use the kernel approach for the development of generalized decision models, an additional important feature of SVM formulations, is the implementation of the regularization concept to handle the complexity of the models. Evgeniou et al. [39] gave an interpretation of this regularization approach within the context of estimating a linear value function $V(\mathbf{x}) = \mathbf{w}\mathbf{x}$ used for ranking purposes (ordinal regression). From a geometric point of view, assuming the simplest case where the reference data are representable by such a linear model with no errors, the value function that minimizes $\|\mathbf{w}\|$ corresponds to the most robust solution in the feasible space defined by constraints of the form:

$$\mathbf{w}(\mathbf{x}_i - \mathbf{x}_j) \geq 1, \quad \forall\, \mathbf{x}_i \succ \mathbf{x}_j$$

The term "robust" in this case refers to the solution that is the center of the largest sphere in the polyhedron defined by the constraints [39]. In the general case, when the reference data include some inconsistent comparisons, Evgeniou et al. intuitively explained the minimization of the violations of the above constraints together with the minimization of $\|\mathbf{w}\|$, as the search of a decision model that minimizes the errors compared to the DM's judgments, while satisfying the correct comparisons as much as possible. The authors also describe the generalization of this framework to nonlinear (polynomial) value function models.

Doumpos and Zopounidis [37] analyzed a similar methodology for the construction of additive value functions, using the L_1 norm of the parameters of the function. They showed that formulating an augmented linear objective function considering both the errors of the model and its complexity, leads to interesting insights on the quality of the reference set. Experimental results on both ranking and classification problems showed that such a modification improves both the generalizing performance of the obtained decision models and their robustness.

In a different context, Dembczynski et al. [28] combined concepts from the dominance-based rough set approach and SVMs towards the development of additive value function models in classification problems. Contrary to the methodology of Doumpos and Zopounidis [37], the authors used a regularization term based on the L_2 norm, and illustrated how the formulation can be expressed in kernel form.

The margin maximization principle, which is implemented in SVMs as a regularization mechanism, has also been explored in connection to some ensemble algorithms, such as boosting [43, 118]. Within this context, Freund et al. [44] developed the RankBoost algorithm, which provides a single linear ordering of the given set of objects by combining a set of given linear orderings on a set of ranking features. Instead of using the evaluation criteria as ranking features, Freund et al. developed an algorithm to combine multiple "weak" rankings defined over the criteria, in a weighted additive model (similar to an additive value function).

7.5.2 MCDA Concepts in Statistical Learning Models

7.5.2.1 MCDA Methodologies for Building Statistical Learning Models

As mentioned in section 7.3.1, the development of statistical learning models is based on the minimization of a loss function measured on the basis of a set of training samples. This general setting, however, can be implemented in many different ways. The variety of loss functions employed in different models indicates that model development is not based on a straightforward, universally accepted criterion. For instance, in regression problems measures such as the mean squared error or the mean absolute error may lead to completely different models. In classification problems, similar L_1, L_2 and L_∞ norms have been used, together with measures such as the classification error rate or the area under the receiver operating characteristic curve. The introduction of the regularization terms also adds complexity and degrees of freedom on the model development process.

Similar "multicriteria" issues also arise in specification of the predictor variables (feature selection and extraction), the construction of ensemble models, the pruning and selection of decision rules, as well as the extension of case-based reasoning models on the basis of generalized distance metrics.

The existing research on all the above topics is indeed quite rich. Some indicative works include:

- Learning through multiobjective optimization [38, 47, 59, 74, 83, 84, 100, 132].
- Feature selection and feature extraction [46, 68, 103, 149].
- Construction of ensemble models [58, 69, 70, 73].
- Pruning and use of decision rules [105, 126].
- Case-based reasoning [85, 86, 107].

7.5.2.2 Model Performance Evaluation

The approaches discussed in the previous subsection aim towards the consideration of multiple performance measures at the model construction (optimization) phase. Obviously, multiple performance measures can also be used when evaluating the suitability and performance of a given set of models. For instance, Osei-Bryson [104] proposed a multicriteria methodology to evaluate decision trees based on criteria related to their discriminatory power, simplicity, and stability, taking into account the DM's subjective judgements on the relative importance of these criteria. In a later study Osei-Bryson applied this modeling framework to the problem of pruning decision trees [105]. In a similar context, Choi et al. [25] and Chen [22] introduced multiple criteria for the evaluation of association rules using MCDA approaches. In a neural network setting, Das [26] used the TOPSIS multicriteria method to evaluate neural network models using multiple performance measures (e.g., mean squared error, the Akaike's information criterion, the Bayesian information criterion, cross-validation accuracy, etc.), whereas Ni et al. [102] used the PROMETHEE multicriteria method to evaluate different neural network architectures developed for the prediction of carbamate concentrations in ternary mixtures.

7.5.2.3 Monotonicity in Predictive Modeling

Monotonicity plays a crucial role in decision modeling and aiding. In simple terms, given two alternatives such that $\mathbf{x}_i \geq \mathbf{x}_j$, the monotonicity principle implies that alternative j cannot be preferred over alternative i. Assuming, for instance, a functional decision model (e.g., value/ranking function) $f(\mathbf{x})$, the monotonicity condition requires that $f(\cdot)$ is monotone with respect to the inputs, i.e., $\mathbf{x}_i \geq \mathbf{x}_j \Rightarrow f(\mathbf{x}_i) \geq f(\mathbf{x}_j)$.

Monotonicity is usually not taken into consideration in a data mining context. However, in several cases, the users of prediction models would like the models not only to predict well, but also to make sense within the context of a specific application domain. Furthermore, studies have shown that the introduction of specific domain knowledge into the statistical learning process, may actually improve the generalizing ability of the obtained models [3, 92, 133], by reducing overfitting, minimizing the effect of noisy data, and controlling the complexity of the model.

Within this context, monotonicity can be considered as a special form of domain knowledge. In simple linear decision models of the form $f(\mathbf{x}) = \mathbf{w}\mathbf{x}$, monotonicity can be easily introduced by imposing the condition $\mathbf{w} \geq \mathbf{0}$, which requires the scaling constants to be non-negative. For generalized non-linear decision models, however, the introduction of monotonic conditions is more involved.

Ben-David et al. [12] were among the first to explore this issue within the context of rule-based models. Some recent works on learning monotonic rule-based models and decision trees can be found in [27, 82, 110, 134]. Studies involving other learning models, such as neural networks and SVMs, include [2, 7, 66, 123, 137].

7.6 Conclusions

Data mining/statistical learning and the disaggregation approach of MCDA, both study similar problems in a different context. Data mining has focused on the development of generalized prediction models from a statistical point of view and statistical learning has focused on the theory of the learning process aiming at the development of scalable algorithms for accurate predictive modeling with large and complex data sets. On the other hand, the disaggregation approach of MCDA has mainly focused on the development of comprehensible decision models from small data sets, whose main objective is to support decision aiding through an interactive model calibration process.

Despite the conceptual and modeling differences in the two paradigms, there are clear connections. This chapter highlighted these connections together with the existing differences. The literature review also shows that the interactions between the two fields have already been explored, thus enabling the development of new improved techniques, which can be used either for predictive purposed in a pure data mining context or for aiding the DMs in complex decision problems.

The road ahead should mainly focus on exploring further ways to integrate the two fields. The use of statistical learning approaches for modeling new types of preference models, the addition of comprehensibility into data mining tools, the issues of validation, regularization, and robustness of preference models developed through DA, the scalability of disaggregation methods to large data sets, and the applications of new models into innovative fields, are only some indicative topics where future research can focus.

References

1. H.A. Abbass. Speeding up backpropagation using multiobjective evolutionary algorithms. *Neural Computation*, 15(11):2705–2726, 2003.
2. Y.S. Abu-Mostafa. Learning from hints in neural networks. *Journal of Complexity*, 6(2):192–198, 1990.
3. Y.S. Abu-Mostafa. Machines that learn from hints. *Scientific American*, 272(4):64–69, 1995.
4. R. Agrawal, T. Imieliński, and A. Swami. Mining association rules between sets of items in large databases. In P. Buneman and S. Jajodia, editors, *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, New York, NY, 1993. ACM.
5. S. Angilella, S. Greco, and B. Matarazzo. Non-additive robust ordinal regression: A multiple criteria decision model based on the Choquet integral. *European Journal of Operational Research*, 201:277–288, 2010.
6. O.U. Araz. A simulation based multi-criteria scheduling approach of dual-resource constrained manufacturing systems with neural networks. *Lecture Notes in Computer Science*, 3809:1047–1052, 2005.
7. N.P. Archer and S. Wang. Application of the back propagation neural network algorithm with monotonicity constraints for two-group classification problems. *Decision Sciences*, 24(1):60–75, 1993.
8. C.A. Bana e Costa and J-C. Vansnick. A critical analysis of the eigenvalue method used to derive priorities in AHP. *European Journal of Operational Research*, 187(3):1422–1428, 2008.
9. J.M. Bates and Granger C.W.J. The combination of forecasts. *Operational Research Quarterly*, 20(4):451–468, 1969.
10. M. Behzadian, R.B. Kazemzadeh, A. Albadvi, and M. Aghdasi. PROMETHEE: A comprehensive literature review on methodologies and applications. *European Journal of Operational Research*, 2009. In press.
11. N. Belacel, H. Bhasker Raval, and A.P. Punnenc. Learning multicriteria fuzzy classification method PROAFTN from data. *Computers and Operations Research*, 34:1885–1898, 2007.
12. A. Ben-David, L. Sterling, and Y.-H. Pao. Learning and classification of monotonic ordinal concepts. *Computational Intelligence*, 5:45–49, 1989.

13. M. Beuthe and G. Scannella. Comparative analysis of UTA multicriteria methods. *European Journal of Operational Research*, 130:246–262, 2001.
14. J-P. Brans and B. Mareschal. PROMETHEE methods. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis-State of the Art Surveys*, pages 163–195. Springer, Boston, 2005.
15. L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.
16. L. Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.
17. L. Breiman. Statistical modeling: The two cultures. *Statistical Science*, 16:199–231, 2001.
18. L. Breiman, J.H. Friedman, R.A. Olsen, and C.J. Stone. *Classification and Regression Trees*. Wadsworth, Pacific Grove, CA, 1984.
19. V. Bugera, H. Konno, and S. Uryasev. Credit cards scoring with quadratic utility function. *Journal of Multi-Criteria Decision Analysis*, 11:197–211, 2002.
20. C. Campbell. Kernel methods: A survey of current techniques. *Neurocomputing*, 48:63–84, 2002.
21. J. Chen and S. Lin. An interactive neural network-based approach for solving multiple criteria decision-making problems. *Decision Support Systems*, 36:137–146, 2003.
22. M-C. Chen. Ranking discovered rules from data mining with multiple criteria by data envelopment analysis. *Expert Systems with Applications*, 33:1110–1116, 2007.
23. Y. Chen, K.W. Hipel, and M.D. Kilgour. Multiple-criteria sorting using case-based distance models with an application in water resources management. *IEEE Transactions on Systems, Man, and Cybernetics-Part A*, 37(5):680–691, 2007.
24. Y. Chen, M.D. Kilgour, and K.W. Hipel. A case-based distance method for screening in multiple-criteria decision aid. *Omega*, 36(3):373–383, 2008.
25. D.H. Choi, B.S. Ahn, and S.H. Kim. Prioritization of association rules in data mining: Multiple criteria decision approach. *Expert Systems with Applications*, 29(4):876878, 2005.
26. P. Das. In search of best alternatives: a topsis driven mcdm procedure for neural network modeling. *Neural Computing and Applications*, 2009. In press.
27. J. Dembczynski, W. Kotlowski, and R. Slowinski. Ensemble of decision rules for ordinal classification with monotonicity constraints. *Lecture Notes in Computer Science*, 5009:260–267, 2008.
28. K. Dembczynski, W. Kotlowski, and R. Slowinski. Additive preference model with piecewise linear components resulting from dominance-based rough set approximations. *Lecture Notes in Computer Science*, 4029:499–508, 2006.
29. L. Dias, V. Mousseau, J. Figueira, and J. Clímaco. An aggregation/disaggregation approach to obtain robust conclusions with ELECTRE TRI. *European Journal of Operational Research*, 138(2):332–348, 2002.
30. L.C. Dias and V. Mousseau. Inferring Electre's veto-related parameters from outranking examples. *European Journal of Operational Research*, 170(1):172–191, 2006.
31. T.G. Dietterich. Ensemble methods in machine learning. *Lecture Notes in Computer Science*, 1857:1–15, 2000.
32. J. Dombi and A. Zsiros. Learning multicriteria classification models from examples: Decision rules in continuous space. *European Journal of Operational Research*, 160:663–675, 2005.
33. M. Doumpos, Y. Marinakis, M. Marinaki, and C. Zopounidis. An evolutionary approach to construction of outranking models for multicriteria classification: The case of the ELECTRE TRI method. *European Journal of Operational Research*, 199(2):496–505, 2009.
34. M. Doumpos and C. Zopounidis. *Multicriteria Decision Aid Classification Methods*. Springer, New York, 2002.
35. M. Doumpos and C. Zopounidis. On the development of an outranking relation for ordinal classification problems: An experimental investigation of a new methodology. *Optimization Methods and Software*, 17(2):293–317, 2002.
36. M. Doumpos and C. Zopounidis. A multicriteria classification approach based on pairwise comparisons. *European Journal of Operational Research*, 158:378–389, 2004.

37. M. Doumpos and C. Zopounidis. Regularized estimation for preference disaggregation in multiple criteria decision making. *Compututational Optimization and Applications*, 38:61–80, 2007.
38. R.M. Everson and J.E. Fieldsend. Multi-class roc analysis from a multi-objective optimisation perspective. *Pattern Recognition Letters*, 27:918–927, 2006.
39. T. Evgeniou, C. Boussios, and G. Zacharia. Generalized robust conjoint estimation. *Marketing Science*, 24(3):415–429, 2005.
40. E. Fernandez, J. Navarroa, and S. Bernal. Multicriteria sorting using a valued indifference relation under a preference disaggregation paradigm. *European Journal of Operational Research*, 198(2):602–609, 2009.
41. J.R. Figueira, S. Greco, and R. Slowinski. Building a set of additive value functions representing a reference preorder and intensities of preference: GRIP method. *European Journal of Operational Research*, 195:460–486, 2009.
42. P. Fortemps, S. Greco, and R. Slowinski. Multicriteria decision support using rules that represent rough-graded preference relations. *European Journal of Operational Research*, 188:206–223, 2008.
43. Y. Freund. Boosting a weak learning algorithm by majority. *Information and Computation*, 121(2):256–285, 1995.
44. Y. Freund, R. Iyer, R.E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969, 2003.
45. J. Gama and P. Brazdil. Cascade generalization. *Machine Learning*, 41:315–343, 2000.
46. J. García-Nieto, E. Alba, L. Jourdan, and E. Talbi. Sensitivity and specificity based multiobjective approach for feature selection: Application to cancer diagnosis. *Information Processing Letters*, 109(16):887–896, 2009.
47. N. García-Pedrajas and D. Ortiz-Boyer. A cooperative constructive method for neural networks for pattern recognition. *Pattern Recognition*, 40:8098, 2007.
48. M.R. Gholamian, S.M.T. Fatemi Ghomi, and M. Ghazanfari. A hybrid intelligent system for multiobjective decision making problems. *Computers and Industrial Engineering*, 51:26–43, 2006.
49. Y. Goletsis, C. Papaloukas, D.I. Fotiadis, A. Likas, and L.K. Michalis. Automated ischemic beat classification using genetic algorithms and multicriteria decision analysis. *IEEE Transactions on Biomedical Engineering*, 51(10):1717–1725, 2004.
50. J. Gonzalez-Pachon and C. Romero. A method for dealing with inconsistencies in pairwise comparisons. *European Journal of Operational Research*, 158(2):351–361, 2004.
51. M. Grabisch, I. Kojadinovic, and P. Meyer. A review of methods for capacity identification in Choquet integral based multi-attribute utility theory: Applications of the Kappalab R package. *European Journal of Operational Research*, 186:766–785, 2008.
52. S. Greco, B. Matarazzo, and R. Slowinski. Rough approximation of a preference relation by dominance relations. *European Journal of Operational Research*, 117:63–83, 1999.
53. S. Greco, B. Matarazzo, and R. Slowinski. Rough sets theory for multicriteria decision analysis. *European Journal of Operational Research*, 129:1–47, 2001.
54. S. Greco, B. Matarazzo, and R. Slowinski. Rough sets methodology for sorting problems in presence of multiple attributes and criteria. *European Journal of Operational Research*, 138:247–259, 2002.
55. S. Greco, B. Matarazzo, and R. Slowinski. Axiomatic characterization of a general utility function and its particular cases in terms of conjoint measurement and rough-set decision rules. *European Journal of Operational Research*, 158(2):271–292, 2004.
56. S. Greco, V. Mousseau, and R. Slowinski. Ordinal regression revisited: Multiple criteria ranking using a set of additive value functions. *European Journal of Operational Research*, 191(2):415–435, 2008.
57. S. Greco, R. Slowinski, and Y. Yao. Bayesian decision theory for dominance-based rough set approach. In J. Yao, P. Lingras, W-Z. Wu, M. Szczuka, N.J. Cercone, and S. Ślęzak, editors, *Rough Sets and Knowledge Technology*, pages 134–141. Springer, Berlin, 2007.

58. M. Guijarro and G. Pajares. On combining classifiers through a fuzzy multicriteria decision making approach: Applied to natural textured images. *Expert Systems with Applications*, 36:7262–7269, 2009.

59. A. Guillén, H. Pomares, J. González, I. Rojas, O. Valenzuela, and B. Prieto. Parallel multiobjective memetic rbfnns design and feature selection for function approximation problems. *Neurocomputing*, 72(16-18):3541–3555, 2009.

60. P.L. Hammer, A. Kogan, B. Simeone, and S. Szedmák. Pareto-optimal patterns in logical analysis of data. *Discrete Applied Mathematics*, 144:79–102, 2004.

61. D. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining*. MIT Press, Cambridge, 2001.

62. T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, 2001.

63. S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, Singapore, 2nd edition, 1999.

64. R. Herbrich, T. Graepel, and K. Obermayer. Large margin rank boundaries for ordinal regression. In A.J. Smola, P.L. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 115–132. MIT Press, Cambridge, MA, 2000.

65. T. Hofmann, B. Schölkopf, and A.J. Smola. Kernel methods in machine learning. *The Annals of Statistics*, 36(3):1171–1220, 2008.

66. A. Howard and T. Jebara. Learning monotonic transformations for classification. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 681–688. MIT Press, Cambridge, MA, 2008.

67. Y-C. Hu. Bankruptcy prediction using ELECTRE-based single-layer perceptron. *Neurocomputing*, 72:3150–3157, 2009.

68. B. Huang, B. Buckley, and T.-M. Kechadi. Multi-objective feature selection by using NSGA-II for customer churn prediction in telecommunications. *Expert Systems with Applications*, 2007. In press.

69. E. Hüllermeier and K. Brinker. Learning valued preference structures for solving classification problems. *Fuzzy Sets and Systems*, 159(18):2337–2352, 2008.

70. E. Hüllermeier and S. Vanderlooy. Combining predictions in pairwise classification: An optimal adaptive voting strategy and its relation to weighted voting. *Pattern Recognition*, 43(1):128–142, 2010.

71. E. Jacquet-Lagrèze and Y. Siskos. Assessing a set of additive utility functions for multicriteria decision making: The UTA method. *European Journal of Operational Research*, 10:151–164, 1982.

72. E. Jacquet-Lagrèze and Y. Siskos. Preference disaggregation: 20 years of MCDA experience. *European Journal of Operational Research*, 130:233–245, 2001.

73. T. Jiao, J. Peng, and T. Terlaky. A confidence voting process for ranking problems based on support vector machines. *Annals of Operations Research*, 166:23–38, 2009.

74. J. Jin. *Multi-Objective Machine Learning*. Springer, Berlin Heidelberg, 2006.

75. T. Joachims. Optimizing search engines using clickthrough data. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 133–142, 2002.

76. G.V. Kass. An exploratory technique for investigating large quantities of categorical data. *Journal of Applied Statistics*, 29:119–127, 1980.

77. R.L. Keeney and H. Raiffa. *Decisions with Multiple Objectives: Preferences and Value Trade-offs*. Cambridge University Press, Cambridge, 1993.

78. I. Kojadinovic. Estimation of the weights of interacting criteria from the set of profiles by means of information-theoretic functionals. *European Journal of Operational Research*, 155:741–751, 2004.

79. I. Kojadinovic. Minimum variance capacity identification. *European Journal of Operational Research*, 177:498–514, 2007.

80. M. Köksalan and S.B. Özpeynirci. An interactive sorting method for additive utility functions. *Computers & Operations Research*, 36(9):2565–2572, 2009.

81. M. Köksalan and C. Ulu. An interactive approach for placing alternatives in preference classes. *European Journal of Operational Research*, 144:429–439, 2003.

82. W. Kotlowski and R. Slowinski. Rule learning with monotonicity constraints. In *ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning*, pages 537–544, New York, NY, USA, 2009. ACM.
83. K. Kottathra and Y. Attikiouzel. A novel multicriteria optimization algorithm for the structure determination of multilayer feedforward neural networks. *Journal of Network and Computer Applications*, 19:135–147, 1996.
84. G. Kou, X. Liu, Y. Peng, Y. Shi, M. Wise, and W. Xu. Multiple criteria linear programming approach to data mining: Models, algorithm designs and software development. *Optimization Methods and Software*, 18(4):453–473, 2003.
85. H. Li and J. Sun. Hybridizing principles of the ELECTRE method with case-based reasoning for data mining: ELECTRE-CBR-I and ELECTRE-CBR-II. *European Journal of Operational Research*, 197(1):214–224, 2009.
86. H. Li and J. Sun. Business failure prediction using hybrid^2 case-based reasoning (H^{2}CBR). *Computers and Operations Research*, 37(1):137–151, 2010.
87. Y. Li, K. Ida, M. Gen, and R. Kobuchi. Neural network approach for multicriteria solid transportation problem. *Computers and Industrial Engineering*, 33(3-4):465–468, 1997.
88. J. Ma, Z.P. Fan, Y.P. Jiang, J.Y. Mao, and L. Ma. A method for repairing the inconsistency of fuzzy preference relations. *Fuzzy Sets and Systems*, 157(1):20–33, 2006.
89. B. Malakooti and V. Raman. Clustering and selection of multiple criteria alternatives using unsupervised and supervised neural networks. *Journal of Intelligent Manufacturing*, 11:435–451, 2000.
90. B. Malakooti and Y. Zhou. A recursive ann for solving adaptive multiple criteria problems. *Pure Mathematics and Applications Series C*, 2(2-4):165–176, 1991.
91. B. Malakooti and Y.Q. Zhou. Feedforward artificial neural networks for solving discrete multiple criteria decision making problems. *Management Science*, 40(11):1542–1561, 1994.
92. O.L. Mangasarian and E.W. Wild. Nonlinear knowledge-based classification. *IEEE Transactions on Neural Networks*, 19(10):1826–1832, 2008.
93. J.-L. Marichal and M. Roubens. Determination of weights of interacting criteria from a reference set. *European Journal of Operational Research*, 124:641–650, 2000.
94. J-M. Martel and B. Matarazzo. Other outranking approaches. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis-State of the Art Surveys*, pages 197–262. Springer, Boston, 2005.
95. P.R. McMullen. A kohonen self-organizing map approach to addressing a multiobjective mixed-model JIT sequencing problem. *International Journal of Production Economics*, 72:59–71, 2001.
96. V. Mousseau, L.C. Dias, and J. Figueira. Dealing with inconsistent judgments in multiple criteria sorting models. *4OR*, 4(3):145–158, 2006.
97. V. Mousseau, L.C. Dias, J. Figueira, C. Gomes, and J.N. Clímaco. Resolving inconsistencies among constraints on the parameters of an MCDA model. *European Journal of Operational Research*, 147(1):72–93, 2003.
98. V. Mousseau, J. Figueira, and J.-Ph. Naux. Using assignment examples to infer weights for ELECTRE TRI method: Some experimental results. *European Journal of Operational Research*, 130:263–275, 2001.
99. V. Mousseau and R. Slowinski. Inferring an ELECTRE-TRI model from assignment examples. *Journal of Global Optimization*, 12(2):157–174, 1998.
100. H. Nakayama, Y.B. Yun, T. Asada, and M. Yoon. MOP/GP models for machine learning. *European Journal of Operational Research*, 166:756–768, 2005.
101. A. Ngo The and V. Mousseau. Using assignment examples to infer category limits for the ELECTRE TRI method. *Journal of Multi-Criteria Decision Analysis*, 11:2943, 2002.
102. Y. Ni, C. Huang, and S. Kokot. Application of multivariate calibration and artificial neural networks to simultaneous kinetic-spectrophotometric determination of carbamate pesticides. *Chemometrics and Intelligent Laboratory Systems*, 71:177193, 2004.
103. L.S. Oliveira, M. Morita, and R. Sabourin. Feature selection for ensembles applied to handwriting recognition. *International Journal on Document Analysis and Recognition*, 8(4):262–279, 2006.

104. K-M. Osei-Bryson. Evaluation of decision trees: A multi-criteria approach. *Computers and Operations Research*, 31:1933–1945, 2004.
105. K-M. Osei-Bryson. Post-pruning in decision tree induction using multiple performance measures. *Computers and Operations Research*, 34:3331–3345, 2007.
106. M. Öztürk, A. Tsoukiàs, and Ph. Vincke. Preference modelling. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis-State of the Art Surveys*, pages 27–71. Springer, Boston, 2005.
107. C.-P. Park and I. Han. A case-based reasoning with the feature weights derived by analytic hierarchy process for bankruptcy prediction. *Expert Systems with Applications*, 23(3):255–264, 2002.
108. Z. Pawlak. Rough sets. *International Journal of Information and Computer Sciences*, 11:341–356, 1982.
109. Z. Pawlak and R. Slowinski. Rough set approach to multi-attribute decision analysis. *European Journal of Operational Research*, 72:443–459, 1994.
110. R. Potharst and A.J. Feelders. Classification trees for problems with monotonicity constraints. *ACM SIGKDD Explorations Newsletter*, 4(1):1–10, 2002.
111. L. Qu and Y. Chen. A hybrid MCDM method for route selection of multimodal transportation network. *Lecture Notes in Computer Science*, 5263:374–383, 2008.
112. J.R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.
113. J.R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, Los Altos, California, 1993.
114. K.S. Raju, D.N. Kumar, and L. Duckstein. Artificial neural networks and multicriterion analysis for sustainable irrigation planning. *Computers and Operations Research*, 33:1138–1153, 2006.
115. B. Roy. The outranking approach and the foundations of ELECTRE methods. *Theory and Decision*, 31:49–73, 1991.
116. B. Roy. *Multicriteria Methodology for Decision Aiding*. Springer, New York, 1996.
117. T.L. Saaty. *Fundamentals of the Analytic Hierarchy Process*. RWS Publications, Pittsburgh, PA, 2006.
118. R.E. Schapire, Y. Freund, P. Bartlett, and W.S. Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26(5):1651–1686, 1998.
119. B. Schölkopf and A. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, Cambridge, 2002.
120. B. Schölkopf and A.J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, Cambridge, Massachusetts, 2002.
121. J.-B. Sheu. A hybrid neuro-fuzzy analytical approach to mode choice of global logistics management. *European Journal of Operational Research*, 189(3):971–986, 2008.
122. Y. Shimizu, Y. Tanaka, and A. Kawada. Multi-objective optimization system, MOON^2 on the internet. *Computers and Chemical Engineering*, 28:821–828, 2004.
123. J. Sill. Monotonic networks. In M.I. Jordan, M.J. Kearns, and S.A. Solla, editors, *Advances in Neural Information Processing Systems 10*, pages 661–667. MIT Press, Cambridge, MA, 1997.
124. J. Siskos and D.K. Despotis. A DSS oriented method for multiobjective linear programming problems. *Decision Support Systems*, 5:47–55, 1989.
125. J. Siskos, E. Grigoroudis, and N.F. Matsatsinis. UTA methods. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis-State of the Art Surveys*, pages 297–343. Springer, Boston, 2005.
126. R. Slowinski and J. Stefanowski. Rough classification with valued closeness relation. In E. Diday, Y. Lechevallier, M. Schader, P. Bertrand, and B. Burtschy, editors, *New Approaches in Classification and Data Analysis*, pages 482–489. Springer, Berlin, 1994.
127. A. Stam, M. Sun, and M. haines. Artificial neural network representations for hierarchical preference structures. *Computers and Operations Research*, 23(12):1191–1201, 1996.
128. I. Steinwart and A. Christmann. *Support Vector Machines*. Springer, New York, 2008.
129. R.E. Steuer and E.U. Choo. An interactive weighted Tchebycheff procedure for multiple objective programming. *Mathematical Programming*, 26(1):326–344, 1983.

130. M. Sun, A. Stam, and R.E. Steuer. Solving multiple objective programming problems using feed-forward artificial neural networks: The interactive FFANN procedure. *Management Science*, 42(6):835–849, 1996.
131. M. Sun, A. Stam, and R.E. Steuer. Interactive multiple objective programming using Tchebycheff programs and artificial neural networks. *Computers and Operations Research*, 27(7-8):601–620, 2000.
132. R.A. Teixeira, A.P. Braga, R.H.C. Takahashi, and R.R. Saldanha. Improving generalization of mlps with multi-objective optimization. *Neurocomputing*, 35(14):189194, 2000.
133. G.G. Towell and J.W. Shavlik. Knowledge-based artificial neural networks. *Artificial Intelligence*, 70(1-2):119–165, 1994.
134. R. van de Kamp, A. Feelders, and N. Barile. Isotonic classification trees. *Lecture Notes in Computer Science*, 5772:405–416, 2009.
135. V.N. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999.
136. V.N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, 2nd edition, 2000.
137. M. Velikova, H. Daniels, and A. Feelders. Mixtures of monotone networks for prediction. *International Journal of Computational Intelligence*, 3(3):205–214, 2006.
138. W. Waegeman, B. De Baets, and B. Boullart. Kernel-based learning methods for preference aggregation. *4OR*, 7:169–189, 2009.
139. J. Wang. A neural network approach to modeling fuzzy preference relations for multiple criteria decision making. *Computers and Operations Research*, 21(9):991–1000, 1994.
140. J. Wang. A neural network approach to multiple criteria decision making based on fuzzy preference information. *Information Sciences*, 78:293–302, 1994.
141. J. Wang and B. Malakooti. A feedforward neural network for multiple criteria decision making. *Computers and Operations Research*, 19(2):151–167, 1992.
142. J. Wang, J.-Q. Yang, and H. Lee. Multicriteria order acceptance decision support in overdemanded job shops: A neural network approach. *Mathematical and Computer Modelling*, 19(5):1–19, 1994.
143. S. Wang and N.P. Archer. A neural network technique in modeling multiple criteria multiple person decision making. *Computers and Operations Research*, 21(2):127–142, 1994.
144. Y. Wang. Multicriteria neural network approach to turbulent image reconstruction. *Optics Communications*, 143:279–286, 1997.
145. D.H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.
146. D.H. Wolpert. The existence of a priori distinctions between learning algorithms. *Neural Computation*, 8(7):1391–1420, 1996.
147. D.H. Wolpert. The lack of a priori distinctions between learning algorithms. *Neural Computation*, 8(7):1341–1390, 1996.
148. D.H. Wolpert and W.G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.
149. Y. Zhang and P.I. Rockett. Domain-independent feature extraction for multi-classification using multi-objective genetic programming. *Pattern Analysis and Applications*, 2009. In press.
150. C. Zopounidis and M. Doumpos. Building additive utilities for multi-group hierarchical discrimination: The MHDIS method. *Optimization Methods and Software*, 14(3):219–240, 2000.