Anais do IX Congresso Brasileiro de Redes Neurais / Inteligência Computacional (IX CBRN)
Ouro Preto 25-28 de Outubro de 2009

MODELLING THE DECISION-MAKER UTILITY FUNCTION THROUGH ARTIFICIAL NEURAL NETWORKS

Luciana R. Pedro*, Ricardo H. C. Takahashi†

*Department of Electric Engineering
Universidade Federal de Minas Gerais
Belo Horizonte, Minas Gerais, Brasil

†Department of Mathematics
Universidade Federal de Minas Gerais
Belo Horizonte, Minas Gerais, Brasil

Emails: lrocha@cpdee.ufmg.br, taka@mat.ufmg.br

Abstract— In this article, we present a method to approximate the decision-maker utility function in a decision model based on the multiattribute utility theory (MAUT). This approximation is built using through the construction of a partial sorting for the feasible alternatives named ranking and an artificial neural network, which captures informations of the original utility function through this ranking. We present one utility function model and the results obtained using the proposed method.

Keywords— Multicriteria decision analysis, artificial neural networks.

Resumo— Neste artigo, apresentamos um método para aproximar a função de utilidade de um decisor em um modelo de decisão baseado na teoria da utilidade multiatributo (MAUT). Esta aproximação é feita através da construção de uma ordenação entre as alternativas disponíveis e uma rede neural artificial, que obtém informações da função original através desta ordenação. Apresentamos dois modelos de função de utilidade e os resultados obtidos com o método proposto.

Palavras-chave— Análise de decisão multicritério, redes neurais.

1 Introduction

The multicriteria analysis consists of a set of methods and techniques to assist or to support people and organizations in order to make decisions, considering multiple criteria.

A multicriteria decision making problem involves the following basic elements:

- *set A of alternatives (possible actions or choices).* Each element $a \in A$ corresponds to a feasible alternative. This set can be discrete or continuous and it is denoted as the domain of the decision making problem.
- *set B of consequences or attributes.* Each alternative in set A has attributes, which reflect the consequences of its execution.
- *one decision-maker.* In order to determine which one is the best alternative, a decision-maker provides ordinal information about the preference relating to the alternatives.

In literature, there are many real applications of decision problems in diverse areas, such as:

- transport area: for planning the revitalization of subway stations (Roy et al., 1986);
- energy area: for locating thermal power plants (Barda et al., 1990).
- marketing: for estimating of the customers satisfaction (Siskos et al., 1998);
- medicine: in medical diagnosis, for patients illnesses categories based on their symptoms (Belacel, 2000);

This article considers the construction of a function which models the decision-maker preferences based on the multiattribute utility theory (Section 2). This function is built from a ranking of feasible alternatives, which is established considering the answers provided by the decision-maker (Section 3). This ranking is used for constructing an artificial neural network which models the decision-maker (Section 4). The results (Section 5) and the conclusions (Section 6) are presented.

2 Decision making methods

2.1 Introduction

The decision based on mathematical models for human preference considers that it is always possible to sort any set of alternatives through the decision-maker preferences. This sorting can be used to identify the best alternative or to classify its elements in categories.

Currently, there are two main research areas in decision making: the decision based on the multiattribute utility theory and the decision based on outranking relations.

2.2 Multiattribute utility theory

According to the multiattribute utility theory (MAUT), in decision problems it is possible to construct a function U, denoted utility function, which represents the decision-maker preferences. Using U, a scalar value is attributed to alternatives in A, which can be sorted by the simple comparison of the values.

The usage of the MAUT-based methods is appropriate for cases in which the decision-maker can be considered perfectly rational, knowing all necessary information about the problem. Although these methods usually require low computational cost, they demand a lot of information provided by the decision-maker.

Among MAUT-based methods, we can cite: Programming for Goals (Lee, 1972), Average Point (Chankong and Haimes, 1983), AHP (Saaty, 1986), Smarts/Smarter (Edwards and Barron, 1994), Macbeth (Bana e Costa and Vansnick, 1997) and Interval Smart-Swing (Mustajoki et al., 2001).

2.3 Outranking relations

The methods based on outranking relations have been developed for dealing with situations which can not be modelled by MAUT-based methods. In general, these methods are characterized by two training periods: construction of the outranking relation and exploitation of the results obtained on the previous stage (Bouyssou, 2001).

The usage of methods based on outranking relations is relevant when the decision-maker does not have total knowledge of the preferences. In general, these methods involve more complex algorithms, but they demand less information of the decision-maker preference.

Among the methods based on outranking relations, we can cite: Electre I (Roy, 1968), Electre III (Roy, 1978), Promethee I (Brans et al., 1986), Promethee II (Brans et al., 1986) and Multiplicative Promethee (Parreiras and Vasconcelos, 2007).

3 Decision problem solution

3.1 Introduction

In this paper, we consider that the decision-maker knows the preferences at the beginning of the decision process and these preferences are defined regarding all the alternatives. The decision-maker answers are not quantitative, that is, given two alternatives a_i and a_j, with $i \neq j$, the alternative a_i is preferred to the alternative a_j or vice versa, but it is not possible to determine how preferable this solution is.

According to MAUT, we have an utility function U which represents the decision-maker preferences assigning a scalar value to each alternative. In this paper, we consider that this utility function is modelled by a continuous function. This assumption is consistent with the problem, since given two alternatives which slightly differ between them, it is reasonable to consider that the decision-maker preference regarding these alternatives presents small changes in a continuous form.

3.2 Exact soluction

The simplest way to get the best alternative in a decision making problem is to ask queries to the decision-maker about each pair of alternatives. When making all the queries, we find the preferred alternative for the decision-maker. This method is not efficient since the number of queries for the decision-maker becomes huge.

3.3 Ranking

One of the ways of finding the best solution in a decision problem, without asking all the possible queries to the decision-maker, is to use a partial sorting process, denoted *ranking*. This process is performed through the following steps:

- an alternative is chosen and compared with the remaining ones; this alternative is called *pivot*;
- the alternatives which are prefered in relation to the pivot pass to the next step;
- the process is repeated until we have only one alternative; this alternative is the choice of the decision-maker.

4 Utility function approximation

4.1 Introduction

In this section, the ranking process is used to find a partial sorting for the alternatives and construct an artificial neural network which should ideally model the original decision-maker preferences.

The proposed method consists in three main steps:

Grid of fictitious alternatives	Choose a domain for the utility function approximation
Ranking	Build the ranking for the alternatives, assigning a scalar value to each alternative and finding a partial sorting for the alternatives
Artificial neural network	Construct an artificial neural network which interpolates the results and represents the decision-maker preferences

4.2 Grid of fictitious alternatives

Given a decision-making problem, we need to determine a domain for the utility function approximation. We define the domain of the approximated function as the minimal box which contain the feasible alternatives. In this domain, we construct a grid of fictitious alternatives and ask queries to the decision-maker regarding to these alternatives.

Figure 1 presents a set of feasible alternatives and a grid of fictitious alternatives construct from the feasible alternatives.

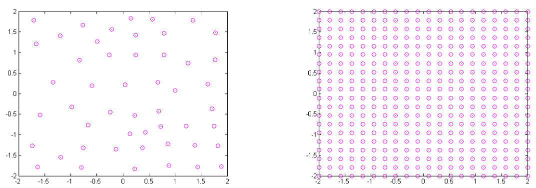


Figure 1: Obtained domain through feasible alternatives.

This grid is constructed to find an uniform representation of the utility function in the desired domain. The number of fictitious alternatives to construct the grid is related to the quality of solution: a fine grid supplies a better approximation, but, in this case, many queries are asked to the decision-maker.

4.3 Ranking

In each step of the process presented in the Section 3.3, we generate a partial sorting for the considered alternatives, since the alternatives which evolve are better than or equivalent to the previous ones. Through the ranking we have a way for quantifying the decision-maker preferences.

Considering each process step as a ranking level, we have a merit function which evaluates the alternatives based on their evolution. The process is developed as follows:

- in the beggining, all the alternatives receive level zero;
- in each step algorithm, the level of the alternatives which pass to the next step is increased by one.

Since the ranking values are discrete, we do not expect to achieve an accurate representation for the decision-maker utility function. Instead, we want that this approximated function presents the same level sets of the original utility function and preserves the partial sorting of the alternatives.

4.4 Artificial neural network

An artificial neural network (ANN) is an information processing paradigm which is inspired by the way which biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working together to solve specific problems.

ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true for ANNs as well.

The main objective of the learning in ANN's is the attainment of models with good generalization capacity, associated to the network capacity to learn by a reduced set of examples and to supply coherent answers to unknown data.

In this paper, we work with only one ANN architecture: multilayer perceptron (MLP).

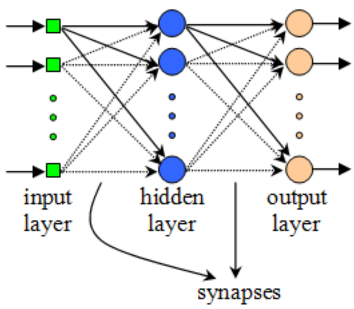


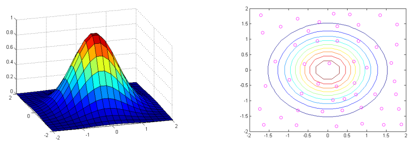
Figure 2: Multilayer perceptron (MLP).

Using the grid of fictitious alternatives and the decision-maker preferences regarding these alternatives, we construct the ranking, as described in Section 4.3. For constructing the ANN which will approximate the decision-maker preferences, we use the grid of fictitious alternatives as input and the ranking level as output. Since the ranking keeps the partial sorting of the alternatives, when it is used to construct the ANN input, we find a function with similar level sets to the original utility function and that possess the necessary information to find the decision-maker preferences.

Given new alternatives in the same domain, we can simply use the function to choose the preferred alternative, without consulting the decision-maker again. The preferred alternative is the one which has a greater value in the approximated function.

5 Results

For representing an analytical model, which will be used for the purpose of simulating the decision-maker preferences, we initially choose a bidimensional unimodal Gaussian (see Figure 3). This model was chosen because is intuitive that the preferences of the decision-maker should be an unimodal function, the preferences about each alternative must not be negative numbers and should decay for zero for bad alternatives.

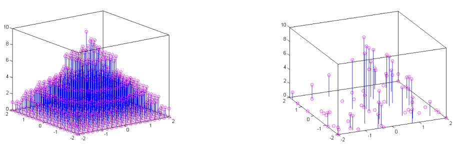


Surface Level sets

Figure 3: Expected model of the decision-maker preferences.

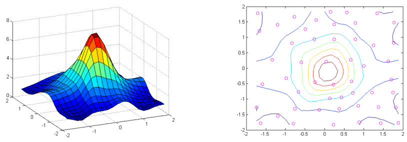
Considering a decison making problem with 50 feasible alternatives in the $[-2, 2]$ interval, we use these alternatives to establish the domain of the approximated function and construct one grid with 400 fictitious alternatives. With this grid, we build the ranking and construct an utility function approximation through a MLP 2-18-1.

Figure 4 shows the ranking and Figure 5 shows the ANN function approximation.



All the alternatives 20% of the alternatives

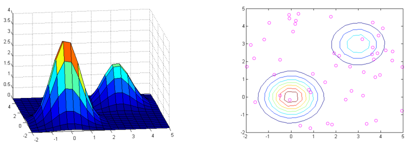
Figure 4: Ranking found to the alternatives of the grid.



Surface Level sets

Figure 5: Artificial neural network which models the decision-maker preferences.

Now, we present another analytical model: a bidimensional bimodal Gaussian (see Figure 6). This model is appropriate when the decision-maker has two tendencies well defined for the preferences. As the domain is represented uniformly by the grid of fictitious alternatives, these tendencies are modelled by the ANN in an accurate way.

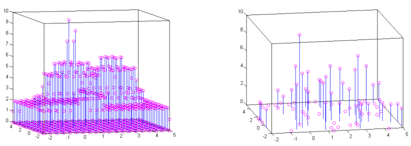


Surface Level sets

Figure 6: Expected model of the decision-maker preferences.

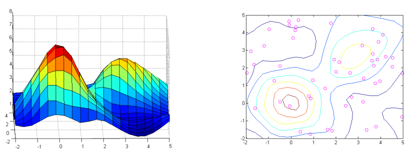
Considering a decison making problem with 50 feasible alternatives in the $[-2, 5]$ interval, we use these alternatives to establish the domain of the approximated function and construct one grid with 400 fictitious alternatives. With this grid, we build the ranking and construct an utility function approximation through a MLP 2-24-1.

Figure 7 shows the ranking and Figure 8 shows the ANN function approximation.



Surface Level sets

Figure 7: Ranking found to the alternatives of the grid.



Surface Level sets

Figure 8: Artificial neural network which models the decision-maker preferences.

In 50 algorithm executions, the proposed methodology found the solution that would be chosen by the decision-maker in a scenario of queries performing an exhaustive comparison among the alternatives (See Section 3.2) for both models. These results show that the decision-maker preferences could be represented for the approximated function found by the ANN.

6 Conclusions

In this paper, we presented an utility function approximation in a decision problem using a ranking process and an ANN. This approximation reflects the decision-maker preferences in a specified domain, in which the ANN is trained. This method is suitable when the decision-maker has the preferences in accordance with the usual assumptions of multiattribute utility theory.

With this approximation, problems in which the same decision-maker is consulted many times in similar processes, as in approximation of a Pareto set, can be solved in the following way:

- given a decision making problem, we define a domain for the utility function approximation and construct a grid of fictitious alternatives;
- with the grid, we build a ranking and get a parcial sorting for the alternatives;
- with the grid and the ranking, we construct the artificial neural network input and output;
- using the artificial neural network, we get an utility function approximation in the chosen domain.

With this approximation, no more queries to the decision-maker are necessary, since the preferences are modelled through the approximated function.

References

Bana e Costa, C. and Vansnick, J. (1997). Applications of the macbeth approach in the framework of an additive aggregation model, *Journal of Multi-Criteria Decision Analysis* **6**: 107–114.

Barda, O. H., Dupuis, J. and Lencioni, P. (1990). Multicriteria location of thermal power plants, *European Journal of Operational Research* **45**(2-3): 332–346.

Belacel, N. (2000). Multicriteria assignment method proaftn: Methodology and medical application, *European Journal of Operational Research* **125**(1): 175–183.

Bouyssou, D. (2001). Outranking methods, *in* C. Floudas and P. Pardalos (eds), *Encyclopedia of optimization*, Kluwer.

Brans, J., Vincke, P. and Mareschal, B. (1986). How to select and how to rank projects: the promethee method, *European Journal of Operational Research* **24**: 228–238.

Chankong, V. and Haimes, Y. (1983). *Multiobjective Decision Making Theory and Methodology*, Elsevier Science, New York.

Edwards, W. and Barron, F. H. (1994). Smarts and smarter: Improved simple methods for multiattribute utility measurement, *Organizational Behavior and Human Decision Processes* **60**(3): 306–325.

Lee, S. M. (1972). *Goal programming for decision analysis*, 1st ed. edn, Auerbach Publishers, Philadelphia,.

Mustajoki, J., Hamalainen, R. P. and Salo, A. (2001). Decision support by interval smart/swing - methods to incorporate uncertainty into multiattribute analysis, *in the SMART and SWING methods, Decision Sciences 36*, pp. 317–339.

Parreiras, R. and Vasconcelos, J. (2007). A multiplicative version of promethee ii applied to multiobjective optimization problems, *European Journal of Operational Research* **183**(2): 729–740.

Roy, B. (1968). Classement et choix en présence de points de vue multiples: La méthode ELECTRE, *Revue Francaise d' Informatique et de Recherche Opérationnelle* **8**: 57–75.

Roy, B. (1978). ELECTRE III : Un algorithme de classements fondé sur une représentation floue des préférences en présence de critères multiples, *Cahiers du CERO* **20**(1): 3–24.

Roy, B., Présent, M. and Silhol, D. (1986). A programming method for determining which Paris metro stations should be renovated, *European Journal of Operational Research* **24**: 318–334.

Saaty, T. (1986). Axiomatic foundation of the analytic hierarchy process, *Management Science* **32**(7): 841–855.

Siskos, Y., Grigoroudis, E., Zopounidis, C. and Saurais, O. (1998). Measuring customer satisfaction using a collective preference disaggregation model, *Journal of Global Optimization* **12**(2): 175–195.