ELSEVIER

Decision Aiding

Assessing weights of product attributes from fuzzy knowledge in a dynamic environment

Yi-Chung Hu [a], Jian-Shiun Hu [b], Ruey-Shun Chen [c], Gwo-Hshiung Tzeng [d,*]

[a] *Department of Business Administration, Chung Yuan Christian University, Chungli 320, Taiwan, ROC*
[b] *Department of Information Management, Hsuan Chuang University, Hsinchu 300, Taiwan, ROC*
[c] *Institute of Information Management, National Chiao Tung University, Hsinchu 300, Taiwan, ROC*
[d] *Institute of Management of Technology, National Chiao Tung University, Hsinchu 300, Taiwan, ROC*

Abstract

Fuzzy knowledge of consumers' frequent purchase behaviors can be extracted from transaction databases. To effectively supporting decision makers, it is necessary to use fuzzy knowledge to assess weights or degrees of consumers' attentiveness to product attributes. From the standpoint of habitual domains, frequent purchase behaviors can be viewed as ideas that are contained in the reachable domain of customers. In addition, this reachable domain is changeable with time, due to the dynamic environment. This paper thus proposes a two-phase learning method with adaptive capability. The first phase builds a fuzzy knowledge base by discovering frequent purchase behaviors from transaction databases; the second phase finds weights of product attributes by a single-layer perceptron neural network. Indeed, customers are asked to evaluate alternatives and attributes through questionnaire. Then, each alternative can be transformed into a piece of input training data for the neural network by the fuzzy knowledge base and part-worths of attributes' levels. After completing the training task, we can find weights from connection weights. Simulation results demonstrate that the proposed methods can use fuzzy knowledge to effectively find customers' attentive degrees of attributes.

© 2002 Elsevier B.V. All rights reserved.

Keywords: Fuzzy sets; Neural networks; Data mining; Habitual domain; Decision-making

1. Introduction

Data mining is a methodology for the extraction of knowledge from data, specifically, knowledge relating to a problem that we want to solve [1]. Fuzzy knowledge relating to consumers' purchase behaviors such as "large amounts of orange juices were frequently purchased" or "small amounts of apple juice and large amounts of grape juice were frequently purchased" can be discovered from transaction databases by

---

data mining techniques. Weights or customers' attentive degrees of products' attributes are important factors in decision-making [3], indicating what people are most concerned with [2]. Thus, decision makers can be interested to know which attributes were attentive by customers in each purchase behavior, enabling the decision makers to propose a more competitive marketing strategy for promoting their products or improving their services. For example, a sales manager of one supermarket might wish to know which attributes are more important in purchase behaviors such as "large amounts of a particular brand, say Apple-1, are frequently purchased"? If the weight of attribute "price" is larger than those of other attributes, then that behavior may be interpreted as indicating that Apple-1 is cheaper than other brands, and it seems that a marketing strategy for reducing prices of other brands can be provided. For simplicity, each quantitative attribute of a database is linguistically interpreted as "purchase amounts" in later sections.

From the viewpoint of habitual domains, as proposed by Yu [3], all possible purchase behaviors can be viewed as ideas that are contained in the potential domain of customers. Ideas in the reachable domain usually can be activated with higher probability. That is, behaviors that frequently occurred for a period of time can be viewed as ideas that are contained in the reachable domain. In each behavior, there is a different grade of importance for each attribute considered by customers. However, the reachable domain can change with time because of a dynamic environment. For example, people can acquire new information or external stimuli from possible sources. Therefore, methods developed to find frequent purchase behaviors must be adaptive to a dynamic environment by treating decision-making as a dynamically adjusting process [4]. Methods with adaptive capability should thus be employed.

On the other hand, Tzeng et al. [4] proposed a learning algorithm that combined the habitual domain theory and winner-take-all learning rules to assess weights of criteria in multiple criteria decision-making, then applying the proposed model to effectively analyze mode-choice behaviors of Taipei City motorcycle users. On the other hand, Hashiyama et al. [5–8] proposed a fuzzy neural network to implement conjoint analysis, which is used to evaluate important attributes for customers' decision-making [9]. They identified relationships between part-worths of attributes' levels and overall evaluations of alternatives for selecting second-hand motorcycles. Moreover, their method could analyze attributes' weights for various combinations of linguistic interpretations (e.g., "price is cheap" and "external appearance is good"). From these previous studies, we can see that it is also feasible and necessary to assess weights of attributes from purchase behaviors to effectively support decision-making.

The goal of this paper is to propose an effective two-phase learning algorithm with adaptive capability that both extracts frequent purchase behaviors, and also finds attributes' weights from those frequent behaviors. Thus, we first propose a fuzzy data mining technique to discover frequent purchase behaviors from transaction databases to build a fuzzy knowledge base, and we also find weights of products' attributes by a single-layer perceptron (SLP) neural network. We stress frequent purchase behaviors since it seems that frequent purchase behaviors, whose individual fuzzy supports are larger than or equal to the user-specified minimum fuzzy support, are much more valuable for analysis than infrequent purchase behaviors. The definition of the fuzzy support is presented in the following section. Based on frequent purchase behaviors, we can build a fuzzy knowledge base consisting of fuzzy if-then rules. Subsequently, each alternative (e.g., Apple-1) is transformed to a training data of SLP by the fuzzy knowledge base and the evaluation of attributes through questionnaire. After completing the training task of the SLP, we can directly assess weights of products' attributes from connection weights.

In the following sections, the concepts of habitual domains are briefly introduced in Section 2. The simple fuzzy partition method is introduced in Section 3. In Section 4, we introduce the proposed algorithm and the method for constructing a fuzzy knowledge base. In Section 5, we propose a method together with an example in order to explain how to generate input training data from the fuzzy knowledge base and the part-worths of attributes' levels. In Section 6, a simulation is used to illustrate a detailed process for finding weights of products' attributes. Discussions and conclusions are presented in Sections 7 and 8, respectively.

2. Habitual domains

The set of ideas and concepts that are encoded and stored in the brain tend to progressively stabilize with time, and in the absence of an extraordinary destabilizing event, will approach a steady state [13] (i.e., habits). Purchase behaviors can be viewed as explicit expressions of ideas relating to purchase, and are gradually fixed by habits. There are four primary components in habitual domains [14]: (a) potential domain, PD_ω, which is the set of all possible ideas or operators that can be potentially activated and correspond to electrochemical patterns in the brain cell at time ω; (b) actual domain, AD_ω, which is the set of ideas or operators that are actually activated at time ω; (c) activated probability, AP_ω, which is the probability that a subset of PD_ω is actually activated; and, (d) reachable domain, $R(I_\omega, O_\omega)$, which is the set of ideas or operators that can potentially be reached from the initial set of ideas, I_ω, together with the initial set of operators, O_ω. Generally, ideas can be activated in thinking processes, and operators transform activated ideas into other ideas [4]. Actually, $R(I_\omega, O_\omega)$ is a subset of PD_ω and can be activated with higher probability. In addition, ω can be further viewed as a period of the past time.

For simplicity, the above-mentioned domains (i.e., PD_ω, AD_ω, and $R(I_\omega, O_\omega)$) can be thus viewed as individual sets of various purchase behaviors. Thus, at time ω, AD_ω is composed of purchase behaviors that actually occurred and $R(I_\omega, O_\omega)$ is composed of frequent purchase behaviors. Moreover, all possible purchase behaviors (i.e., infrequent or frequent behaviors) make up PD_ω at time ω. Actually, each purchase behavior in PD_ω contributes partial evaluation of the overall evaluation of an alternative (e.g., the aforementioned brand Apple-1) with multiple attributes. That is, it seems that there implicitly exist multiple fuzzy if-then rules, and the number of rules is equal to all possible purchase behaviors. We consider that the antecedent part of each fuzzy if-then rule is just a purchase behavior, and the consequent part provides an evaluation of the alternative. Then, the overall evaluation, obtained through a questionnaire, of that alternative is the combination of those evaluations from fuzzy rules. For example, according to past purchase habits (e.g., large amounts of orange juice were frequently purchased, or medium amounts of grape juice were frequently purchased), customers can evaluate competing alternatives (e.g., Apple-1) through a questionnaire. That is, an overall evaluation is obtained by fuzzy inferences with multiple fuzzy rules.

However, it also seems that those behaviors in the reachable domain, which can be activated with higher probability or frequently occurred, can give significant contributions. In addition, the reachable domain is changeable with time because of the dynamic environment. For effectively supporting decision-making, it is necessary to employ an effective learning method, which is adaptive to the dynamic environment, to find frequent purchase behaviors.

3. Fuzzy partition methods

The notation used in this paper is as follows:

d	total number of databases' attributes, where $d \geqslant 1$;
k	dimension of one fuzzy grid, where $1 \leqslant k \leqslant d$;
K	maximum number of linguistic values in each quantitative attribute of databases;
$A_{K,i_m}^{x_m}$	i_mth linguistic value of K linguistic values defined in linguistic variable x_m of databases, where $1 \leqslant m \leqslant d$, and $1 \leqslant i_m \leqslant K$;
$\mu_{K,i_m}^{x_m}$	membership function of $A_{K,i_m}^{x_m}$;
t_p	pth transaction, where $t_p = (t_{p_1}, t_{p_2}, \ldots, t_{p_d})$, and $p \geqslant 1$;
L	total number of frequent purchase behaviors;
r	total number of products' attributes in MADM;
s	total number of alternatives.

Fuzzy sets were initially proposed by Zadeh [15], who also proposed the concepts of the linguistic variables [16,17]. Formally, a linguistic variable is characterized by a quintuple [26], denoted by $(x, T(x), U, G, M)$, in which x is the name of the variable; $T(x)$ denotes the term set of x, that is, the set of names of linguistic values or terms, which are linguistic words or sentences in a natural language, of x; U denotes a universe of discourse; G is a syntactic rule for generating values of x; and M is a semantic rule for associating a linguistic value with a meaning. For example, we can view "Age" as a linguistic variable, $T(\text{Age}) = \{$young, close to 30, close to 50, old$\}$, where G is a rule which generates the linguistic values in $T(\text{Age})$, and $U = [0, 60]$. $M(\text{young})$ assigns a membership function to young.

In addition, the simple fuzzy partition method has also been widely used in pattern recognition and fuzzy reasoning. Two examples are the pattern classification problems solving by Ishibuchi and coworkers [11,12,19,20,29] and Hu et al. [10,31], and the fuzzy rules generation by Wang and Mendel [21]. In addition, several fuzzy methods for partitioning a feature space were discussed by Sun [22].

For partitioning quantitative attributes, in our methods, the maximum number of various linguistic values (i.e., K) in each quantitative attribute is prespecified by the system (i.e., a default assignment) before performing the proposed method. Of course, the decision makers can also subjectively determine the number, locations and shapes of fuzzy sets in each quantitative attribute depending on their preferences, past experiences, or prior knowledge. For simplicity, triangular membership functions are used for the linguistic values. However, we emphasize that Pedrycz [32] had pointed out the usefulness and effectiveness of the triangular membership functions in the fuzzy modeling. Hence, each linguistic value is a fuzzy number and its linguistic interpretation can thus be easily obtained. It is noted that a fuzzy number is a fuzzy set in the universe of discourse that is both convex and normal [18,23]. Generally, $\mu_{K,i_m}^{x_1}$ of $A_{K,i_m}^{x_1}$ is represented as follows:

$$\mu_{K,i_m}^{x_1}(x) = \max\{1 - |x - a_{i_m}^K|/b^K, 0\} \tag{1}$$

where

$$a_{i_m}^K = mi + (ma - mi) \cdot (i_m - 1)/(K - 1) \tag{2}$$

$$b^K = (ma - mi)/(K - 1) \tag{3}$$

where ma is the maximal value of domain, and mi is the minimum value. It is clear that $ma = 60$ and $mi = 0$ for the above-mentioned attribute x_1. Moreover, if we view x_1 as a linguistic variable, then the $A_{K,i_m}^{x_1}$ can be described in the sentences with different i_m:

$$A_{K,1}^{x_1}: \text{small} \tag{4}$$

$$A_{K,K}^{x_1}: \text{large} \tag{5}$$

$$\begin{aligned} A_{K,i_m}^{x_1}: &\text{ close to } (i_m - 1) \cdot [60 - 60/(K - 1)] \quad \text{and between} \quad (i_m - 2) \\ &\cdot [60 - 60/(K - 1)] \quad \text{and} \quad i_m \cdot [60 - 60/(K - 1)], \quad \text{for } 1 < i_m < K \end{aligned} \tag{6}$$

Actually, each linguistic value is viewed as a candidate one-dimensional (1-dim) fuzzy grid. Those candidate 1-dim fuzzy grids whose fuzzy supports are larger than or equal to the user-specified minimum fuzzy support are frequent. Furthermore, if we divide both x_1 and another quantitative attribute x_2 into three linguistic values, then a attribute space is divided into nine two-dimensional (i.e., 2-dim) candidate fuzzy grids, as shown in Fig. 1. For the shaded 2-dim fuzzy grid depicted in Fig. 1, we can obtain a corresponding purchase behavior denoted by $A_{3,1}^{x_1} \times A_{3,1}^{x_2}$, which is composed of two various linguistic values (i.e., "small" for x_1 and "small" for x_2). That is, fuzzy grids with any number of dimensions can be viewed as a purchase behavior. But frequent fuzzy grids relating to frequent purchase behaviors are more desirable.
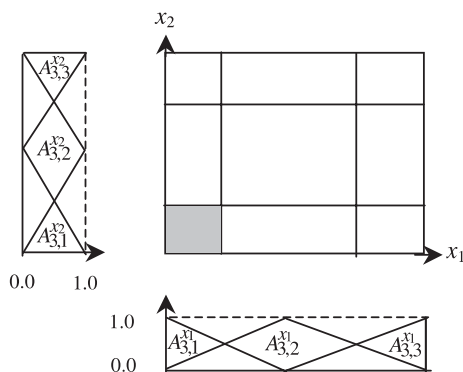
Fig. 1. $K = 3$ for x_1 and x_2.

The definition of the fuzzy support is introduced in Section 4 to determine if a candidate purchase behavior is frequent.

For partitioning categorical attributes, if the distinct attribute values are n' (n' is finite), then this attribute can only be divided into n' linguistic values. For example, the attribute "Gender" is categorical, and its values include "Female" and "Male"; then two linguistic values can be defined in "Gender". Also, each linguistic value is viewed as a candidate 1-dim fuzzy grid.

The next important task is using the initial 1-dim fuzzy grids to find frequent purchase behaviors. Therefore, we propose a two-phase learning method, as described in following sections.

4. Constructing a fuzzy knowledge base

The complete architecture that finds weighs of products' attributes from frequent purchase behaviors is shown in Fig. 2. From Fig. 2, we can see that frequent purchase behaviors are discovered from transaction databases. Based on frequent purchase behaviors, we can build a fuzzy knowledge base consisting of fuzzy if-then rules. Subsequently, each alternative is transformed to training data of the SLP by the fuzzy

Fig. 2. Architecture that can find attentive degrees of products' attributes from fuzzy concepts.

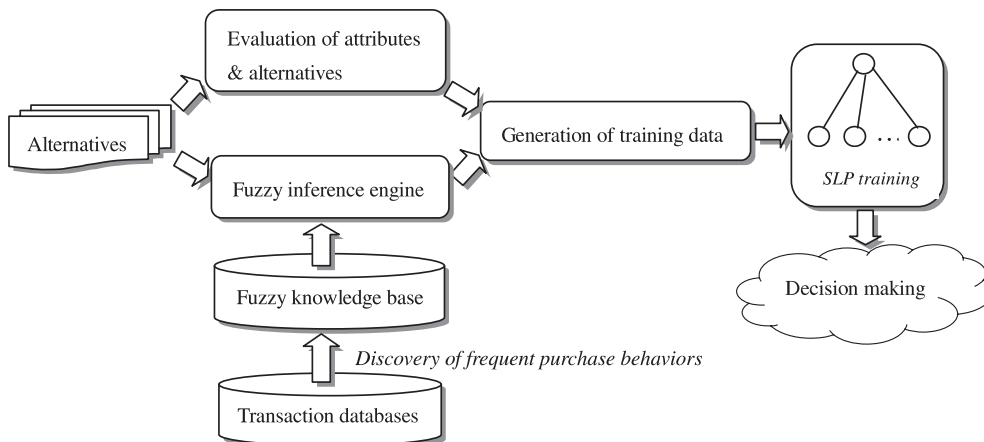knowledge base and the evaluation of attributes. After completing the training task of the SLP, we can directly assess weights of products' attributes from connection weights.

We describe the methods for discovering fuzzy knowledge in Section 4.1; the method for generating fuzzy rules is introduced in Section 4.2; and an example is demonstrated in Section 4.3.

4.1. Discovering fuzzy knowledge

In this section, we first introduce the definition of frequent fuzzy grids in Section 4.1.1. In Section 4.1.2, we describe data structures and table operations used in the proposed algorithm, which is presented in Section 4.1.3.

4.1.1. Frequent purchase behaviors

Suppose each quantitative attribute, x_m, is divided into K linguistic values. Without losing generality, given a candidate k-dim fuzzy grid $A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}$, which is a fuzzy subset, the degree which t_p belongs to this fuzzy grid can be computed as $\mu_{K,i_1}^{x_1}(t_{p_1}) \cdot \mu_{K,i_2}^{x_2}(t_{p_2}) \cdot \cdots \cdot \mu_{K,i_{k-1}}^{x_{k-1}}(t_{p_{k-1}}) \cdot \mu_{K,i_k}^{x_k}(t_{p_k})$. To check whether this fuzzy grid is frequent or not, we define the fuzzy support $\mathrm{FS}(A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k})$ [31,33] as follows:

$$\mathrm{FS}(A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}) = \left[\sum_{p=1}^{n} \mu_{K,i_1}^{x_1}(t_{p_1}) \cdot \mu_{K,i_2}^{x_2}(t_{p_2}) \cdot \cdots \cdot \mu_{K,i_{k-1}}^{x_{k-1}}(t_{p_{k-1}}) \cdot \mu_{K,i_k}^{x_k}(t_{p_k}) \right] \bigg/ n \quad (7)$$

The algebraic product [26] is actually used in (7). When $\mathrm{FS}(A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k})$ is larger than or equal to the user-specified minimum fuzzy support (minFS), we can say that $A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}$ is a frequent k-dim fuzzy grid. As we have mentioned in last section, those frequent fuzzy grids can be viewed as frequent purchase behaviors.

4.1.2. Data structures and table operations

Table FGTTFS is implemented to generate frequent purchase behaviors. It consists of the following substructures:

(a) Fuzzy grids table (FG): each row represents a fuzzy grid, and each column represents a linguistic value $A_{K,i_m}^{x_m}$. By using FG, we can easily determine which purchase behavior is generated and which linguistic values are contained in this purchase behavior.
(b) Transaction table (TT): each column represents t_p, while each element records the membership degree of the corresponding purchase behavior.
(c) Column FS: stores the fuzzy support corresponding to the purchase behavior in FG.

An initial tabular FGTTFS is shown in Table 1 as an example, from which we can see that there are two samples t_1 and t_2 and two attributes x_1 and x_2 in a given database. Both x_1 and x_2 are divided into three linguistic values (i.e., $K = 3$). Since each row of FG is a bits string consisting of 0 and 1, FG[u] and FG[v] (i.e., uth row and vth row of FG) can be combined to generate certain desired results by applying the Boolean operations. For example, a candidate 2-dim fuzzy grid $A_{3,1}^{x_1} \times A_{3,1}^{x_2}$ is generated by performing FG[1] OR FG[4] $= (1,0,0,1,0,0)$. It should be noted that the OR operation is just used to combine two rows of FG rather than combine various linguistic values. Then, $\mathrm{FS}(A_{3,1}^{x_1} \times A_{3,1}^{x_2}) = \mathrm{TT}[1] \cdot \mathrm{TT}[4] = [\mu_{3,1}^{x_1}(t_{1_1}) \cdot \mu_{3,1}^{x_2}(t_{1_2}) + \mu_{3,1}^{x_1}(t_{2_1}) \cdot \mu_{3,1}^{x_2}(t_{2_2})]/2$ is obtained to compare with the min FS. If $A_{3,1}^{x_1} \times A_{3,1}^{x_2}$ is large, then corresponding data (i.e., FG[1] OR FG[4], TT[1] · TT[4], and $\mathrm{FS}(A_{3,1}^{x_1} \times A_{3,1}^{x_2})$) will be inserted to corresponding data structures (i.e., FG, TT, and FS).

Table 1
Initial FGTTFS

Fuzzy grid	FG						TT		FS
	$A_{3,1}^{x_1}$	$A_{3,2}^{x_1}$	$A_{3,3}^{x_1}$	$A_{3,1}^{x_2}$	$A_{3,2}^{x_2}$	$A_{3,3}^{x_2}$	t_1	t_2	
$A_{3,1}^{x_1}$	1	0	0	0	0	0	$\mu_{3,1}^{x_1}(t_{1_1})$	$\mu_{3,1}^{x_1}(t_{2_1})$	$FS(A_{3,1}^{x_1})$
$A_{3,2}^{x_1}$	0	1	0	0	0	0	$\mu_{3,2}^{x_1}(t_{1_1})$	$\mu_{3,2}^{x_1}(t_{2_1})$	$FS(A_{3,2}^{x_1})$
$A_{3,3}^{x_1}$	0	0	1	0	0	0	$\mu_{3,3}^{x_1}(t_{1_1})$	$\mu_{3,3}^{x_1}(t_{2_1})$	$FS(A_{3,3}^{x_1})$
$A_{3,1}^{x_2}$	0	0	0	1	0	0	$\mu_{3,1}^{x_2}(t_{1_2})$	$\mu_{3,1}^{x_2}(t_{2_2})$	$FS(A_{3,1}^{x_2})$
$A_{3,2}^{x_2}$	0	0	0	0	1	0	$\mu_{3,2}^{x_2}(t_{1_2})$	$\mu_{3,2}^{x_2}(t_{2_2})$	$FS(A_{3,2}^{x_2})$
$A_{3,3}^{x_2}$	0	0	0	0	0	1	$\mu_{3,3}^{x_2}(t_{1_2})$	$\mu_{3,3}^{x_2}(t_{2_2})$	$FS(A_{3,3}^{x_2})$

In the well-known Apriori algorithm proposed by Agrawal et al. [30], two frequent $(k-1)$-itemsets are joined to be a candidate k-itemset $(3 \leqslant k \leqslant d)$, and these two frequent itemsets share $(k-2)$ items. Similarly, a candidate k-dim fuzzy grid is also derived by merging two frequent $(k-1)$-dim fuzzy grids, and these two frequent grids share $(k-2)$ linguistic values. For example, if $A_{3,2}^{x_1} \times A_{3,1}^{x_2}$ and $A_{3,2}^{x_1} \times A_{3,3}^{x_3}$ are two frequent fuzzy grids, then we can use $A_{3,2}^{x_1} \times A_{3,1}^{x_2}$ and $A_{3,2}^{x_1} \times A_{3,3}^{x_3}$ to generate the candidate 3-dim fuzzy grid $A_{3,2}^{x_1} \times A_{3,1}^{x_2} \times A_{3,3}^{x_3}$ because $A_{3,2}^{x_1} \times A_{3,1}^{x_2}$ and $A_{3,2}^{x_1} \times A_{3,3}^{x_3}$ share the linguistic term $A_{3,2}^{x_1}$.

However, $A_{3,2}^{x_1} \times A_{3,1}^{x_2} \times A_{3,3}^{x_3}$ can also be constructed from $A_{3,2}^{x_1} \times A_{3,1}^{x_2}$ and $A_{3,1}^{x_2} \times A_{3,3}^{x_3}$. This means that we must ensure that no extra constructions of a candidate fuzzy grid are made. To cope with this problem, the method we adopt here is that if there exist k integers numbers $e_1, e_2, \ldots, e_{k-1}, e_k$ where $1 \leqslant e_1 < e_2 < \cdots < e_{k-1} < e_k \leqslant d$, such that $FG[u, e_1] = FG[u, e_2] = \cdots = FG[u, e_{k-2}] = FG[u, e_{k-1}] = 1$ and $FG[v, e_1] = FG[v, e_2] = \cdots = FG[v, e_{k-2}] = FG[v, e_k] = 1$, where $FG[u]$ and $FG[v]$ correspond to frequent $(k-1)$-dim fuzzy grids, then $FG[u]$ and $FG[v]$ can be paired to generate a candidate k-dim fuzzy grid. However, it should be noted that any two linguistic values defined in the same attribute cannot be contained in the same candidate k-dim fuzzy grid $(k \geqslant 2)$. For example, since $FG[1]$ OR $FG[2] = (1, 0, 0, 0, 0, 0)$ OR $(0, 1, 0, 0, 0, 0)$, $(1, 1, 0, 0, 0, 0)$ is thus invalid. Therefore, $(1, 0, 1, 0, 0, 0)$, $(0, 0, 0, 1, 1, 0)$ and $(0, 0, 0, 1, 0, 1)$ are all invalid.

4.1.3. Discovering frequent purchase behaviors

We can employ the following algorithm to efficiently discover frequent purchase behaviors.

Algorithm: *An algorithm for finding frequent purchase behaviors*
Input: a. Transaction databases
     b. User-specified minimum fuzzy support
     c. *K*
Output: Frequent purchase behaviors
Method:
Step 1: Divide each quantitative attribute into various linguistic values
Step 2: Scan transaction databases, and then construct the initial FGTTFS
Step 3: Generate frequent 1-dim fuzzy grids
     Set $k = 1$ and eliminate the rows of the initial FGTTFS that correspond to infrequent 1-dim fuzzy grids.
Step 4: Generate frequent k-dim fuzzy grids $(k \geqslant 2)$
     Set $k := k + 1$. If there is only one $(k-1)$-dim fuzzy grid, then go to Step 5.
     For two unpaired rows, FGTTFS[u] and FGTTFS[v] $(u \neq v)$, corresponding to frequent $(k-1)$-dim fuzzy grids do
     Compute (FG[u] OR FG[v]) corresponding to a candidate k-dim fuzzy grid c.

4-1. Examine the validity of c. From nonzero elements of (FG[u] OR FG[v]), if any two linguistic values are defined in the same attribute, then discard c and skip Steps 4-2 and 4-3. That is, c is invalid.

4-2. Check if those two frequent grids share $(k - 2)$ linguistic values and c is not redundant.

4-3. Insert (FG[u] OR FG[v]) to FG, ($\text{TT}[e_1] \cdot \text{TT}[e_2] \cdots \text{TT}[e_k]$) to TT and fs to FS when the fuzzy support of c is larger than the minimum fuzzy support; otherwise, discard c.

　　End

Step 5: Check whether or not any frequent k-dim fuzzy grid is generated

If any frequent k-dim fuzzy grid is generated, then go to Step 4.

Frequent fuzzy grids discovered by the above algorithm are just frequent purchase behaviors. At the end of the mining task, the phase I is stopped. Before finding weights of attributes, a fuzzy knowledge base is build based on those frequent purchase behaviors demonstrated in the following section.

4.2. Generating fuzzy if-then rules

As we have mentioned in Section 2, there exist multiple fuzzy if-then rules for processing inferences, such that the antecedent part of each fuzzy if-then rule is a purchase behavior, and the consequent part provides an evaluation of the alternative. To implement the consequent part of each fuzzy rule, we use the simple additive weighting method. This method assumes that attributes are independent of each other and can yield extremely close approximations to "true" value functions [2] (i.e., overall evaluations), and we use it to summarize the part-worths of attribute levels with individual weights for each rule. Since partial evaluations are obtained from individual rules, we can further obtain the overall evaluation of an alternative. It is noted that the part-worth of an attribute stand for the measurement obtained by assigning the levels such as "how good," or "how satisfactory" to the attribute's value. The part-worths of various attribute levels can be acquired through the questionnaire.

We thus employ a fuzzy inference model, which is similar to the first-order Sugeno fuzzy model [24], whose consequent part can be used to identify relationships between part-worths of attributes' levels and the partial evaluation of each alternative. Significantly, the antecedence of each fuzzy rule in the fuzzy knowledge base is a frequent purchase behavior, and the consequence is a linear combination of attributes. Without loss of generality, if $A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}$ is the antecedence of the jth fuzzy rule $R^{(j)}$, then $R^{(j)}$ can be shown as following:

$$R^{(j)} : A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k} \rightarrow y_j^{(v)} = \sum_{z=1}^{r} \alpha_{j,z} f_z^{(v)} \tag{8}$$

where $0 \leqslant \alpha_{j,z} \leqslant 1$, $f_z^{(v)}$ is the part-worth of the zth attribute with respect to the vth alternative P_v $(1 \leqslant v \leqslant s)$ and $\alpha_{j,z}$ $(1 \leqslant j \leqslant L, 1 \leqslant z \leqslant r)$ is the weight of the zth attribute in $A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}$. Actually, $\alpha_{j,z}$ $(1 \leqslant j \leqslant L, 1 \leqslant z \leqslant r)$ is adjusted by presenting alternative P_v to SLP since $\alpha_{j,z}$ is served as a connection weight of the SLP architecture. The left-hand side of "$\rightarrow$" is the antecedent part of $R^{(j)}$ and the right-hand side is the consequent part. $R^{(j)}$ represents that: if x_1 is $A_{K,i_1}^{x_1}$ and x_2 is $A_{K,i_2}^{x_2}$ and $\cdots$ and x_k is $A_{K,i_k}^{x_k}$, then the evaluation of P_v provided by the jth rule is $y_j^{(v)}$. Significantly, $y_j^{(v)}$ is a partial evaluation of the overall evaluation o_v of P_v. In other words, since o_v of P_v is the combination of those evaluations from multiple fuzzy rules, o_v can be calculated by the weighted sum of partial evaluations shown as Eq. (9).

$$o_v = \sum_{j=1}^{L} \overline{w_j^{(v)}} y_j^{(v)} \tag{9}$$

Table 2
List of alternatives

Alternative	Attribute			
	Brand	Flavor	Capacity	Price
P_1	A	Orange	1000	50
P_2	A	Orange	375	15
P_3	B	Orange	1500	70
P_4	B	Orange	375	20
P_5	C	Orange	500	20
P_6	A	Apple	1000	60
P_7	A	Apple	375	20
P_8	B	Apple	2000	120
P_9	B	Apple	1000	65
P_{10}	C	Apple	500	15

Moreover,

$$\overline{w_j^{(v)}} = \frac{w_j^{(v)}}{\sum_i^L w_i^{(v)}} \tag{10}$$

where $w_j^{(v)}$ is the firing strength of the jth fuzzy rule for P_v. In the subsequent section, we give an example to demonstrate how a fuzzy knowledge base is built by the proposed algorithm introduced in Section 4.1.

4.3. An example

A list of products (i.e., alternatives) with four attributes (i.e., $r = 4$), sold in one supermarket is shown in Table 2. In practice, it is feasible to cluster all transactions in databases into several groups. That is, since the transactions in each customer group are similar with respect to the clustering variables (i.e., purchase amounts of various products), we can employ the proposed method to find frequent behaviors from the representative records (e.g., means of various groups). In this example, we assume that all transactions are clustered into 10 groups. The representative purchase records are stored in the database table PURCHASE shown in Table 3. Those products which are not purchased in t_p ($p = 1, \ldots, 10$) are marked by asterisks. In this case, even the size of transaction databases is very large, only few records can be taken into account for further processing.

Table 3
Database relation PURCHASE

Transaction	Alternative									
	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}
t_1	*	10	*	4	3	*	4	*	*	*
t_2	*	*	1	*	*	*	*	1	*	*
t_3	1	*	*	*	1	2	4	*	2	*
t_4	0	1	1	1	*	*	*	*	*	1
t_5	2	4	3	7	*	*	*	*	*	*
t_6	1	6	*	3	6	2	*	1	*	2
t_7	5	*	*	*	*	*	*	*	*	1
t_8	*	3	*	*	1	3	2	*	1	*
t_9	5	10	*	*	3	*	*	*	*	*
t_{10}	5	*	5	10	0	1	3	*	1	*

Table 4
PURCHASE-A transferred from PURCHASE

Transaction	Purchase amount	
	x_1 (Orange juice)	x_2 (Apple juice)
t_1	17	4
t_2	1	1
t_3	2	8
t_4	3	1
t_5	16	*
t_6	16	5
t_7	5	1
t_8	4	6
t_9	18	*
t_{10}	20	5

If we only wish to find frequent behaviors of purchasing orange juices and apple juices, then PUR-CHASE must be transformed into the desired form PURCHASE-A shown in Table 4. It is clear that there are two quantitative attributes in the PURCHASE-A: one is "amounts of orange juice that were purchased", denoted by x_1; and the other is "amounts of apple juice that were purchased", denoted by x_2. The pairs (*ma*, *mi*) for x_1 and x_2 are all assumed to be (1, 20). For PURCHASE-A, those asterisks provide membership value zero for any linguistic values in x_1 and x_2.

For simplicity, we consider $K = 3$ for each attribute (i.e., x_1 and x_2). Then, we can obtain two sets of linguistic values resulting from the simple fuzzy partition method; they are $\{A_{3,1}^{x_1}, A_{3,2}^{x_1}, A_{3,3}^{x_1}\}$ and $\{A_{3,1}^{x_2}, A_{3,2}^{x_2}, A_{3,3}^{x_2}\}$ defined in x_1 and x_2, respectively. The linguistic interpretation for each linguistic term is easily acquired. For example, $A_{3,1}^{x_1}$ can be interpreted as "small amounts of orange juices that were purchased". After scanning the PURCHASE-A, we can construct the initial FGTTFS shown in Table 5, where we can see that candidate 1-dim fuzzy grids are generated. Assuming that the user-specified minimum FS (min FS) is 0.25, we can rebuild FGTTFS shown in Table 6 resulting from deleting rows corresponding to infrequent fuzzy grids from initial FGTTFS.

From Table 6, we can find that FG[1] OR FG[2] = (1, 1, 0), which corresponds to an invalid 2-dim fuzzy grid $A_{3,1}^{x_1} \times A_{3,3}^{x_1}$ since those two linguistic values are defined in the same attribute x_1. We select FGTTFS[1] and FGTTFS[3] as an example to show how a frequent 2-dim fuzzy grid is generated. Clearly, FG[1] and TT[1] corresponding to $A_{3,1}^{x_1}$ are (1, 0, 0) and (0.00, 1.00, 0.89, 0.79, 0.00, 0.00, 0.58, 0.68, 0.00, 0.00), respectively. And FG[3] and TT[3] corresponding to $A_{3,3}^{x_1}$ are (0, 0, 1) and (0.68, 0.00, 0.00, 0.00, 0.58, 0.58, 0.00, 0.00, 0.79, 1.00), respectively. Thus, a candidate 2-dim fuzzy grid $A_{3,1}^{x_1} \times A_{3,1}^{x_2}$ is generated by FG[1] OR FG[3] = (1, 0, 1). In addition, the fuzzy support of $A_{3,1}^{x_1} \times A_{3,1}^{x_2}$ is computed by $(TT[1] \cdot TT[3]) = (0.00, 1.00, 0.24, 0.79, 0.00, 0.00, 0.58, 0.32, 0.00, 0.00) = 0.29$. Since *fs* is larger than the min FS (i.e., 0.25),

Table 5
Initial FGTTFS for PURCHASE-A

Fuzzy grid	FG						TT										FS
	$A_{3,1}^{x_1}$	$A_{3,2}^{x_1}$	$A_{3,3}^{x_1}$	$A_{3,1}^{x_2}$	$A_{3,2}^{x_2}$	$A_{3,3}^{x_2}$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	
$A_{3,1}^{x_1}$	1	0	0	0	0	0	0.00	1.00	0.89	0.79	0.00	0.00	0.58	0.68	0.00	0.00	0.39
$A_{3,2}^{x_1}$	0	1	0	0	0	0	0.32	0.00	0.11	0.21	0.42	0.42	0.42	0.32	0.21	0.00	0.24
$A_{3,3}^{x_1}$	0	0	1	0	0	0	0.68	0.00	0.00	0.00	0.58	0.58	0.00	0.00	0.79	1.00	0.36
$A_{3,1}^{x_2}$	0	0	0	1	0	0	0.68	1.00	0.26	1.00	0.00	0.58	1.00	0.47	0.00	0.58	0.56
$A_{3,2}^{x_2}$	0	0	0	0	1	0	0.32	0.00	0.74	0.00	0.00	0.42	0.00	0.53	0.00	0.42	0.24
$A_{3,3}^{x_2}$	0	0	0	0	0	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 6
Frequent 1-dim fuzzy grids inserted to rebuild FGTTFS

Frequent fuzzy grid	FG			TT										FS
	$A_{3,1}^{x_1}$	$A_{3,3}^{x_1}$	$A_{3,1}^{x_2}$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	
$A_{3,1}^{x_1}$	1	0	0	0.00	1.00	0.89	0.79	0.00	0.00	0.58	0.68	0.00	0.00	0.39
$A_{3,3}^{x_1}$	0	1	0	0.68	0.00	0.00	0.00	0.58	0.58	0.00	0.00	0.79	1.00	0.36
$A_{3,1}^{x_2}$	0	0	1	0.68	1.00	0.26	1.00	0.00	0.58	1.00	0.47	0.00	0.58	0.56

Table 7
Frequent 2-dim fuzzy grids

Frequent fuzzy grid	FG			TT										FS
	$A_{3,1}^{x_1}$	$A_{3,3}^{x_1}$	$A_{3,1}^{x_2}$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	
$A_{3,1}^{x_1} \times A_{3,1}^{x_2}$	1	1	0	0.00	1.00	0.24	0.79	0.00	0.00	0.58	0.32	0.00	0.00	0.29

we insert FG[1] OR FG[4], $TT_3[1] \cdot TT_3[4]$, and $FS(A_{3,1}^{x_1} \times A_{3,1}^{x_2})$ to corresponding data structures (i.e., FG, TT, and FS). Frequent 2-dim fuzzy grids that can be inserted to FGTTFS are shown in Table 7. Since no candidate 3-dim fuzzy grid can be further generated, we stop the proposed algorithm. Therefore, we find four purchase behaviors that frequently occurred (i.e., $A_{3,1}^{x_1}, A_{3,3}^{x_1}, A_{3,1}^{x_2}$, and $A_{3,1}^{x_1} \times A_{3,1}^{x_2}$).

Subsequently, the fuzzy knowledge base composed of four fuzzy if-then rules is shown as Eqs. (11)–(14).

$$R^{(1)} : A_{3,1}^{x_1} \rightarrow y_1^{(v)} = \alpha_{1,1}f_1^{(v)} + \alpha_{1,2}f_2^{(v)} + \alpha_{1,3}f_3^{(v)} + \alpha_{1,4}f_4^{(v)} \tag{11}$$

$$R^{(2)} : A_{3,3}^{x_1} \rightarrow y_2^{(v)} = \alpha_{2,1}f_1^{(v)} + \alpha_{2,2}f_2^{(v)} + \alpha_{2,3}f_3^{(v)} + \alpha_{2,4}f_4^{(v)} \tag{12}$$

$$R^{(3)} : A_{3,1}^{x_2} \rightarrow y_3^{(v)} = \alpha_{3,1}f_1^{(v)} + \alpha_{3,2}f_2^{(v)} + \alpha_{3,3}f_3^{(v)} + \alpha_{3,4}f_4^{(v)} \tag{13}$$

$$R^{(4)} : A_{3,1}^{x_1} \times A_{3,1}^{x_2} \rightarrow y_4^{(v)} = \alpha_{4,1}f_1^{(v)} + \alpha_{4,2}f_2^{(v)} + \alpha_{4,3}f_3^{(v)} + \alpha_{4,4}f_4^{(v)} \tag{14}$$

From weights of products' attributes in each purchase behavior, we can find the relative importance between attributes. For example, if $\alpha_{1,1}$ is largest among weights in $A_{3,1}^{x_1}$ (i.e., small amounts of orange juices that were frequently purchased), then "brand" is more attentive when customers wish to purchase small amounts of orange juices. Actually, those weights are obtained by training SLP, as we demonstrate in the following section.

5. Training single-layer perceptron

For the SLP employed in this paper, each input training data can be obtained by transforming alternative P_v ($1 \leqslant v \leqslant s$) to an appropriate form. From Eq. (9), we can further obtain the expression of the desired output, denoted by o_v, of P_v as follows.

$$o_v = \sum_{j=1}^{L} \overline{w_j^{(v)}} \left(\sum_{z=1}^{r} \alpha_{j,z} f_z^{(v)} \right) = \sum_{j=1}^{L} \overline{w_j^{(v)}} (\alpha_{j,1} f_1^{(v)} + \alpha_{j,2} f_2^{(v)} + \cdots + \alpha_{j,r-1} f_{r-1}^{(v)} + \alpha_{j,r} f_r^{(v)})$$

$$= \sum_{j=1}^{L} \alpha_{j,1} \left(\overline{w_j^{(v)}} f_1^{(v)} \right) + \alpha_{j,2} \left(\overline{w_j^{(v)}} f_2^{(v)} \right) + \cdots + \alpha_{j,r-1} \left(\overline{w_j^{(v)}} f_{r-1}^{(v)} \right) + \alpha_{j,r} \left(\overline{w_j^{(v)}} f_r^{(v)} \right) \tag{15}$$

Of course, both $f_z^{(v)}$ $(1 \leqslant z \leqslant r)$ and o_v must be given before training the SLP. From Eq. (15), we can see that each alternative can be transformed to an input training data of the SLP neural network by the fuzzy knowledge base and part-worths of attributes' levels. That is, P_v can be transformed to input training data $(\overline{w_1^{(v)}} f_1^{(v)}, \overline{w_1^{(v)}} f_2^{(v)}, \ldots, \overline{w_L^{(v)}} f_r^{(v)})$ with $L \cdot r$ dimensions. It is appropriate to design $\alpha_{j,z}$ to be the $[(j-1) \cdot r + z]$th connection weight $(1 \leqslant j \leqslant L, 1 \leqslant z \leqslant r)$ of SLP with $L \cdot r$ connections. The key issue is to obtain the firing strength $w_j^{(v)}$ of $R^{(j)}$ for P_v before proceeding to train SLP. On the other hand, since $\alpha_{j,z}$ ranges from zero to one, it is necessary to incorporate the Lagrange multiplier into SLP training. Below in Section 5.1, we describe the fuzzy projection method that can obtain $w_j^{(v)}$. The architecture of SLP with Lagrange multiplier used in our framework is briefly demonstrated in Section 5.2.

5.1. Deriving the firing strength from fuzzy rules

Since $A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}$ is a fuzzy subset, then we can define it as Eq. (16).

$$
\begin{aligned}
A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k} &= \sum_{p=1}^{n} \mu_{A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}} \\
&\quad \times (t_p, P_{x_1}^{(p)} \cup \cdots \cup P_{x_k}^{(p)}) / (t_p, P_{x_1}^{(p)} \cup \cdots \cup P_{x_k}^{(p)}) \\
&= \sum_{p=1}^{n} \mu_{A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}} (t_p, P^{(p)}) / (t_p, P^{(p)})
\end{aligned}
\tag{16}
$$

where $P_{x_m}^{(p)}$ $(1 \leqslant m \leqslant d)$ is a set of alternatives or products, that were actually purchased in transaction t_p, relating to x_m. For example, in Section 4.3, since x_2 in PURCHASE-A denotes "amounts of apple juice that were purchased", those products $P_{x_2}^{(1)} = \{P_7\}$ and $P_{x_2}^{(3)} = \{P_6, P_7, P_9\}$ were actually purchased in transaction t_1 and t_3, respectively. Without losing generality, let P_v belong to $P^{(p)}$ for t_p, and $A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}$ be the antecedence of $R^{(j)}$ as Eq. (8). Then we define $w_j^{(v)}$ as Eq. (17) by projecting fuzzy subspace [25,26] $A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}$ onto $\{P_v\}$.

$$
\begin{aligned}
w_j^{(v)} &= \mu_{A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_1} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k} \downarrow \{P_v\}}(\{P_v\}) \\
&= \max_{(t_p, P(p))} \mu_{A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_1} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}}(t_p, P(p)), \quad \text{where } P^{(p)} \text{ contains } \{P_v\} \quad \text{for } t_p
\end{aligned}
\tag{17}
$$

where "$\downarrow$" denotes the projection of $A_{K,i_1}^{x_1} \times A_{K,i_2}^{x_2} \times \cdots \times A_{K,i_{k-1}}^{x_{k-1}} \times A_{K,i_k}^{x_k}$ onto $\{P_v\}$.

Example. We use the PURCHASE-A shown in Section 4.3 to be an example. From Tables 6 and 7, $A_{3,1}^{x_1}, A_{3,3}^{x_1}, A_{3,1}^{x_2}$ and $A_{3,1}^{x_1} \times A_{3,1}^{x_2}$ can be represented as Eqs. (18)–(21), respectively.

$$
\begin{aligned}
A_{3,1}^{x_1} &= 1.00/(t_2, \{P_3\}) + 0.89/(t_3, \{P_1, P_5\}) + 0.79/(t_4, \{P_2, P_3, P_4\}) + 0.58/(t_7, \{P_1\}) \\
&\quad + 0.68/(t_8, \{P_2, P_5\})
\end{aligned}
\tag{18}
$$

$$
\begin{aligned}
A_{3,3}^{x_1} &= 0.68/(t_1, \{P_2, P_4, P_5\}) + 0.58/(t_5, \{P_1, P_2, P_3, P_4\}) + 0.58/(t_6, \{P_1, P_2, P_4, P_5\}) \\
&\quad + 0.79/(t_9, \{P_1, P_2, P_5\}) + 1.00/(t_{10}, \{P_1, P_2, P_4\})
\end{aligned}
\tag{19}
$$

$$
\begin{aligned}
A_{3,1}^{x_2} &= 0.68/(t_1, \{P_7\}) + 1.00/(t_2, \{P_7\}) + 0.26/(t_3, \{P_6, P_7, P_8, P_9\}) + 1.00/(t_4, \{P_{10}\}) \\
&\quad + 0.58/(t_6, \{P_6, P_8, P_{10}\}) + 1.00/(t_7, \{P_{10}\}) + 0.47/(t_8, \{P_6, P_7, P_9\}) \\
&\quad + 0.58/(t_{10}, \{P_6, P_7, P_9, P_{10}\})
\end{aligned}
\tag{20}
$$

Table 8
Firing strength of each rule

Alternative	Firing strength			
	$R^{(1)}$	$R^{(2)}$	$R^{(3)}$	$R^{(4)}$
P_1	0.89	0.79	0.00	0.58
P_2	0.79	0.79	0.00	0.79
P_3	1.00	0.58	0.00	1.00
P_4	0.79	1.00	0.00	0.79
P_5	0.89	0.79	0.00	0.32
P_6	0.00	0.00	0.58	0.32
P_7	0.00	0.00	1.00	1.00
P_8	0.00	0.00	0.58	0.23
P_9	0.00	0.00	0.58	0.32
P_{10}	0.00	0.00	1.00	0.79

$$A_{3,1}^{x_1} \times A_{3,1}^{x_2} = 1.00/(t_2, \{P_3, P_7\}) + 0.24/(t_3, \{P_1, P_5, P_6, P_7, P_8, P_9\}) + 0.79/(t_4, \{P_2, P_3, P_4, P_{10}\})$$
$$+ 0.58/(t_7, \{P_1, P_{10}\}) + 0.32/(t_8, \{P_2, P_5, P_6, P_7, P_9\}) \tag{21}$$

Therefore, we can obtain $w_j^{(v)}$ as shown in Table 8 (i.e., $w_j^{(v)}$ is placed in the vth row and jth column). For example, $w_1^{(2)} = \max\{\mu_{A3,1}^{x_1}(t_4, \{P_2, P_3, P_4\}), \mu_{A3,1}^{x_1}(t_8, \{P_2, P_5\})\} = \max\{0.79, 0.68\} = 0.79$. Note that, the desired output of each input data is the overall evaluation obtained by questionnaire.

5.2. Single-layer perceptron with Lagrange multiplier

The architecture of SLP used for assessing $\alpha_{j,z}$ is depicted in Fig. 3, where ζ_v is the actual output obtained by presenting alternative P_v to SLP. By incorporating the Lagrange multiplier into the learning rule, which was proposed by Zhang and Constantinides [27], we can restrict each weight to range from zero to one (i.e., $0 \leqslant \alpha_{j,z} \leqslant 1$ for $1 \leqslant j \leqslant L$ and $1 \leqslant z \leqslant r$). The cost function of the SLP can be described as Eq. (22).

$$E[\alpha_{j,z}, \lambda_{j,z,1}, \lambda_{j,z,2}, \delta_{j,z,1}, \delta_{j,z,2}] = \frac{1}{2} \sum_{v=1}^{s} [o_v - \zeta_v]^2 + \sum_{j=1}^{L} \sum_{z=1}^{r} \lambda_{j,z,1}(\alpha_{j,z} - 1 + \delta_{j,z,1}^2) + \sum_{j=1}^{L} \sum_{z=1}^{r} \lambda_{j,z,2}(-\alpha_{j,z} + \delta_{j,z,2}^2)$$
$$\tag{22}$$

where $\lambda_{j,z,1}$ and $\lambda_{j,z,2}$ are real numbers, and simply differentiable positive functions $\delta_{j,z,1}^2$ and $\delta_{j,z,2}^2$ are introduced in the cost function [27]. Moreover, o_v and ζ_v are the desired output and the actual output of P_v for SLP, respectively. State equations for each parameter are thus derived from Eq. (22) by the gradient descent method, which finds actual parameters to minimize the cost function [27]. Those state equations are thus described as Eqs. (23)–(27).
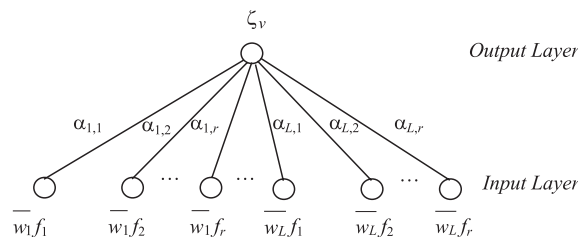

Fig. 3. Architecture of the SLP.

$$\frac{\mathrm{d}\alpha_{j,z}}{\mathrm{d}t} = -\frac{\partial E}{\partial \alpha_{j,z}} - \sum_{j=1}^{L}\sum_{z=1}^{r}\lambda_{j,z,1} + \sum_{j=1}^{L}\sum_{z=1}^{r}\lambda_{j,z,2} \tag{23}$$

$$\frac{\mathrm{d}\lambda_{j,z,1}}{\mathrm{d}t} = \alpha_{j,z} - 1 + \delta_{j,z,1}^{2} \tag{24}$$

$$\frac{\mathrm{d}\lambda_{j,z,2}}{\mathrm{d}t} = -\alpha_{j,z} + \delta_{j,z,2}^{2} \tag{25}$$

$$\frac{\mathrm{d}\delta_{j,z,1}}{\mathrm{d}t} = -2\lambda_{j,z,1}\delta_{j,z,1}, \tag{26}$$

$$\frac{\mathrm{d}\delta_{j,z,2}}{\mathrm{d}t} = -2\lambda_{j,z,2}\delta_{j,z,2} \tag{27}$$

where $1 \leqslant j \leqslant L$ and $1 \leqslant z \leqslant r$. Furthermore, learning rules can be directly derived from Eqs. (23)–(27). For example, the learning rule derived from Eq. (24) or $\lambda_{j,z,1}$ is described as Eq. (28).

$$\lambda_{j,z,1}(\beta + 1) = \lambda_{j,z,1}(\beta) + \eta(\alpha_{j,z} - 1 + \delta_{j,z,1}^{2}), \quad \beta \geqslant 0 \tag{28}$$

where β is referred to as the epoch size and η is the learning rate that should approximate zero to avoid the fluctuation that occurs in the learning process. Initial values of the other parameters (i.e., $\alpha_{j,z}$, $\lambda_{j,z,1}$, $\lambda_{j,z,2}$, $\delta_{j,z,1}$ and $\delta_{j,z,2}$) are set to small random values. In addition, we use an on-line learning strategy to adjust all parameters. That is, parameters are immediately updated by individual learning rules after each alternative has been presented to SLP [25] (i.e., one epoch). It is noted that the user-specified minimum mean-squared error is the stopping condition of SLP.

In the following section, we use the example demonstrated in Section 4.3 to examine the feasibility and effectiveness of the proposed algorithm by obtaining weights of products' attributes from the trained SLP.

6. Simulations

By the example illustrated in Section 4.3, we try to acquire weights (i.e., consumers' attentive degrees) of product attributes from the trained SLP. In fact, candidate customers for filling out a questionnaire can be selected from transaction databases which contain customer ID number and other information [28]. These people are allowed to give an overall evaluation of P_v only if they ever purchased P_v. For simplicity, we assume that part-worths of attributes and overall evaluations of alternatives had been acquired through the questionnaire. In fact, the part-worth of each attribute's level can be obtained by asking decision makers to pick a statement that best describes the given attribute's level [2]. Part-worths with respect to various attributes used here are briefly described as follows:

Brand: described on a three-point scale as unfavorable with part-worth 0.3, neutral with part-worth 0.6, and favorable with part-worth 0.9.
Flavor: described on a two-point scale as unfavorable with part-worth 0.5, and favorable with part-worth 0.8.
Capacity: described on a five-point scale as very unfavorable with part-worth 0.1, unfavorable with part-worth 0.3, neutral with part-worth 0.5, favorable with part-worth 0.75, and very favorable with part-worth 0.9.
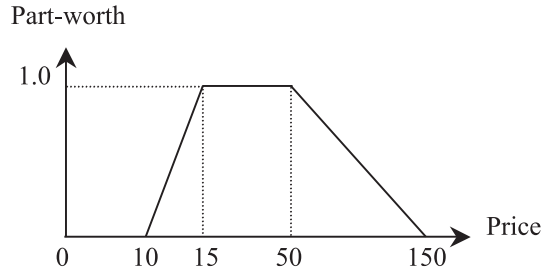Price: depicted in Fig. 4.

Fig. 4. Evaluation of prices.

Overall evaluation of alternatives: described on a three-point scale as unfavorable with part-worth 0.3, neutral with part-worth 0.6, and favorable with part-worth 0.9.

Part-worths of various attribute levels and overall evaluations of alternatives are summarized in Table 9 by the above-mentioned preference measures.

Before performing the training task, we set η and the tolerance error to be 10^{-6} and 10^{-5}, respectively. During the learning process, when the total cost is below 10^{-5} during training, SLP reaches convergence and thus stops. We can then obtain connection weights through the trained SLP. It should be noted that each alternative can be transformed to input training data with 16 dimensions (i.e., $L \cdot r = 4 \times 4 = 16$). Connection weights corresponding to the antecedence of each rule are shown in Table 10. We may further provide any possible analysis from simulation results.

Table 9
Evaluation of attributes and alternatives

Alternative	Part-worth				Desired output
	Brand	Flavor	Capacity	Price	
P_1	0.9	0.8	0.75	1.0	0.9
P_2	0.9	0.8	0.95	1.0	0.9
P_3	0.6	0.8	0.50	0.8	0.6
P_4	0.6	0.8	0.95	1.0	0.9
P_5	0.3	0.8	0.30	1.0	0.6
P_6	0.9	0.5	0.75	0.9	0.6
P_7	0.9	0.5	0.95	1.0	0.9
P_8	0.6	0.5	0.10	0.30	0.3
P_9	0.6	0.5	0.75	0.85	0.3
P_{10}	0.3	0.5	0.30	1.0	0.3

Table 10
Connection weights ($1 \leqslant j \leqslant 4$)

Rule label ($R^{(j)}$)	Antecedence	Weights			
		Brand	Flavor	Capacity	Price
$R^{(1)}$	$A_{3,1}^{x_1}$	0.1748	0.1892	0.1777	0.2044
$R^{(2)}$	$A_{3,3}^{x_1}$	0.1685	0.1829	0.1757	0.2007
$R^{(3)}$	$A_{3,1}^{x_2}$	0.1658	0.1489	0.1613	0.1518
$R^{(4)}$	$A_{3,1}^{x_1} \times A_{3,1}^{x_2}$	0.1895	0.1898	0.1907	0.1533

From Table 10, we can see that the major attribute considered in $A_{3,1}^{x_1}$ and $A_{3,3}^{x_1}$ is the price. This may mean that customers were attracted to buy orange juice on sale. That is, being on sale may promote orange juice. In addition, the two major attributes considered in $A_{3,1}^{x_2}$ (i.e., small amounts of apple juice that were purchased) are the brand and the capacity. If decision makers wants to promote those two attributes, then we can thus find the more popular brands with different capacities from transaction databases by a query language such as SQL. That is, query results can help retailers to plan which items should be placed in close proximity to promote the sale of apple juice. For instance, from the viewpoint of sale amounts for different brands, two products, including brand A and brand B, seems to be more popular. Moreover, from the viewpoint of sale amounts for different capacities, 375 and 1000 ml seems to be most popular. Hence, possible combinations of these brands and capacities may be placed close together in order to promote the sale.

However, in actual practice, managers may improve those attributes having worse performance such as the flavor. Then, a display should be designed to emphasize the advantages of drinking apple juice or the good taste of apple juice sold in the store. On the other hand, the major attribute considered in $A_{3,1}^{x_1} \times A_{3,1}^{x_2}$ is the capacity. Moreover, weights of both brand and flavor are quite similar. We can still use the database query language to find which capacities, brands or flavors are more popular. Then, proper layouts can be determined by investigating all possible combinations. Managers should also take into account the contents of displays to promote their products.

We stress that all alternatives could contribute to the SLP learning. In the numerical example, if the frequent behaviors are composed of $A_{3,1}^{x_1}$ and $A_{3,3}^{x_1}$, then we can use only P_1–P_5 to train the SLP. However, P_6–P_{10} can be also served as the input training data since the connection weights cannot be adjusted while presenting P_6–P_{10} to the SLP. That is, all alternatives can be served as the input training data.

We demonstrate that the proposed methods can effectively find customers' attentive degrees of attributes from connection weights of the trained SLP. It seems that the results can help decision makers to set up strategies for promoting products, improve services or plan store layouts.

7. Discussions

From the viewpoint of the habitual domain, the reachable domain, which includes frequent purchase behaviors, is changeable with time due to a dynamic environment. Each frequent purchase behavior implicitly serves as the antecedent part of a fuzzy rule that provides partial evaluation of each alternative. Furthermore, we can find weights (i.e., customers' attentive degrees) of attributes from those frequent purchase behaviors by using SLP to identify relationships between part-worths of various attribute levels and the overall evaluation of each alternative. By treating decision-making as a dynamically adjusting process, we propose an effective method with adaptive capability by combining a data mining technique that can discover frequent purchase behaviors together with the SLP that can find weights of various attributes. From simulation results, we can see that it is feasible to support decision makers to set up strategies for promoting products, improve services or plan store layouts. Furthermore, since the transactions in each group are similar with respect to the clustering variables (i.e., products), it is feasible to cluster all transactions in databases into several groups. That is, an alternative is to employ the proposed method to find frequent behaviors from the representative records rather than large scale transaction databases.

The weights assessed in the current period, say the first season, could support the decision makers to set up strategies for promoting products, improve services or plan store layouts for the next season. In the next season, the sets of frequent purchase behaviors must be extracted again from transaction databases since frequent behaviors will be influenced by the marketing strategies and advertising plans set up in the previous season. Moreover, the customers will re-evaluate alternatives and attributes through questionnaire.

Table 11
The comparisons of various methods for assessing weights

Method	Methodology	Input data
Tzeng et al.s' method	Habitual domain theory Fuzzy measure theory Winner-take-all technique	Connectivity matrix Tolerance value of each criterion Actual values of each criterion
Hashiyama et al.s' method	Multi-layer fuzzy neural network	Part-worths of various attribute levels and overall evaluations of alternatives
The proposed method	Habitual domain theory Fuzzy data mining technique Single-layer perceptron	Transaction databases Part-worths of various attribute levels and overall evaluations of alternatives

The contents of the questionnaire in the first season should be different from those in the second season. Then, the SLP must be re-trained with the transactions generated in the second season to assess the weights of product attributes. That is, each season has its unique SLP used to assess the weights. To verify the effectiveness of the strategies and the plans set up in the first season, the weights assessed in the first season will be compared with those assessed in the second seasons. From the aforementioned viewpoints, it seems that it is not necessary to test the generalization ability of the SLP used in this paper.

On the other hand, we can hardly compare the proposed method with the learning algorithm proposed by Tzeng et al. [4] and the fuzzy neural networks proposed by Hashiyama et al. [5–8]. The primary reason is that the input data for processing is quite different from each other since each method has its unique methodologies and characteristics. For example, in Tzeng et al.s' method, the respondents are asked to evaluate which attributes or criteria are influential through the first stage questionnaire. Then, a connectivity matrix is further constructed through the second stage questionnaire. This matrix is built by asking each respondent to evaluate the connectivity, which means the degrees of difficulty being associated from one criterion to another criterion [4]. As for the multi-layer fuzzy neural network proposed by Hashiyama et al., they only ask the respondents to evaluate alternatives and attributes through questionnaire. Moreover, the neural network can predict the upper bound and lower bound of the overall evaluation. For the comparisons of our method with the other methods, the methodologies and the input data used in various methods are briefly summarized in Table 11.

From Table 11, we can see that the data sources are mainly acquired through questionnaire in Tzeng et al.s' method and Hashiyama et al.s' method. However, to effectively find the customers' frequent behaviors, the transaction databases are further needed in our method for constructing a fuzzy rule base. Since each method is developed to analyze various data sources to assess weights, it would be difficult to criticize which one is superior to the other methods. To sum up, according to the goals and characteristics of the decision problems, the decision makers should carefully select one appropriate method to solve the decision problems.

Our method can also be applied to analyze another customer behaviors. For example, a bank could evaluate customers who already had home equity loans to determine the best strategy for increasing its market share [23]. Therefore, it seems to be appropriate for that bank to analyze weights of various attributes (e.g., interest rate) from the behavior of home equity loans.

8. Conclusions

Central to the multi-criteria decision analysis is to evaluate the weights or importance of criteria [3]. Significantly, based on the habitual domain theory, the proposed method is undertaken to develop a new model to assess weights of product weights using the fuzzy data mining technique and the SLP neural

network. Our method can help decision makers to plan marketing and advertising strategies, or to design store layouts, similar to market basket analysis [28].

In comparison with other traditional weighting methods [2] which usually identify weights of attributes for strategies, which have yet not been performed, we can assess customers' attentive degrees of products' attributes in each frequently purchase behavior discovered from transaction databases. Simulation results demonstrate that the proposed methods can effectively identify weights of attributes from connection weights of a trained SLP.

Acknowledgements

References

[1] S. Myra, Web usage mining for web site evaluation, Communications of the ACM 43 (8) (2000) 127–134.

[2] K.P. Yoon, C.L. Hwang, Multiple Attribute Decision Making: An Introduction, Sage Publications, London, 1995.

[3] P.L. Yu, Behavior bases and habitual domains of human decision/behavior-concepts and applications, in: G. Fandel, T. Gal (Eds.), Multiple Criteria Decision Making Theory and Application, Springer, New York, 1980, pp. 511–539.

[4] G.H. Tzeng, T.Y. Chen, J.C. Wang, A weight-assessing method with habitual domains, European Journal of Operational Research 110 (2) (1998) 342–367.

[5] T. Hashiyama, T. Furuhashi, Y. Uchikawa, A decision making model using a fuzzy neural network, in: Proceedings of the 2nd International Conference on Fuzzy Logic and Neural Networks, Iizuka, Japan, 1992, pp. 1057–1060.

[6] T. Hashiyama, T. Furuhashi, Y. Uchikawa, A fuzzy neural network for identifying changes of degrees of attention in a multi-attribute decision making process, in: Proceedings of International Joint Conference on Neural Networks, 1993, pp. 705–708.

[7] T. Hashiyama, T. Furuhashi, Y. Uchikawa, A proposal of a decision making model and its realization with a fuzzy neural network, Journal of Japan Society for Fuzzy Theory and Systems 5 (3) (1993) 587–599 (in Japanese).

[8] T. Hashiyama, T. Furuhashi, Y. Uchikawa, A study on varying degrees of attention in multi-attribute decision making processes, Journal of Japan Society for Fuzzy Theory and Systems 7 (4) (1995) 826–838 (in Japanese).

[9] J.J. Louviere, Analyzing Decision Making: Metric Conjoint Analysis, Sage Publications, London, 1988.

[10] Y.C. Hu, R.S. Chen, G.H. Tzeng, Finding fuzzy classification rules using data mining techniques, Pattern Recognition Letters 24 (2003) 509–519.

[11] H. Ishibuchi, K. Nozaki, N. Yamamoto, H. Tanaka, Selecting fuzzy if-then rules for classification problems using genetic algorithms, IEEE Transactions on Fuzzy Systems 3 (3) (1995) 260–270.

[12] H. Ishibuchi, T. Nakashima, T. Murata, Performance evaluation of fuzzy classifier systems for multidimensional pattern classification problems, IEEE Transactions on Systems, Man, and Cybernetics 29 (5) (1999) 601–618.

[13] P.L. Yu, Forming Winning Strategies: An Integrated Theory of Habitual Domains, Springer, New York, 1990.

[14] P.L. Yu, Habitual domains, Operations Research 39 (6) (1991) 869–876.

[15] L.A. Zadeh, Fuzzy sets, Information Control 8 (3) (1965) 338–353.

[16] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning, Information Science (Part 1) 8 (3) (1975) 199–249 (Part 2) 8(4) (1975) 301–357.

[17] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning, Information Science (Part 3) 9 (1) (1976) 43–80.

[18] S.M. Chen, W.T. Jong, Fuzzy query translation for relational database systems, IEEE Transactions on Systems, Man, and Cybernetics 27 (4) (1997) 714–721.

[19] H. Ishibuchi, T. Nakashima, T. Murata, Three-objective genetics-based machine learning for linguistic rule extraction, Information Sciences 136 (2001) 109–133.

[20] H. Ishibuchi, K. Nozaki, H. Tanaka, Efficient fuzzy partition of pattern space for classification problems, Fuzzy Sets and Systems 59 (3) (1993) 295–304.

[21] L.X. Wang, J.M. Mendel, Generating fuzzy rules by learning from examples, IEEE Transactions on Systems, Man, and Cybernetics 22 (6) (1992) 1414–1427.

[22] C.T. Sun, Rule-base structure identification in an adaptive-network-based fuzzy inference system, IEEE Transactions on Fuzzy Systems 2 (1) (1994) 64–73.

[23] M. Berry, G. Linoff, Data Mining Techniques: for Marketing, Sales, and Customer Support, John Wiley & Sons, New York, 1997.

[24] J.S.R. Jang, C.T. Sun, Neuro-fuzzy modeling and control, Proceedings of the IEEE 83 (3) (1995) 378–406.

[25] J.S. R. Jang, C.T. Sun, E. Mizutani, Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence, Prentice Hall, NJ, 1997.

[26] H.-J. Zimmermann, Fuzzy Set Theory and Its Applications, Kluwer, Boston, 1991.

[27] S.W. Zhang, A.G. Constantinides, Lagrange programming neural networks, IEEE Transactions on Circuits and Systems 39 (7) (1992) 441–452.

[28] J.W. Han, M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, San Francisco, 2001.

[29] H. Ishibuchi, T. Murata, A genetic-algorithm-based fuzzy partition method for pattern classification problems, in: F. Herrera, J.L. Verdegay (Eds.), Genetic Algorithm and Soft Computing, Physica-Verlag, Heidelberg, 1996, pp. 555–578.

[30] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, A.I. Verkamo, Fast discovery of association rules, in: U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy (Eds.), Advances in Knowledge Discovery and Data Mining, AAAI Press, Menlo Park, 1996, pp. 307–328.

[31] Y.C. Hu, R.S. Chen, G.H. Tzeng, Mining fuzzy association rules for classification problems, Computers and Industrial Engineering 43 (4) (2002) 735–750.

[32] W. Pedrycz, Why triangular membership functions?, Fuzzy Sets and Systems 64 (1994) 21–30.

[33] H. Ishibuchi, T. Yamamoto, T. Nakashima,. Fuzzy data mining: effect of fuzzy discretization, in: Proceedings of the 1st IEEE International Conference on Data Mining, San Jose, USA, 2001, pp. 241–248.