Categorization of Numerical Values for DEX Hierarchical Models

Martin Žnidaršič[1], Marko Bohanec[1,2] and Ivan Bratko[1,3]
[1] Department of Intelligent Systems, Jožef Stefan Institute
Jamova 39, Ljubljana, Slovenia
[2] Faculty of Administration, University of Ljubljana
Gosarjeva 5, Ljubljana, Slovenia
[3] Faculty of Computer and Information Science, University of Ljubljana
Tržaška 25, Ljubljana, Slovenia

Keywords: decision support, multi-attribute decision models, categorization

Received: July 28, 2003

DEX is a multi-attribute decision modelling methodology. Its specialty is the use of ordinal data and qualitative utility functions. Numerical attributes must be therefore categorized before use in DEX models. We present the problem of numerical data categorization and propose two methods which simplify and partly automate this task. The methods suggest interval bounds according to the desired number of categories and the preference curve of attribute values. We implemented both methods and made some experiments with typical inputs. The most interesting results are presented and analysed.

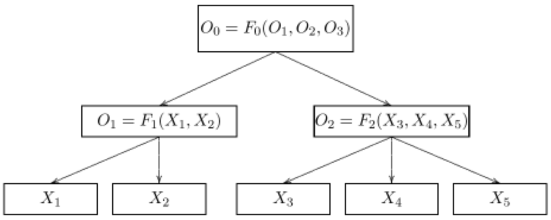
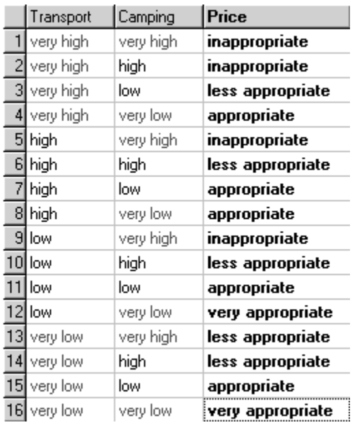


Figure 1: Example of a general MADM.

1 Introduction

Data comes in many forms and usually only some of them are useful for a specific task. In this paper, data represents alternatives for multi-attribute decision models (MADM) [8, 7, 4, 9, 1]. With MADM, we are trying to evaluate, compare and study alternatives. Examples of alternatives are for instance cars, job candidates, office locations, etc. Usually we are trying to select the most appropriate alternative for our goals, the one with the highest utility.

Decision problem-solving with MADM is based on hierarchical decomposition of the problem. Alternatives are hierarchically decomposed into subconcepts (or aggregate attributes) and finally to a finite set of basic attributes. Utilities of aggregate attributes are evaluated with functions, which depend on the corresponding attributes located on the lower levels of the hierarchy. A scheme of a general MADM is shown in Figure 1. The MADM tools usually allow the analysis (alternative ranking, sensitivity analysis) and graphical representation of the decision problem.



	Transport	Camping	Price
1	very high	very high	**inappropriate**
2	very high	high	**inappropriate**
3	very high	low	**less appropriate**
4	very high	very low	**appropriate**
5	high	very high	**inappropriate**
6	high	high	**less appropriate**
7	high	low	**appropriate**
8	high	very low	**appropriate**
9	low	very high	**inappropriate**
10	low	high	**less appropriate**
11	low	low	**appropriate**
12	low	very low	**very appropriate**
13	very low	very high	**less appropriate**
14	very low	high	**less appropriate**
15	very low	low	**appropriate**
16	very low	very low	**very appropriate**

Figure 2: Utility function in tabular form defined in application DEXi [6]. The utility of the price for a camping place is evaluated on the base of the transportation cost and the camping fee.

One of many MADM methodologies is DEX, developed at Jožef Stefan Institute [1, 6]. Unlike traditional methodologies, DEX uses qualitative variables instead of numerical, what makes it suitable for less formalized decision problems. Utility functions in DEX are adjusted to qualitative variables and therefore represented with if-then rules, usually given in tabular form (as in Figure 2). This qualitative approach proved to be very useful in practice, since DEX was used in many real-world decision problems [2, 3].

Numerical attributes often occur in the lower levels of DEX models. In such cases the qualitative approach of DEX is a drawback, as we can not define a rule for every value of a numerical attribute. Sometimes we can describe a certain feature with categoric, instead of numeric values, but when the use of numeric data is obligatory, the only solution is to define intervals of numerical values and to use such intervals as attribute values. The process of splitting attribute values into intervals is called categorization. Selecting appropriate interval bounds is not a trivial task, as the selection is not always obvious and can have a significant impact on the model. We propose two methods to simplify and partly automate this task.

2 Transforming numerical attribute to ordinal one

Numerical attributes are attributes that have numerical values, either continuous or discrete. We will consider only numerical attributes that have continuous or quasi-continuous (discrete with many values) values. Discrete attributes with a very limited range of values can be considered categorical when used in DEX models. Categorical attributes can be nominal (unordered) or ordinal (ordered).

A numerical attribute can be transformed into an ordinal one with categorization. Values of the numerical attribute are divided into intervals, which are then considered as possible values of the new attribute. The division of continuous values into intervals is an unnatural procedure and is difficult even for skilled experts, especially when they can not find sensible bounds. Simple automatic procedures for interval bounds determination, as equal-width for instance, are inappropriate as they usually increase the difference between a decision model and the corresponding real decision environment.

Manual categorization consists of interval bounds selection and ordering of intervals. When domain experts think there is no sensible way of selecting interval bounds, we believe it is best to think about ordering values in continuous space and express our preference with a continuous preference curve. Categorization should then be carried out by a MADM tool. Preference curve is a continuous curve that has a value of preference in the range [0,1] at each attribute value. Lower values mean that attribute values are considered less preferred, higher values on the other hand are at the values of the attribute that are more preferred. The preference 1 denotes an ideal attribute value, whereas the preference 0 denotes the least preferred attribute value.

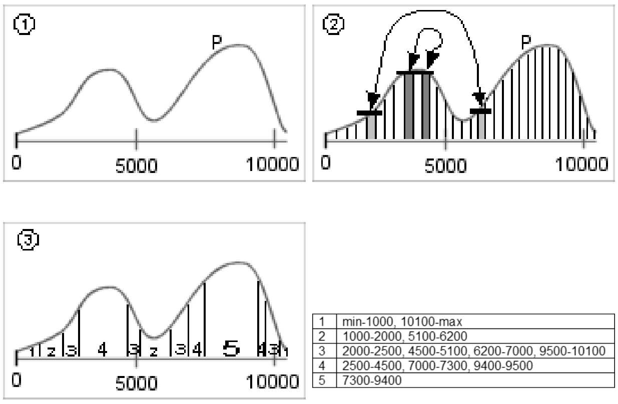
Two automatic procedures that, given the preference curve, simplify the process of categorization, are proposed in sections 2.2 and 2.3.

2.1 Manual Categorization

DEX methodology currently offers no help with the transformation of numerical attributes to ordinal. The user has to perform this transformation manually. Numerical values of the attribute have to be divided into intervals that form a range of values for the new ordinal attribute. If the intervals are left unordered, the new attribute is nominal. However, it is prefered to order the intervals to obtain an ordinal attribute.

2.2 Method 1: Linking intervals

The first proposed categorization method needs only two inputs from the user, the preference curve P for attribute values and the desired size of the set of values for the new ordinal attribute $|D_{\text{ord}}|$. The values of numerical attribute are initially divided into many small intervals (default $10 \times |D_{\text{ord}}|$). Mean preference value $avgP$ is computed for each interval. Then the two intervals with the most similar $avgP$ are linked (see sketch 2 on Figure 3) and their $avgP$ is updated with the mean of previous two values. Procedure of linking similar intervals continues until there are only $|D_{\text{ord}}|$ different $avgP$ values. Remaining $avgP$ values are ranked and given values from 1 to $|D_{\text{ord}}|$ where a higher number means a more preferred value. Linked intervals with common bounds (neighbors) are merged together. With this final step, the transformation from a given numerical attribute to an ordinal one is completed.



1	min-1000, 10100-max
2	1000-2000, 5100-6200
3	2000-2500, 4500-5100, 6200-7000, 9500-10100
4	2500-4500, 7000-7300, 9400-9500
5	7300-9400

Figure 3: Sketches of the *Linking intervals* method.

Example: transport
A simple explanation of the concepts introduced so far is in Figure 3. Suppose we are in the role of a truck driver. From all the possible transport configurations and routes, we have to choose the most profitable one. To analyze our decision problem, we build a MADM. Basic attributes are for instance: length of transport, price of transport, countries on the way, weight of the load, road fees on the way and similar. Let us analyze the attribute *weight of the load* in more detail. When weight reaches 5000 kilograms, we have to use an additional trailer because of physical limitations and traffic regulations. The optional use of trailer reflects in a bimodal preference curve (see sketch 1 on Figure 3). The most preferred weight of the load is the weight of fully loaded truck and trailer, and a minor local maximum is at the point where the truck without trailer is fully

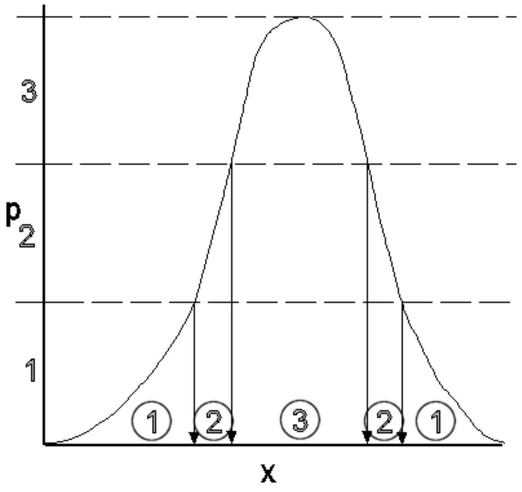


Figure 4: *Following preference* method on Gaussian preference curve.

loaded. Truck with almost empty trailer is not preferred and almost empty truck even less. On the base of such considerations we can construct a preference curve that is used by the method *Linking intervals* to select interval bounds and ordinal values (sketch 3 on Figure 3).

2.3 Method 2: Following preference

Our second proposed method is a simple and computationally less demanding procedure. The preference is divided into $|D_{\text{ord}}|$ equal size intervals that represent the set of ordinal values. Each of these intervals transforms the corresponding attribute values into its ordinal value. The procedure is presented in Figure 4.

It might seem that both proposed methods give identical results, but there are some important differences. *Following preference* method sets the ordinal values strictly regarding the preference, whereas *Linking intervals* method takes the preference into account only to a certain extent. It is more "fuzzy", what makes it less sensitive to sudden changes in preference curve, as well as a bit less accurate regarding preference. Each method has some advantages and some drawbacks, therefore the choice of the right method for a specific preference curve could be important.

3 Implementation and Experiments

Methods *Following preference* and *Linking intervals* were implemented in Delphi environment. Application ACNA for experimenting with the methods is presented in section 3.1. The analysis of methods properties and results is presented in section 3.2.

3.1 ACNA

ACNA (Automatic Categorization of Numeric Attributes) is an application for experimental evaluation and use of the proposed methods for automatic categorization. Graphical user interface facilitates an easy input of parameters and a quick presentation of results in textual and graphical form.

The desired power of the set of values for ordinal attribute and preference curve must be provided. Preference curve must be given as (x, y) coordinate pairs and can be read from file. When entered, preference curve is presented with coordinates and a graph that can be saved as a file.

When all input data is provided, we can start each of the proposed methods. The resulting intervals and their ordinal values appear together with a graph of preference curve divided into calculated intervals. This graph is very useful for a quick overview of results and can also be saved as a file.

3.2 Comparison of Methods

Both proposed methods were tested on four distinctive and very different preference curves. We tested categorization into 2, 3, 4, 5, 6, 7 and 12 values with each curve. We were expecting the method *Following preference* to suggest mostly appropriate divisions. The method *Linking intervals* was expected to be somewhat inaccurate, but we were hoping that it will find a more suitable division in some unusual situations. Of all the experiments we selected only the most interesting ones for the presentation of differences.

Given the preference curve ’increasing’, the results of both methods were very similar. An example of categorization into three values is shown in Figure 5. Differences given this type of preference curve were not expected.

Some differences appear when using preference curve ’normal’, that has a shape of a Gaussian function. Given this type of preference, regardless of the number of desired categoric values, the method to use is *Following preference* (Figure 6). It is a simple problem that calls for simple solution procedure. Any variations from results of *Following preference* are unwanted. Slight inaccuracy of *Linking intervals* method is reflected in suboptimal selection of intervals.

At a bit more stirred preference curve ’bimodal’ reveales an interesting difference in results of the two methods. Results visually do not differ much, but are very different with regard to the number of intervals used in each solution. *Following preference* method proposed for instance 11 intervals for categorization to 5 values. In the same setting, *Linking intervals* method provides a solution with only 7 intervals (Figure 7). A similar trend can be noticed using any other number of categoric values. This effect is a consequence of rigidity of the method *Following preference* and would be even more obvious if the preference curve was noisy (automatic generation) or had a particularly unsuitable shape.

Preference curve that emphasizes stiffness of *Following preference* method is ’stairs’ (Figure 8). In some cases of

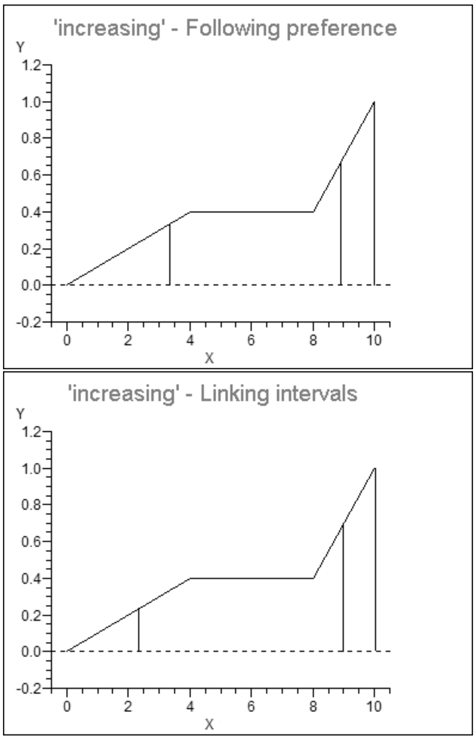


Figure 5: Results of categorization to three values given preference curve 'increasing'.

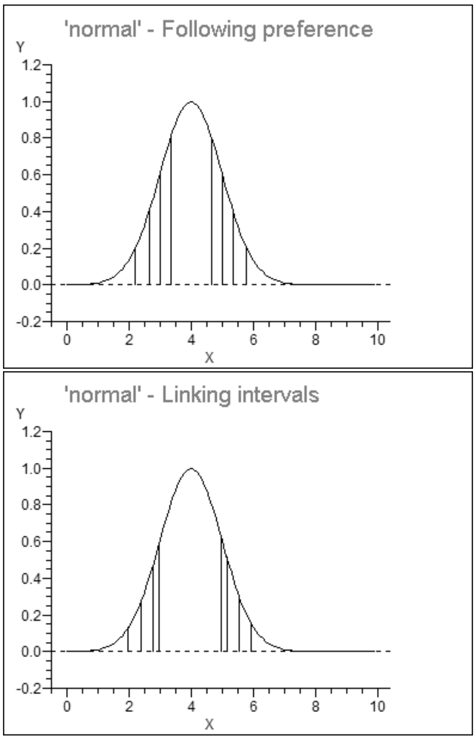


Figure 6: Results of categorization to five values given preference curve 'normal'.

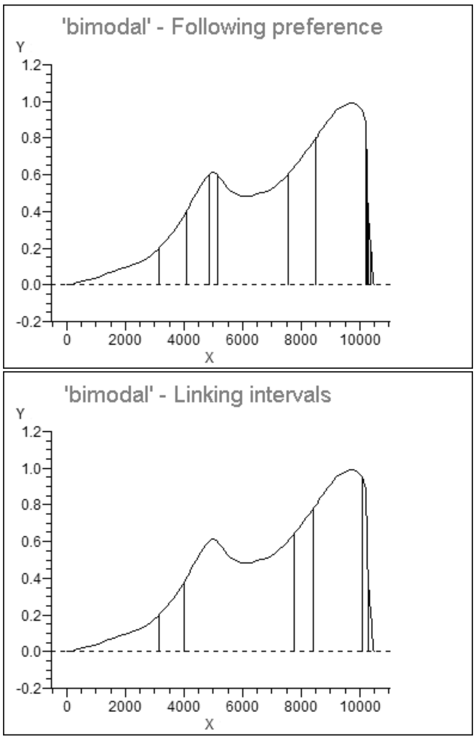


Figure 7: Results of categorization to five values given preference curve 'bimodal'.

desired number of categorical values, *Following preference* misses the natural course of preference curve. In addition to that, it suggests many unnecessary small intervals when preference curve suddenly drops. In such cases of preference curve we should use *Linking intervals*, that adapts itself to the shape of preference curve.

Generally the *Following preference* method provides suitable results. Proposed solutions of the *Linking intervals* method are usually less appropriate and for use with preference curves similar to Gaussian curve, not appropriate. However, in some cases of uncommon preference curves, *Linking intervals* provides a more natural and simple solution. None of the two methods is best in every situation, so both should be used with care.

4 Conclusion

We introduced two methods to simplify the categorization of numerical attributes in DEX methodology models. The main advantage of presented approach is providing the ability to present data and knowledge in a natural way, suitable for continuous data.

Experiments indicated some interesting features of the methods. Results of the *Following preference* method are generally better, but in certain cases its stiffness demonstrates in unnatural selection of interval bounds. In that situations the more flexible *Linking intervals* method usually gives more appropriate results.

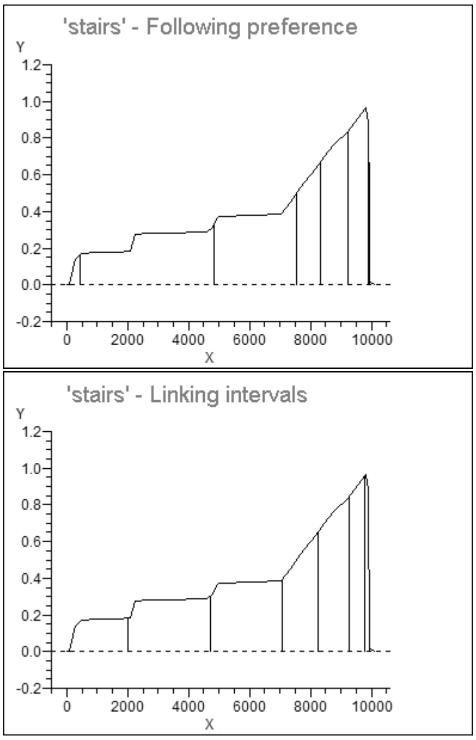


Figure 8: Results of categorization to six values given preference curve ‘stairs’.

Method features we discovered will have to be confirmed in practice. Practical experiences and further work are necessary to properly evaluate the applicability of proposed methods. Further work will be focused on development of a method that combines the good features of both presented methods and on study of situations where results of one of the methods prevail. Derivatives could be for instance used to detect changes in course of preference curves. As a minor improvement we plan to allow the input of preference curve in form of a mathematical expression. It would also be interesting to test the methods with automatically acquired preference curves.

References

[1] M. Bohanec and V. Rajkovič. DEX: An expert system shell for decision support. *Sistemica*, 1(1):145–157, 1990.

[2] M. Bohanec and V. Rajkovič. Multi-Attribute Decision Modeling: Industrial Applications of DEX. *Informatica*, 23(4):487–491, 1999.

[3] M. Bohanec, V. Rajkovič, and B. Cestnik. Five decision support applications. In D. Mladenić, N. Lavrač, M. Bohanec, and S. Moyle, editors, *Data Mining and decision support : integration and collaboration, (The Kluwer international series in engineering and computer science, SECS 745)*, pages 177–189. Kluwer Academic Publishers, Boston; Dordrecht; London, 2003.

[4] R. T. Clemen. *Making Hard Decisions: an Introduction to Decision Analysis*. Wadsworth Publishing Company, 1996.

[5] V. J. Easton and J. H. McColl. Statistics glossary - presenting data. www.cas.lancs.ac.uk/glossary_v1.1/presdata.html.

[6] E. Jereb, M. Bohanec, and V. Rajkovič. *DEXi-Računalniški program za večparametrsko odločanje (DEXi-Computer Program for Multi-Attribute Decision Making)*. Moderna organizacija, Kranj, SI, 2003.

[7] R. L. Keeney, H. Raiffa, and R. Meyer. *Decisions with Multiple Objectives: Preferences and Value Trade-offs*. Cambridge Univ Press, 1993.

[8] T. L. Saaty. *The Analytic Hierarchy Process*. McGraw-Hill, 1980.

[9] E. Turban and J. E. Aronson. *Decision Support Systems and Intelligent Systems*. Prentice Hall, 2001.