Front. Comput. Sci., 2014, 8(6): 933–947
DOI 10.1007/s11704-014-3359-4

RESEARCH ARTICLE

Feature selection on probabilistic symbolic objects

Djamal ZIANI (✉)

Information Systems Department, College of Computer and Information Sciences,
King Saud University, Riyadh 11543, Saudi Arabia

Abstract In data analysis tasks, we are often confronted to very high dimensional data. Based on the purpose of a data analysis study, feature selection will find and select the relevant subset of features from the original features. Many feature selection algorithms have been proposed in classical data analysis, but very few in symbolic data analysis (SDA) which is an extension of the classical data analysis, since it uses rich objects instead to simple matrices. A symbolic object, compared to the data used in classical data analysis can describe not only individuals, but also most of the time a cluster of individuals. In this paper we present an unsupervised feature selection algorithm on probabilistic symbolic objects (PSOs), with the purpose of discrimination. A PSO is a symbolic object that describes a cluster of individuals by modal variables using relative frequency distribution associated with each value. This paper presents new dissimilarity measures between PSOs, which are used as feature selection criteria, and explains how to reduce the complexity of the algorithm by using the discrimination matrix.

Keywords symbolic data analysis, feature selection, probabilistic symbolic object, discrimination criteria, data and knowledge visualization.

1 Introduction

In symbolic data analysis, an object is a representation of a group or a class of individuals. Variables used in symbolic objects can handle single quantitative values, single categorical values, intervals, and a set of values [1]. We have two categories of symbolic objects, boolean symbolic objects (BSO) and probabilistic symbolic objects (PSO). In BSO, the variables do not use modal descriptions; for example, Color = {red, green}; however in PSO, each value is followed by a probability, for example, Color = {0.7 red, 0.3 green}. This probability can have different semantics [2, 3]:

- A variation semantic is used to show the variation of individual properties inside a class. For example, the red color is very much used (90%) by the class individuals [4].
- A typical semantic can have different meanings [5], such as:
  1) Frequency: a property is typical if it is frequent in the class.
  2) Specificity: a property is typical if it is specific to this class, and not much used in other classes.
  3) Scholastic: a property is typical if it represents a state that matches a given theory model.

The PSOs are rich and complex objects; in literature, we can find some interesting research treating PSOs. Actually, we can have decision trees on PSOs, similarities and dissimilarities, and even distances for special representations of PSOs [6]. However, until now no feature selection algorithm has been developed on probabilistic symbolic objects. This gap in this research area motivated us to develop a new feature selection algorithm on probabilistic symbolic objects named Minset-Plus. Since the algorithm Minset-Plus has already been developed to treat BSOs [7], we will see in this paper how to adapt it for PSOs.

To adapt Minset-Plus for treating PSOs, we should first

Received September 24, 2013; accepted June 10, 2014

E-mail: dziani@ksu.edu.sa

find new feature selection criteria that can deal with PSOs. The selection criteria used in PSOs should take into consideration not only the descriptive values but also the probability associated with these values. We will cite in this paper some existing dissimilarity measures on PSOs, and we will present our new feature selection criterion, which is an improvement of an existing dissimilarity measure on PSOs.

Another challenge of feature selection in PSOs is the complexity. Since the PSOs are rich objects, the calculation of a similarity or dissimilarity measure needs complex operations; so the new algorithm should use an optimistic and efficient strategy to process the mandatory calculations and avoid redundant and irrelevant operations. Thus, we will see in the present paper how Minset-Plus algorithm will be optimized.

2 Symbolic objects

Let us give a formal definition of a symbolic object:

$\Omega = \{w_1, w_2, \ldots, w_p\}$ is the set of elementary objects.

$Y = \{y_1, y_2, \ldots, y_n\}$ is the set of variables. For example, $Y = \{age, weight, illness, \ldots\}$

$d = (d_1, d_2, \ldots, d_n)$ is the description of the object, where d_i is the value taken by the variable y_i. For example, $d = ([20, 25], [80, 90], \{diabetes, cholesterol\})$.

$L = \{true, false\}$ (for BSO) or $L = [0, 1]$ (for PSO)

$R = \{R_1, R_2, \ldots, R_p\}$ is a set of relations, where R_i is the relation used by the variable y_i. For instance, $R_1 = \subseteq$.

A symbolic object is defined as a triplet $s = (a, R, d)$, and this explanatory expression defines a symbolic object called an *assertion* [2]. For BSO, an assertion is represented by a symbolic expression, defined from Ω to $\{0, 1\}$:

$$a(w) = \bigwedge\nolimits_{i=1,n} [y_i(w) R_i d_i], \tag{1}$$

where $\wedge$ is the standard logical operator "AND".

For example, $a(w) = [age(w) \subseteq [20, 25]] \wedge [weight(w) \subseteq [80, 90]] \wedge [illness(w) \subseteq \{diabetes, cholesterol\}]$.

For PSO, an assertion is represented by a symbolic expression, defined from Ω to $[0, 1]$:

$$a(w) = \bigwedge\nolimits^{*}_{i=1,n} \left[y_i(w) R_i \{p_j v_j\}_{j=1,m} \right], \tag{2}$$

where $\bigwedge^{*}_{i=1,n} = \prod_{i=1,n}$ and p_j is the probability associated with the value v_j.

To evaluate the expression $a(w)$, we have to define the relation R_i used in Eq. (2). This relation defines the matching between values used in the description of object a and the values used in the description of the individual w. For instance, we can define the "matching" for two discrete density distributions $r = (r_1, r_2, \ldots, r_k)$ and distributions $q = (q_1, q_2, \ldots, q_k)$ of k values, by [2]:

$$rRq = \sum_{j=1,k} r_j q_j e^{(r_j - \min(r_j, q_j))}, \tag{3}$$

where r represents the probabilities used in the symbolic object, and q represents the probability of the elementary object values. If an elementary object value exists in the object a, the associated probability q will be equal to 1, else the associated probability q will be equal to 0.

We define an elementary event of a PSO as:

$$e_i(w) = \left[y_i(w) R_i \{p_j v_j\}_{j=1,m} \right]; \tag{4}$$

so,

$$a(w) = \bigwedge\nolimits^{*}_{i=1,n} e_i(w). \tag{5}$$

Example 1 Let:

$$a(w) = [age(w) \subseteq \{(0.2)[20, 25], (0.8)[26, 30]\}] \wedge^{*} [weight(w) \subseteq \{(0.4)[80, 85], (0.6)[86, 90]\}] \wedge^{*} [illness(w) = (1)\, diabetes].$$

The assertion a represents a cluster and expresses that 20% of the cluster individuals are aged between 20 and 25 years, 80% are between 26 and 30 years, 40% have a weight between 80 and 85 kg, 60% between 86 and 90 kg, and all individuals have diabetes as illness.

The symbolic object a extent is defined referring to Ω, and represents the set of elementary objects satisfying the following condition:

For BSO, $\text{ext}_{\Omega}(a) = \{w_i \in \Omega / a(w_i) = true\}$, and for PSO, by giving a threshold α, $\text{ext}_{\alpha}(a) = \{w_i \in \Omega / a(w_i) \geqslant \alpha\}$.

Example 2 We have an elementary object:

"*Alain*" $= [age = 23] \wedge [weight = 82] \wedge [illness = cholesterol]$.

The symbolic object:

$$a(w) = [age(w) \subseteq \{(0.2)[20, 25], (0.8)[26, 30]\}] \wedge^{*} [weight(w) \subseteq \{(0.4)[80, 85], (0.6)[86, 90]\}] \wedge^{*} [illness(w) = (1)\, diabetes].$$

Using a threshold $\alpha = 0.80$, we will check if Alain belongs to the extent of the PSO a:

$$a(Alain) = [age(Alain) \subseteq \{(0.2)[20, 25], (0.8)[26, 30]\}] \wedge^{*} [weight(Alain) \subseteq \{(0.4)[80, 85], (0.6)[86, 90]\}] \wedge^{*} [illness(Alain) = (1)\, diabetes]$$

Using the matching function defined in (3), we will have:

$$a(Alain) = 0.2 \times 1\, e^{(0.2-\min(0.2,1))} + 0.4 \times 1\, e^{(0.4-\min(0.4,1))} + 1 \times 0\, e^{(1-\min(1,0))},$$
$$a(Alain) = 0.2 \times 1 + 0.4 \times 1 + 0 = 0.6.$$

$a(Alain) < 0.8$, so in this case Alain does not belong to the extent of the PSO a.

Since Alain does not have diabetes as illness, his probability to belong to the extent of the PSO a is not bigger than 0.8.

3 Review of dissimilarity measure on PSO

All feature selection algorithms use a similarity or dissimilarity measure in order to select the features. The quality of the feature selection is highly dependent on the properties of the selection criteria used by the algorithm.

3.1 Dissimilarity properties

A dissimilarity measure D should satisfy some properties in order to be meaningful and strong. Let us define A as the set of symbolic objects.

A dissimilarity measure D is defined $A \times A \to [0, 1]$ [8], with:

- $D(a, b) \geqslant 0 \ \ \forall a, b \in A$ (non-negativity);
- $D(a, a) = 0 \ \ \forall a \in A$ (reflexivity);
- $D(a, b) = D(b, a) \ \ \forall a, b \in A$ (symmetry);
- $D(a, b) \leqslant D(a, c) + D(c, b) \ \ \forall a, b, c \in A$ (triangle inequality);
- $D(a, a^{-1}) = 1 \ \ \forall a \in A$ (opposition),

where a^{-1} is the opposite object of a. We can formalize an opposite object as follows:

$$\forall (p_j, v_j) \text{ in } a, \ \exists\, (p'_j, v_j) \text{ in } a^{-1}, \text{ where } p_j \times p'_j = 0.$$

It means, if $p_j \neq 0$, then $p'_j = 0$ and if $p_j = 0$, then $p'_j \neq 0$.

To understand the opposition property, let us have a small example:

$$a(Alain) = [age(Alain) \subseteq \{(0.2)[20, 25], (0.8)[26, 30], (0)[31, 40], (0)[41, 60]\}] \wedge^* [illness(Alain) = \{(1)diabetes, (0)cholesterol, (0)hypertension\}].$$

We define a dissimilarly measure D as:

$$D(a_l, a_k) = \prod_{i=1,n} \sum_{v_j \in O_j} \frac{\left| p_j i^{(l)} - p_j i^{(k)} \right|}{2},$$

where $p_j i^{(l)}$ and $p_j i^{(k)}$ are the probabilities associated to the value v_j for the variable y_i in the PSO a_l respectively a_k.

We can have many opposition objects of the symbolic object a, for example the object a_1 and a_2:

$$a_1(w) = [age(w) \subseteq \{[(0)[20, 25], (0)[26, 30], (1)[31, 40], (0)[41, 60]\}] \wedge^* [illness(w) = \{(0)diabetes, (1)cholesterol, (0)hypertension\}].$$

$$a_2(w) = [age(w) \subseteq \{[(0)[20, 25], (0)[26, 30], (0)[31, 40], (1)[41, 60]\}] \wedge^* [illness(w) = \{(0)diabetes, (0)cholesterol, (1)hypertension\}].$$

$$D(a, a_1) = \frac{(|0.2 - 0| + |0.8 - 0| + |0 - 1| + |0 - 0|)}{2} \times \frac{(|1 - 0| + |0 - 1| + |0 - 0|)}{2} = 1.$$

$$D(a, a_2) = \frac{(|0.2 - 0| + |0.8 - 0| + |0 - 0| + |0 - 1|)}{2} \times \frac{(|1 - 0| + |0 - 0| + |0 - 1|)}{2} = 1.$$

The dissimilarity measure of our algorithm should satisfy the following criteria: definition domain, reflexivity, symmetry and opposition. The triangle inequality property is a plus; in this case, the measure will be a metric. Also, some experts can give importance to the length of values taken by the variables; in this case, the dissimilarity measure should compute the Boolean part (the description space based on the values taken by the variables) and the probabilistic part of the objects.

The following will formalize the description space: Let us have: $a(w) = [y_1 = v_1] \wedge \cdots \wedge [y_n = v_n]$, $O = O_1 \times O_2 \times \cdots \times O_n$ is the Cartesian product of all variable values, where O_i represents the set of values that the variable y_i can take.

$A = \{a_1, a_2, \ldots, a_m\}$ is a set of probabilistic symbolic objects.

O^n is a vector of the values of O. $O^n = (O_1, O_2, \ldots, O_n)$.

The description space is defined by the function μ [4], and represents a vector of the values taken by a symbolic object:

$$\mu \colon A \to O^n, \quad \mu(a_i) = (v_{i1}, v_{i2}, \ldots, v_{in}). \tag{6}$$

where v_{ij} is the value taken by the variable y_j in the probabilistic symbolic object a_i.

Example 3 Suppose we have the following PSO:

$$a_1(w) = [age(w) \subseteq \{(0.2)[20, 25], (0.8)[26, 30]\}] \wedge [weight(w) \subseteq \{(0.4)[75, 80], (0.6)[81, 95]\}]$$

Figure 1 shows the description space of the object [4]; it is the Cartesian product of values taken by the variables in the object a_1. We can see that this Cartesian product consists of 4 parts $E11$, $E21$, $E31$, and $E41$. $E41$ is the biggest part; it is 12 times bigger than $E11$. Therefore, an expert can introduce in his dissimilarity measure this kind of appreciation.

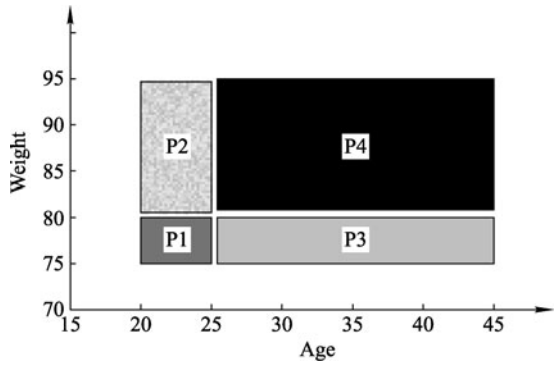

Fig. 1 Description space of the PSO a_1

3.2 Some existing dissimilarity measures on PSO

In literature, we can find very few dissimilarity measures on PSO, but we can find some dissimilarity coefficients that can be used to build a dissimilarity measure. A dissimilarity coefficient is calculated for one elementary event, and then aggregated for the whole object. It means that a dissimilarity coefficient is defined as a function on $A \times A \rightarrow [0, 1]$.

The following list gives some dissimilarity coefficients that were not defined on PSOs, but can be used as dissimilarity on PSOs:

- Rényi's divergence is defined as [9]:

$$m_r^{(s)}(e_i, e_i') = -\log\left(\sum_{v_j} p_j'^{s} \cdot p_j^{1-s}\right).$$

  where e_i, e_i' are 2 elementary events of the PSO a_i respectively, $a'i$; v_j is a value used in e_i and e_i'; and p_j, p_i' are the probability distributions associated with v_j in e_i respectively e_i'.
  We can note that this coefficient is not defined in [0, 1], it is not reflexive, it is not symmetric, and does not respect the opposite property.

- The Kullback–Leibler divergence is based on the difference of two probability distributions [10]:

$$m_{KL}(e_i, e_i') = \sum_{v_j} p_j' \log\left(\frac{p_j'}{p_j}\right).$$

  This coefficient is reflexive, but like Rényi's divergence coefficient, it is not symmetric, and does not respect the opposite property.

- The χ^2 divergence is defined as follows [11]:

$$\chi^2(e_i, e_i') = \sum_{v_j} \frac{|p_j - p_j'|^2}{p_j}.$$

  This coefficient is reflexive, but it is not symmetric, and does not respect the opposite property.

- The variation distance is defined as follows:

$$m_1(e_i, e_i') = \sum_{v_j} |p_j - p_j'|.$$

  This coefficient is reflexive and symmetric, but it is not respecting the opposite property.

The following list gives some dissimilarity measures defined on symbolic objects:

- Diday similarity coefficient [3]: Diday has defined a coefficient based on classical cosines similarity coefficient:

$$\mathrm{comp}(e_i, e_i') = \frac{\sum_{v_j} p_j \cdot p_j'}{\sqrt{\sum_{v_j} p_j^2 \cdot \sum_{v_j} p_j'^2}}.$$

  Hence, the dissimilarity coefficient will be:

$$\mathrm{cd}(e_i, e_i') = 1 - \frac{\sum_{v_j} p_j \cdot p_j'}{\sqrt{\sum_{v_j} p_j^2 \cdot \sum_{v_j} p_j'^2}}.$$

  This coefficient is not reflexive, it is symmetric and it respects the opposite property.

- Discrimination measure of Ziani [12]. Ziani has defined a discrimination (dissimilarity) measure between 2 PSOs. On elementary events, this measure is defined as follows:

$$g(e_i, e_i') = 1 - \left(\frac{\mathrm{card}(v_i \cap v_i')}{\mathrm{card}}(v_i \cup v_i') \times \frac{\sum_{v_j \mathrm{in} v_i \cup v_i'} 1 - |p_j - p_j'|}{r}\right)$$

  where v_i and v_i' are the values taken in the elementary event e_i respectively e_i'. pj and p_j' represent the probabilities associated to the value v_j respectively v_i', r is the number of different values which have a none null probability in the elementary event e_i and e_i'.
  This measure is reflexive, symmetric, and it takes into consideration the description space. However, this measure does not respect the opposite property.

- Probabilistic dissimilarity based on De Carvalho distance [13]: this dissimilarity between 2 PSOs is calculated using a distance between 2 elementary events, which is defined as follows:

$$\delta(e_i, e'_i) = \sum_{v_j \in v_i \cup v'_i} (\gamma_j p_j + \gamma'_j p'_j)^2,$$

where $\gamma_j = \begin{cases} 1, & \text{if } v_j \in v_i \text{ and } v_j \notin v'_i; \\ 0, & \text{otherwise,} \end{cases}$

and $\gamma'_j = \begin{cases} 1, & \text{if } v_j \notin v_i \text{ and } v_j \in v'_i; \\ 0, & \text{otherwise,} \end{cases}$

This dissimilarity is reflexive, symmetric, but this measure does not respect the opposite property.

Since all the dissimilarity measures that we found in literature do not respect all the properties that we want, we decided to introduce new dissimilarity measures on PSOs.

4 New dissimilarity measures on PSO

Before defining the new dissimilarity measures, we will first define the probabilistic space. The probabilistic space for a PSO is the description space of this object, where each vector element of this space is associated with a probability.

Let us have the description space $\mu(a_i) = (v_{i1}, v_{i2}, \ldots, v_{in})$, we define v a subset of description space $E_{rl} \subseteq \mu(a_i)$ as $E_{ri} = (v^{(r)}_{i1}, v^{(r)}_{i2}, \ldots, v^{(r)}_{in})$, where $\forall v^{(r)}_{ij} \subseteq v_{ij}$ (see the Example 4).

We will define two important functions θ and π.

$$\theta\colon O^n \to [0,1]^n$$
$$\theta(E_r) = (p^{(r)}_{i1}, p^{(r)}_{i2}, \ldots, p^{(r)}_{in}), \tag{7}$$

where $p^{(r)}_{li}$ is the probability associated with $v^{(r)}_{li}$.

$v^{(r)}_{li}$ is the value taken in the description space E_r by the variable y_i in object a_l.

$$\pi\colon [0,1]^n \to [0,1]$$
$$\pi(\theta(E_r)) = \prod_{j=1}^{n} p^{(r)}_{ij}. \tag{8}$$

If $E_i = \{E_{li}, \ldots, E_{li}\}$ is a description space of the PSO a_i, we calculate the probabilistic space of the PSO a_i as follows:

$$\rho\colon O^n \to [0,1]^n$$
$$\rho(Ei) = \Big(\pi(\theta(E_{1i})), \pi(\theta(E_{2i})), \ldots, \pi(\theta(E_{ni}))\Big). \tag{9}$$

We calculate the probability that an individual w belongs to a description space $E_i = \{E_{li}, \ldots, E_{li}\}$ is a description space of the PSO a_i, we calculate the probabilistic space of the PSO a_i as follows:

$$\sum_{E_{ri} \in E_i} \pi(\theta(E_{ri})) = \sum_{E_{ri} \in E} \prod_{j=1}^{n} p^{(r)}_{ij}. \tag{10}$$

Example 4 Using the object a_1 of the Example 3, we will have the following probabilistic space (see Fig. 2).

$$\mu(a_1) = (\{[20, 25], [26, 45]\}, \{[75, 80], [81, 95]\}).$$

We want to calculate the probability that an individual w belongs to the description space *weight* $\subseteq [75, 80]$. This means: $E = \{([20, 25], [26, 45]), ([75, 80], [81, 95])\}$, and $P(w \in E = \{E_{11}, E_{31}\}) = 0.20 \times 040 + 0.80 \times 0.40 = 0.40$.

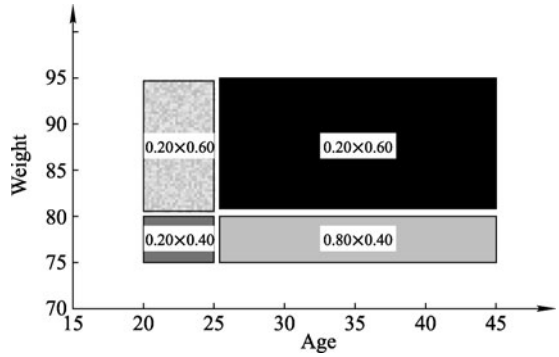

Fig. 2 Probabilistic space of the PSO a_1

We will introduce two new dissimilarity measures on PSOs. The first one is calculated using only the probabilities associated with the values; it is named a probabilistic dissimilarity measure. The second will take into consideration the description space of the values; it is named a description space probabilistic dissimilarity measure. Both dissimilarity measures should be reflexive, symmetric and respect the opposite property.

4.1 Probabilistic dissimilarity measure

The probabilistic dissimilarity measure, named d_a, is defined between 2 PSOs, as follows:

$$d_a\colon A \times A \to [0,1]$$
$$d_a(a_j, a_k) = \frac{1}{2}\Big|\pi\Big(\theta(\mu(a_j)) - \theta(\mu(a_k))\Big)\Big|. \tag{11}$$

Eq. (11) will be easily written as follows:

$$d_a(a_j, a_k) = \frac{1}{2} sum_{E_r \in \mu(a_j) \cup \mu(a_k)} \left| \prod_{i=1}^{n} p^{(r)}_{ij} - \prod_{i=1}^{n} p^{(r)}_{ik} \right|. \tag{12}$$

By using the projection of the symbolic object description on one variable, defined in Eq. (12), we can define a dissimilarity measure between two elementary events. The projection function is defined as follows:

$$\begin{aligned}&\mathrm{Proj}\colon O^n \times Y \to O\\ &\mathrm{Proj}\Big((v_{i1}^{(r)},\ldots,v_{in}^{(r)})/y_i\Big) = v_{li}^{(r)}.\end{aligned} \tag{13}$$

The dissimilarity measure between two elementary events is defined as follows:

$$\begin{aligned}&d_p\colon \varepsilon\times\varepsilon \to [0,1]\\ &d_p(e_{ij},e_{ik}) = \frac{1}{2}\left|\pi\Big(\theta\Big(\mathrm{Proj}(\mu(a_j)/y_i)\Big) - \theta\Big(\mathrm{Proj}(\mu(a_k)/y_i)\Big)\Big)\right|,\end{aligned} \tag{14}$$

where ε is the set of elementary events, and e_{ij}, e_{ik} are elementary events of the PSOs a_j respectively a_k, described by the variable y_i.

Using Eq. (10), $d_p(e_{ij},e_{ik})$ will be calculated as follows:

$$d_p(e_{ij},e_{ik}) = \frac{1}{2}\sum_{E_r\in\mu(e_{ij})\cup\mu(e_{ik})} |p_{ij}^{(r)} - p_{ik}^{(r)}|, \tag{15}$$

where $p_{ij}^{(r)}$, $p_{ik}^{(r)}$ are the probabilities associated with the value $v_{ij}^{(r)}$ and $v_{ik}^{(r)}$ in the PSOs a_j respectively a_k.

This measure satisfies the following properties:

- Reflexivity since: $d_p(e_{ij},e_{ij}) = 0 \forall e_{ij}\in\varepsilon$
- Symmetry since: $d_p(e_{ij},e_{ik}) = d_p(e_{ik},e_{ij}) \forall e_{ik},e_{ij}\in\varepsilon$
- $d_p(e_{ij},e_{ij}^{-1}) = 1 \forall e_{ij}\in\varepsilon$ (opposition)

Since it is easy to prove that this measure is reflexive, symmetric, and holds the opposition property, we will not write the proof in this paper.

4.2 Description space probabilistic dissimilarity measure

The description space probabilistic dissimilarity measure takes into consideration both probability and length of the value sets (cardinality of sets, or the length of interval). This dissimilarity measure is based on the dissimilarity measure d_p defined by Eq. (14). First, we will define the dissimilarity between two PSOs:

$$\begin{aligned}&d_{av}\colon A\times A\to[0,1]\\ &d_{av}(a_j,a_k)\\ &= \frac{1}{2}\sum_{E_r\in\mu(e_{ij})\cup\mu(e_{ik})}\left|\prod_{i=1}^{n}\frac{\mathrm{length}(v_{ij}^{(r)})}{\mathrm{cap}(\mu(e_{ij}))}p_{ij}^{(r)} - \prod_{i=1}^{n}\frac{\mathrm{length}(v_{ik}^{(r)})}{\mathrm{cap}(\mu(e_{ik}))}p_{ik}^{(r)}\right|,\end{aligned} \tag{16}$$

where $\cap(e_{ij}) = \sum_{r=1}^{m}\mathrm{length}(v_{ij}^{r})$.

v_{ij}^{r} is the r^{th} value taken in the elementary event e_{ij} in the PSO a_j.

The dissimilarity measure between two elementary events is defined as follows:

$$\begin{aligned}&d_{pv}\colon \varepsilon\times\varepsilon\to[0,1]\\ &d_{pv}(e_{ij},e_{ik}) = \frac{1}{2}\sum_{E_r\in\mu(e_{ij}\cup\mu(e_{ik})}\left|\frac{\mathrm{length}(v_{ij}^{(r)})}{\mathrm{cap}(\mu(e_{ij}))}p_{ij}^{(r)}\frac{\mathrm{length}(v_{ik}^{(r)})}{\mathrm{cap}(\mu(e_{ik}))}p_{ik}^{(r)}\right|.\end{aligned} \tag{17}$$

Example 5 Using the following PSOs, we will calculate defined dissimilarity measures.

$$\begin{aligned}a_1(w) = [&age(w)\subseteq\{(0.2)[20,25],(0.8)]25,45]\}]\\ &\wedge[weight(w)\subseteq\{(0.4)[75,80],(0.6)]80,95]\}].\end{aligned}$$

$$\begin{aligned}a_2(w) = [&age(w)\subseteq\{(0.6)[20,25],(0.4)]25,45]\}]\\ &\wedge[weight(w)\subseteq\{(0.7)[75,80],(0.3)]80,95]\}].\end{aligned}$$

Then,

$$\begin{aligned}d_{av}(a_1,a_2) = \tfrac{1}{2}(&|0.2\times0.4 - 0.6\times0.7|\\ &+|0.2\times0.6 - 0.6\times0.3|\\ &+|0.8\times0.4 - 0.4\times0.7|\\ &+|0.8\times0.6 - 0.4\times0.3|) = 0.4.\end{aligned}$$

$$\begin{aligned}d_{av}(a_1,a_2) = \frac{1}{2}\Big(&\left|\frac{5}{25}0.2\times\frac{5}{20}0.4 - \frac{5}{25}0.6\times\frac{5}{20}0.7\right|\\ &+\left|\frac{5}{25}0.2\times\frac{15}{20}0.6 - \frac{5}{25}0.6\times\frac{15}{20}0.3\right|\\ &+\left|\frac{20}{25}0.8\times\frac{5}{20}0.4 - \frac{20}{25}0.4\times\frac{5}{20}0.7\right|\\ &+\left|\frac{20}{25}0.8\times\frac{15}{20}0.6 - \frac{20}{25}0.4\times\frac{15}{20}0.3\right|\Big) = 0.12.\end{aligned}$$

$$\begin{aligned}&d_p(e_{11},e_{12}) = \frac{1}{2}(|0.2-0.6| + |0.8-0.4|) = 0.4.\\ &d_{pv}(e_{11},e_{12}) = \frac{1}{2}(|\frac{5}{25}0.2 - \frac{5}{25}0.6| + |\frac{20}{25}0.8 - \frac{20}{25}0.4|).\\ &d_{pv}(e_{11},e_{12}) = 0.2.\end{aligned}$$

Note To simplify the notation, we will use in the next sections only d_p dissimilarity measure, instead of d_{av} or d_pv.

5 Selection criteria

The feature selection algorithm that we developed is named Minset-Plus [7] and needs two criteria:

- The *discriminant* power used as stopping criteria,

- The *original discriminant* power used as selecting criteria.

5.1 Discrimination power

A is a set of assertions, q is a number of assertions in A, Y is a set of variables, and K is the set of assertion pairs $K = A \times A$.

The discriminant power of a variable y_l on the set K, noted by $DP(y_l, K)$, quantifies how much the variable y_l contributes in the discrimination of the assertion pairs.

$$DP\colon Y \times P(K) \to \mathbb{N}$$
$$DP(y_l, K) = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} d_p(e_{li}, e_{lk}), \tag{18}$$

with $(a_i, a_j) \in K$.

The discriminant power of a subset of variables Yd is defined as follows:

$$DP\colon P(Y) \times P(K) \to \mathcal{N}$$
$$DP(Yd, K) = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \max_{y_l \in Yd} d_p(e_{li}, e_{lk}). \tag{19}$$

5.2 Original discrimination power

The original discrimination power, noted ODP, of a variable y_l referred to a set of variable Yd, quantifies how much the variable y_l contributes to the discrimination of the assertion pairs which are not discriminated by any variable in Yd.

$$ODP\colon Y \times P(Y) \times P(K) \to \mathcal{N}$$
$$DP(y_l, Yd, K) = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \max_{(a_i, a_j) \in K} d_p(e_{li}, e_{lk}) \max_{y_p \in Yd} d_p(e_{pi}, e_{pk}). \tag{20}$$

Example 6 Let us calculate DP and ODP on the following set of objects:

$$a_1(w) = [age(w) \subseteq \{(0.2)[20, 25], (0.8)]25, 45]\}]$$
$$\wedge [weight(w) \subseteq \{(0.4)[75, 80], (0.6)]80, 95]\}].$$

$$a_2(w) = [age(w) \subseteq \{(0.6)[20, 25], (0.4)]25, 45]\}]$$
$$\wedge [weight(w) \subseteq \{(0.7)[75, 80], (0.3)]80, 95]\}].$$

$$a3(w) = [age(w) \subseteq \{(0.8)[20, 25], (0.2)]25, 45]\}]$$
$$\wedge [weight(w) \subseteq \{(0.9)[75, 80], (0.1)]80, 95]\}].$$

$$\begin{aligned} DP(\{age, weight\}, K) &= \max(d_p(e_{11}, e_{12}), d_p(e_{21}, e_{22})) \\ &\quad + \max(d_p(e_{11}, e_{13}), d_p(e_{21}, e_{23})) \\ &\quad + \max(d_p(e_{12}, e_{13}), d_p(e_{22}, e_{23})) \\ &= 0.4 + 0.6 + 0.2 = 1.2. \end{aligned}$$

$$\begin{aligned} &ODP(age, \{weight\}, K) \\ &\quad = \max(d_p(e_{11}, e_{12}) - d_p(e_{11}, e_{12}) - d_p(e_{21}, e_{22}), 0) \\ &\qquad + \max(d_p(e_{12}, e_{13}), 0) = 0.1 + 0.1 + 0 = 0.2. \end{aligned}$$

6 Minset-Plus **algorithm**

The Minset-Plus algorithm is formalized as follows [7]: initially we have a knowledge base (Y, O, A). The objective of the algorithm is to find another knowledge base (Y', O', A) such as $Y' \subseteq Y$ with $DP(Y', K) = DP(Y, K)$, where Y and Y' represent two sets of variables, O and O' represent the values taken by the variables of the set Y respectively Y', K represents the set of assertion pairs.

The following represents the algorithm Minset-Plus:

1) Find the indispensable variables that allow to discriminate couples of objects not discriminated by other variables. This means that we select all variables which have their ODP against all other variables not null: $ODP(y_i, Y - y_i, K) \neq 0$.
   Set $Y' = Y$.
   Set $Yd =$ set of selected variables
   While $DP(Yd, K) < DP(Y, K)$
2) Select in each step the variable y_l that has the highest ODP against all none selected variables.
   $$Y' = Y' - Yd$$
   $$Yd = Yd \cup \{y_l / y_l \text{ maximizes}$$
   $$ODP(y_i, Y' - y_i, K) \forall y_i \in Y'\}.$$
3) Eliminate in each step the variables which become redundant.
   $$Yd = Yd - \{y_l \in Yd \text{ where } ODP(y_l, Yd - y_l, K) = 0\}.$$

As you can see, the algorithm has three principal steps:

First, the algorithm selects the indispensable variables. A variable is considered as indispensable, if when you take it off from the set of variables, the DP of this set of variables will be less than the DP of all variables.

In the second step, the algorithm selects, in each iteration a variable with the biggest value in discriminating the parts of

symbolic objects, which are not discriminated yet by the selected variables. This step is performed by selecting the variable with the highest *ODP* against all none selected variables.

The purpose of the third step is to eliminate a variable that becomes redundant, it means that the part of discrimination brought by this variable has been covered by a combination of the other selected variables. This step is performed by eliminating any selected variable with a null *ODP* against all other selected variables.

7 Algorithm optimization

The calculation of the stopping criterion $DP(Yd, K)$ and the calculation of the selection criterion $ODP(y_i, Y' - y_i, K)$, are time consuming. In order to calculate the $ODP(y_i, Y' - y_i, K)$, the algorithm must execute $k \times (1 + p)$ times the function d_p, where $k = \text{card}(K)$ and $p = \text{card}(Y' - y_i)$. $DP(y_l, K)$ is calculated with $k \times p$ executions of the function d_p. To reduce the complexity of our algorithm, we optimized the calculation of the stopping criterion, by using a mathematical property, and we used the concept of discrimination matrix [7] to avoid unnecessary and redundant calculations.

7.1 Use of mathematical properties to reduce complexity

The following property allows us to calculate the discrimination power of a variable set, by adding the discrimination power of the old selected variables, with the original discrimination power of the current selected variable. The benefit of this property is to avoid calculating in each) step the discrimination power of the selected variables.

$$DP(Y_P \cup y_i, K) = DP(Y_P, K) + ODP(y_i, Y_P, K). \quad (21)$$

You will find the proof of this property in [7].

7.2 Discrimination matrix

We notice that the calculation of the *DP* and *ODP* functions is based on the calculation of $d_p(e_{li}, e_{lk})$. This is done repetitively in each step. Furthermore, we know that if we use the description space probabilistic dissimilarity measure, it involves the use of the operations $\cup$ and $\cap$ between sets of values; and these operations are not simple. To avoid executing the same complex operations many times, we saved the old calculations in order to reuse them in further algorithm steps. This idea has been completed by the introduction of the discrimination matrix.

The discrimination matrix allows us to calculate only one time $d_p(e_{li}, e_{lk})$; and during all the algorithm steps, we will use the matrix to do all the necessary operations. This is a great complexity optimization. Also, we want to emphasize that this matrix size is not big, it is: $k \times n$, where $k = \text{card}(K)$ and $n = \text{card}(Y)$, and K is not a big number, since we are dealing with objects that represent concept or individuals' classes.

Example 7 Let $Y = \{y_1, y_2, y_3, y_4, y_5\}$ is the set of variables, $A = \{a_1, a_2, a_3, a_4\}$, $A = \{a_1, a_2, a_3, a_4\}$. So: $K = \{(a_1, a_2), (a_1, a_3), (a_1, a_4), (a_2, a_3), (a_2, a_4), (a_3, a_4)\}$.

The discrimination matrix is represented in Table 1.

Table 1 Discrimination matrix

	(a_1, a_2)	(a_1, a_3)	(a_1, a_4)	(a_2, a_3)	(a_2, a_4)	(a_3, a_4)
y_1	0.7	0	0.3	0.1	0	0.1
y_2	0	0.6	0.1	0.7	0	0.4
y_3	0	0.6	0.5	0.3	0.6	0.3
y_4	0	0.2	0.4	0.2	0.5	0.5
y_5	0	0.3	0.4	0.3	0.6	0.3
Max Yd	0.7	0	0.3	0.1	0	0.1

During the calculation of $DP(Y, K)$ for the algorithm stopping criterion, the algorithm fills the discrimination matrix (only one time). Thus, in the case corresponding to row of y_l and the column (a_i, a_j), the algorithm puts the value of $d_p(e_{li}, e_{lk})$. The Max Yd row is used to save all $\max_{y_p \in Yd}(d_p(e_{pi}, e_{pk}))$.

At the beginning all the cases of Max Yd row are empty, and they will be updated with the maximum values of the indispensable variables' cases.

In this example, y_1 is indispensable, so the algorithm updates Max Yd row with the values saved in the row of y_1 (it means all of $d_p(e_{1i}, e_{1k})$ values).

7.3 Operating with discrimination matrix

The discrimination matrix is used in the three important algorithm steps: selecting the indispensable variables, selecting a new variable, and finding redundant variables.

- Selecting the indispensable variables will be done by doing this test: y_l is indispensable if

$$\exists (a_i, a_j) \in K, \text{ where } d_p(e_{li}, e_{lj}) \neq 0 \text{ and } \max_{y_p \in Y - y_l}(d_p(e_{pi}, e_{pk})) = 0. \quad (22)$$

  This means that a variable is indispensable, if we find a pair of objects discriminated by this variable and not discriminated at all by any other variable. When using the discrimination matrix, we can find indispensable variables without doing complex operations; we only have to check if the discrimination values of y_l, stored

in the discrimination matrix, is greater than the discrimination values of all other variables (they are also stored in the discrimination matrix).

- Selecting a new variable in each step is done by calculating for each unselected variable the value $ODP(y_l, Yd, K)$. Based on Eq. (19) and by using the discrimination matrix, the new variable's selection is done as follows: we know that all values $d_p(e_{pi}, e_{pk})$ are saved in the case corresponding to the row y_l and the column (a_i, a_j) of the discrimination matrix. And we also know that $\max_{y_p \in Yd}(d_p(e_{pi}, e_{pk}))$ is saved in row Max Yd's case. Thus, to calculate of $ODP(y_l, Yd, K)$ the algorithm will do, for each couple of objects (a_i, a_j), only one subtraction between two numbers (the value saved in the case corresponding to the row y_l and the column (a_i, a_j), and the case of the same column in row Max Yd), and it also needs one comparison between the calculated value and zero.
- Finding the redundant variables in each step is generally done by checking if the selected variables are redundant when selecting a new variable; this means $ODP(y_l, (Y_d \cup y_s) - y_l, K) = 0$, where y_s is the new selected variable. Using Eq. (20), we will calculate this as follows:

$$DP((Y_d \cup y_s) - (y_l - y_s), K) = DP(Y_d \cup y_s, K). \quad (23)$$

This test is not complex when using the discrimination matrix, since to compute the expression $DP(Y_d \cup y_s, K)$, the algorithm calculates, for each couple (a_i, a_j), the maximum between the value saved in the case corresponding to the row y_s and the column (a_i, a_j), and the value saved in Max Yd for the same column:

$$\sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \text{Max}(\text{Max}_{i,j} Y_d, d_p(e_{si}, e_{sj})).$$

Therefore, $DP((Y_d \cup y_s) - (y_l - y_s), K)$ will be calculated as follows:

$$\sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \text{Max}(\text{Max}(\max_{y_l \in Yd} d_p(e_{li}, e_{sl}), d_p(e_{si}, e_{sj})) - \text{Max}(d_p(e_{li}, e_{sl}) - d_p(e_{si}, e_{sj})), 0)$$

On the other hand, we know that when every time the algorithm selects a variable y_s, it calculates $\text{Max}(\max_{y_l \in Yd} d_p(e_{li}, e_{sl}), d_p(e_{si}, e_{sj}))$, and its value is saved in the discrimination matrix in row Max Yd. This means that the algorithm have to compute, for each object pair (a_i, a_j), only the subtraction of the case value $(a_i, a_j), y_l$ and the case value $(a_i, a_j), y_s$, and then it compares the substation value with the value found in Max Yd which correspond to the same object pair (a_i, a_j) .

8 Application

8.1 PSO data format

The data format is a critical and very important aspect to take into consideration before doing the feature selection. If the data are not well formatted, this will lead to an incorrect result. For this purpose, we begin our study in the symbolic objects generation's process, a step before the feature selection phase, in order to establish rules for generating objects with the right format.

The most important rule we used for PSO generation is the following: all elementary events of PSO assertions using the same variables and belonging to the same data set should use the same values, in order to be able to compare their probability distributions correctly.

To ensure that all PSOs of a dataset are using the same values, we introduced the notion of split value. The split value is the initial value used by a symbolic object (BSO or PSO) that will be split to a set of values, by including the values used by other objects.

Example 8 We have two BSOs a_1 and a_2 and we will format them in order to have the same values.

$$a_1 = [age \subseteq \{[20, 65]\}]. a_2 = [age \subseteq \{[10, 25], [46, 65]\}].$$

When we process a_1 taking into consideration a_2, the split value is $[20, 65]$; this value will be split to: $[10, 20[$, $[20, 25]$, $[25, 46]$ and $[46, 65]$.

During the PSO generation phase, three options can be used by experts for full-filing the cited rule:

1) The probability of the split value should be equally distributed among the values resulting from this split.
2) The probability of the split value should be distributed among the values resulting from this split, accordingly to their length.
3) The probability of the split value should be affected to the values resulting from this split.

Option 1 and 2 ensure that the sum of the probability distribution among the elementary events is always equal 1.

Example 9 We have two PSOs a_1 and a_2, and we will format them in order to have the same values:

$$a_1 = [age \subseteq \{(0.2)[20, 45], (0.8)[46, 65]\}].$$
$$a_2 = [age \subseteq \{(0.4)[20, 25], (0.6)[26, 65]\}].$$

- If we use the split option 1, we will have:

$$a_1 = [age \subseteq \{(0.1)[20, 25], (0.1)[26, 45], (0.8)[46, 65]\}].$$
$$a_2 = [age \subseteq \{(0.4)[20, 25], (0.3)[26, 45], (0.3)[46, 65]\}].$$

- If we use the split option 2, we will have:

$$a_1 = [age \subseteq \{(0.2 \times r_1)[20, 25], (0.2 \times r_1)[20, 25], (0.8)[46, 65]\}].$$

$$a_2 = [age \subseteq \{(0.4)[20, 25], (0.6 \times r_3)[26, 45], (0.6 \times r_4)[46, 65]\}].$$

$$r_1 = \frac{\text{length}([20, 25])}{\text{length}([20, 45])} = \frac{(25 - 20) + 1}{(45 - 20) + 1} = 0.23.$$

$$r_2 = \frac{\text{length}([26, 45])}{\text{length}([20, 45])} = \frac{(45 - 26) + 1}{(45 - 20) + 1} = 0.77.$$

$$r_3 = \frac{\text{length}([26, 45])}{\text{length}([26, 65])} = \frac{(45 - 26) + 1}{(65 - 26) + 1} = 0.5.$$

$$r_4 = \frac{\text{length}([46, 65])}{\text{length}([26, 65])} = \frac{(65 - 46) + 1}{(65 - 26) + 1} = 0.5.$$

$$a_1 = [age \subseteq \{(0.046)[20, 25], (0.154)[26, 45], (0.8)[46, 65]\}].$$

$$a_2 = [age \subseteq \{(0.4)[20, 25], (0.3)[26, 45], (0.3)[46, 65]\}].$$

- If we use the split option 3, we will have:

$$a_1 = [age \subseteq \{(0.2)[20, 25], (0.2)[26, 45], (0.8)[46, 65]\}].$$

$$a_2 = [age \subseteq \{(0.4)[20, 25], (0.6)[26, 45], (0.6)[46, 65]\}].$$

8.2 Validation

After each selection process, it is necessary to validate the results. The validation process used can be divided into two categories: the validation without test data and the validation with test data.

8.2.1 Validation process without test data

In the validation without test data, the expert can assess the feature selection result by analyzing the selected variables' list, generated by Minset-Plus algorithm. The list includes some information which help him to evaluate the selected variables' quality: name of the variable, type of the variable (quantitative, qualitative, status of the variable (indispensable or not), and selection step (algorithm step, the discrimination power of the selected variables during this step, and the original discrimination power of this variable referring to the selected variables).

Also by using the flexibility provided by Minset-Plus algorithm, the expert can force the selection, and can discard the selection of some variables. Furthermore, the expert can set the level when the algorithm will stop (what percentage of discrimination power to reach). Thus, the expert can repeat the selection process until he gets satisfactory results, by using either the initial objects or the objects generated by Minset-Plus algorithm after selection.

The validation process without test data is shown by Fig. 3.

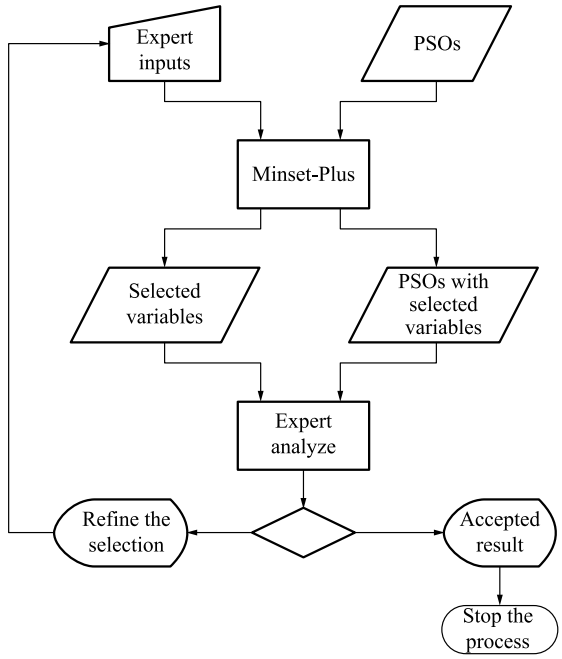

Fig. 3 Validation process without test data

8.2.2 Validation process with test data

In our study, the validation process with test data is done based on object extent calculation. The extent of the PSOs after feature selection should not be far from the extent of the PSOs before the feature selection. If the objects' extent using the selected variables is bigger than the objects' extent before selection, it means that the individuals will have, after feature selection, more chances to be in the intersection of the object extents. In this case, we can say that the objects were not well discriminated using the selected variables. The validation, using object extents, needs individuals test data. This

test data are provided by the expert or generated automatically using the symbolic object generator program, developed for this purpose.

The validation process with test data is shown by Fig. 4.

The quality assessment of the selected variables is done using two criteria:

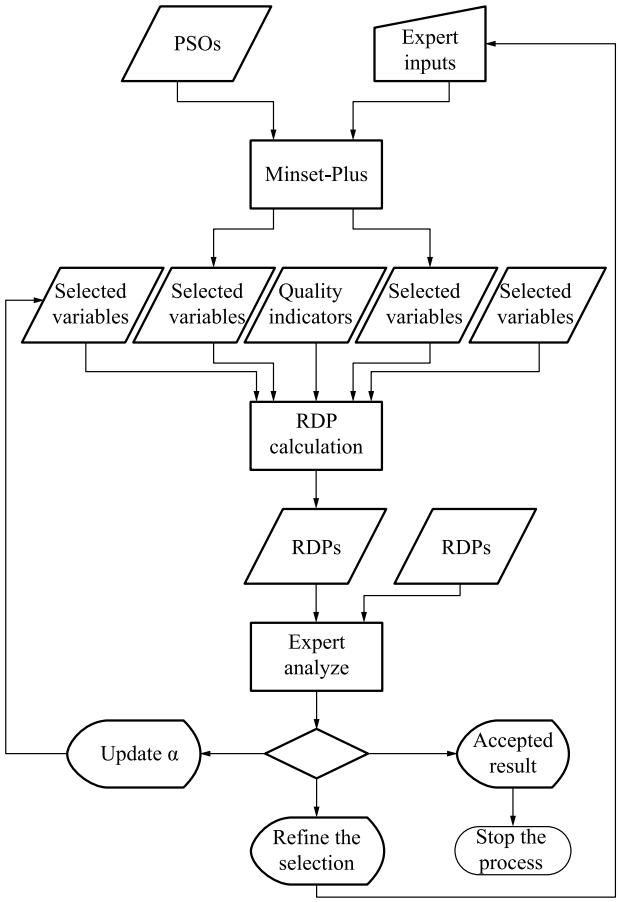

Fig. 4 Validation process with test data

- Real discrimination power variation (RDPV) criterion: defined as follows:

$$\left|\frac{\mathrm{RDP}(Y_d, K) - \mathrm{RDP}(Y, K)}{\mathrm{RDP}(Y_d, K)}\right| \leqslant \beta, \qquad (24)$$

where

$$\mathrm{RDP}(Y_d, K) = 1 - \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \frac{\mathrm{card}(\mathrm{ext}_\alpha(a'_i) \cap \mathrm{ext}_\alpha(a'_j))}{\mathrm{card}(\mathrm{ext}_\alpha(a'_i) \cup \mathrm{ext}_\alpha(a'_j))},$$

and

$$\mathrm{RDP}(Y, K) = 1 - \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \frac{\mathrm{card}(\mathrm{ext}_\alpha(a_i) \cap \mathrm{ext}_\alpha(a_j))}{\mathrm{card}(\mathrm{ext}_\alpha(a_i) \cup \mathrm{ext}_\alpha(a_j))}.$$

a'_i and a'_j are the PSOs describing a_i and a_j using only the selected variables Yd.

We know that $ext_\alpha(a) = \{w_l \in \Omega / a(w_l) \geqslant \alpha\}$, and α is a defined threshold. This threshold plays an important role in the extent calculation. The bigger is α, the smaller is the extent; and the smaller is α, the bigger is the extent. β is the threshold tolerance error.

- Cluster discrimination difference (CDD): this criterion shows to the expert the difference between the discrimination of individual clusters and the discrimination reached by the selected variables on the PSO representing these clusters. This criterion is defined as follows:

$$\left|\frac{\mathrm{RDP}(Y_d, K) - \mathrm{RDP}(Y, K)}{\mathrm{RDP}(Y_d, K)}\right| \leqslant \beta, \qquad (25)$$

where

$$\mathrm{CD}(C) = 1 - \sum_{i=1}^{q-1} \sum_{j=i+1}^{q} \frac{\mathrm{card}(\mathrm{ext}_\alpha(c_i) \cap \mathrm{ext}_\alpha(c_j))}{\mathrm{card}(\mathrm{ext}_\alpha(c_i) \cup \mathrm{ext}_\alpha(c_j))}.$$

C: is the set of q clusters built on individuals, $ext(c_i)$: is the set of individuals that belong to the cluster c_i.

If the cluster discrimination difference is bigger than the threshold defined by the expert, it means that the PSOs do not describe well the clusters. Thus, the feature selection will also be affected.

Since the threshold α has a very important role to determine the length of object extents, and it is hard to the expert to provide this threshold without any help, thus, Minset-Plus algorithm provides for the expert some quality indicators, helping him to set correctly the threshold α and then validate the result of feature selection, using RDPV and CDD quality indicators. These threshold quality indicators are calculated before and after the selection. Four indicators have been defined for this purpose:

Let us have m PSOs, and each object is described by n variables.

- The minimum threshold: gives the minimal condition allowing an individual to belong to the extent of an object.

$$AVG\_Min\_\alpha(a_l) = \frac{1}{m} \sum_{i=1}^{m} Min\_\alpha(a_l)), \qquad (26)$$

where $Min\_\alpha(a_l) = \frac{1}{n} \sum_{i=1}^{n} \min_r(p^r_{il})$ and $(p^r_{il}) \neq 0$.

- The maximum threshold: gives the maximal condition allowing an individual to belong to the extent of an object.

$$AVG\_Max\_\alpha(a_l) = \frac{1}{m} \sum_{i=1}^{m} Max\_\alpha(a_l)), \qquad (27)$$

where $Max\_\alpha(a_l) = \frac{1}{n} \sum_{i=1}^{n} \max_r(p^r_{il})$.

- The average threshold: gives the average condition allowing an individual to belong to the extent of an object.

$$AVG\_AVG\_\alpha(a_l) = \frac{1}{m}\sum_{i=1}^{m} AVG\_\alpha(a_l)), \qquad (28)$$

where $AVG\_\alpha(a_l) = \frac{1}{2}Min\_\alpha(a_l) + Max\_\alpha(a_l)$.

- The length of threshold: using this indicator the expert can know which object have larger extent.

$$AVG\_Length\_\alpha(a_l) = \frac{1}{m}\sum_{i=1}^{m} Length\_\alpha(a_l)), \qquad (29)$$

where $Length\_\alpha(a_l) = Max\_\alpha(a_l) - Min\_\alpha(a_l)$.

8.3 Testing

8.3.1 Test on real data

8.3.1.1 IRIS three clusters test

We did a feature selection on the known Fisher dataset, published in the UCI machine learning repository [14]. The data consist of 150 irises, described by four numerical variables: *Sepal Length*, *Petal Length*, *Sepal Width*, and *Petal Width*. The irises are clustered on three clusters, by using the variable species, which describe three categories (Setosa, Versicolor and Virginica, denoted as 1, 2, 3). We used our symbolic object generator program to generate the description of these clusters by using three probabilistic symbolic objects. The cluster discrimination of this dataset is equal to 100%; it means there is no intersection between the clusters (all individuals belong to only one cluster at the same time). Also, the real discrimination power value before selection is equal to 100%, this means that, using all variables, the symbolic objects are completely discriminated.

Using the probabilistic dissimilarity measure, the algorithm Minset-Plus revealed two variables: *Petal Length* with *DP* of 2.84 and *Petal Width* with *DP* of 2.68. But the algorithm selected only *Petal Length* variable, since all the discrimination part of the variable *Petal Width* is already included in the discrimination for the variable *Petal Length* ($ODP(\{Petal\ Width\}, \{Petal\ Length\}, K) = 0$).

We compared our result to the result obtained by the following feature selection methods and algorithms:

- Rough set data analysis (RSDA) method of Browne [15]. RSDA is a non-numeric method of data analysis, enhancing the traditional rough set data analysis by three procedures: significance testing, data filtering and uncertainty measuring.
- Feature selection for clustering algorithm of Dash [16]. Dash et al. use as feature selection criterion a new entropy measure that is low if the individuals have distinct clusters and high otherwise. Using this measure the algorithm should select the most important subset of features, because the result is affected only by the quality of the clustering.
- Feature selection for unsupervised learning algorithm of Dy [17]. Dy and Brodley proposed a feature subset selection method using expectation-maximization (EM), by applying a cross-projection normalization scheme.

To validate and compare the feature selection on Iris data, we created, after feature selection, sets of PSOs describing the variables selected by each algorithm. For calculating the quality criteria, we used the average threshold generated by the algorithm that was 0.139.

The result of this experimentation is shown in Table 2.

All algorithms have found out that the most discriminant variables are: *Petal Length* and *Petal Width*. Minset-Plus algorithm has selected *Petal Length*, the variable selected by all other algorithms.

Table 2 Feature selection on probabilistic symbolic object datasets

Algorithm	Selected variables	RDP after	CDD	RDPV
Browne	*Petal Length*, *Petal Width*	100%	0.000	0.000
Dash	*Petal Length*, *Petal Width*	100%	0.000	0.000
Dy	*Petal Length*, *Petal Width*	100%	0.000	0.000
Minset-Plus	*Petal Length*	100%	0.000	0.000

The real discrimination power using the variables selected by the fourth algorithms (RDP after column) is for all algorithms equal to 100%; this means that all algorithms selected the discriminant variables. Also, the cluster discrimination difference value (CDD column) is equal to 0 for all algorithms; it means that there is no difference between the discrimination of individual clusters and the discrimination reached by the selected variables by all algorithms. And finally, the real discrimination power variation (the column RDPV) is equal to zero for all algorithms, since the real discrimination power before and after the selection did not change, this proofs that the variables selected by all algorithm discriminate well the objects. We can also notice that, Minset-Plus, by selecting only one variable, got the same quality results as other algorithms. This means, we need only *Petal Length* variable to discriminate the PSOs, so we can conclude that Minset-Plus algorithm selected the minimum subset of variables to discriminate the PSOs.

Other datasets: all the datasets used in this experimentation come from the UCI machine learning repository [14]. We used our symbolic object generator program in order to create the symbolic objects that represent the individual clusters of these datasets. Table 3 describes the datasets.

The result of the experimentation is shown in Table 4. We can notice, by looking on CDD criterion "cluster discrimination difference", that all datasets are well represented by the probabilistic symbolic objects. The result of feature selection was good, since the algorithm has reduced the total number of features from 61% to 78%. To assess the quality of feature selection, we calculated the RDPV "real discrimination power variation", and we used the average threshold in the calculation of all datasets' extent. The RDPV varied from 0 to 0.012; this is an indicator of a very good feature selection quality result, and this means that the PSOs described by the selected variables are discriminated like the PSOs before the selection.

We compared the feature selection on probabilistic symbolic objects to the feature selection on Boolean symbolic objects; the result is shown in Table 5.

We can notice that most of the time, the reduction percentage using feature selection on Boolean symbolic objects is greater than the reduction percentage using feature selection on probabilistic symbolic objects. But, based on CDD, we notice that the objects' quality is better when using probabilistic representation. Finally, we notice also that the feature selection quality is better when using probabilistic objects; since when using Boolean objects the RDPV varied from 0 to 0.05, and when using probabilistic objects the RDPV varied from 0 to 0.012; the difference in the best cases reaches the ratio of 2.4.

8.3.2 Simulated test data

In order to create simulated test data, first we generated individuals using our individual generator program; and by using the symbolic object generator program, we created probabilistic symbolic objects for the clusters that we made on these individuals. The individual file has been used in another step to validate the feature selection. We made four categories of experiences: real discrimination power test, percentage of discrimination versus percentage of reduction, time execution, and value criteria versus probabilistic criteria.

- Discrimination power test
  We tested our algorithm on many generated data sets, and we calculated for each test the clusters discrimination difference (CDD) and the real discrimination power variation (RDPV). You can notice in Fig. 5, that the CDDs were good (from 0.3% to 1.7%); this means that the PSOs are representing well the clusters from a discrimination side. The RDPVs are also good (from 1.7% to 2.5%); this means that the selected variables still discriminating well the PSOs. Note that the extents of PSOs have been calculated using the average threshold defined in Eq. (28).
- Complexity test
  The complexity is a critical part to any algorithm, thus we tested the complexity of our feature selection algorithm on PSOs data. For this, we generated ten datasets,

Table 3 Dataset description

DataSet	Attribute number	Individual number	SO number	Type of data
Audiology (Standardized)	69	226	24	Categorical
Dermatology	33	366	6	Categorical, Integer
Heart disease	13	303	5	Categorical, Integer, Real
Cardiotocographic	22	2 126	10	Categorical, Integer, Real

Table 4 Feature selection on probabilistic symbolic object datasets

DataSet	Cluster discrimination	RDP before	% of Reduction	RDP after	CDD	RDPV
Audiology (Standardized)	100%	100%	78.26%	100%	0.000	0.000
Dermatology	100%	100%	78.78%	100%	0.000	0.000
Heart disease	100%	100%	61.53%	98.80%	0.012	0.012
Cardiotocographic	100%	100%	68.18%	100%	0.000	0.000

Table 5 Feature selection on Boolean symbolic object datasets

DataSet	Cluster discrimination	RDP before	% of Reduction	RDP after	CDD	RDPV
Audiology (Standardized)	100%	100%	84.05%	100%	0.000	0.000
Dermatology	100%	98.0%	81.81%	94.10%	0.020	0.039
Heart disease	100%	83.2%	69.23%	77.60%	0168	0.050
Cardiotocographic	100%	100%	68.18%	100%	0.000	0.000

by increasing each time the number of symbolic objects, since the complexity is based on the couples of symbolic objects.

We compared the time execution of Minset-Plus on BSOs, with the time execution on PSOs using probabilistic criterion defined in Eq. (11) and description space probabilistic criterion defined in Eq. (16), named PROBA-VALUE in Fig. 6.

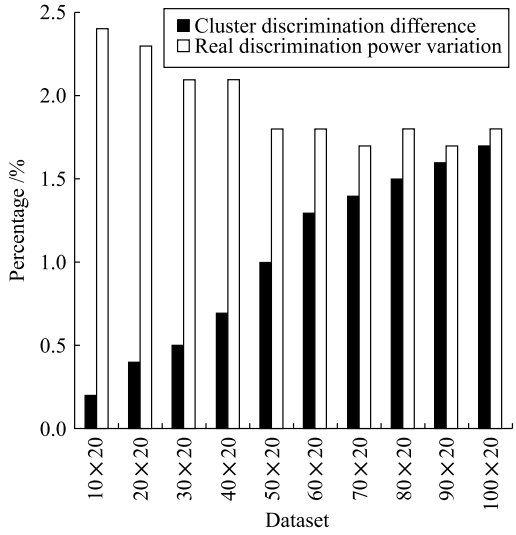

Fig. 5 Discrimination power test

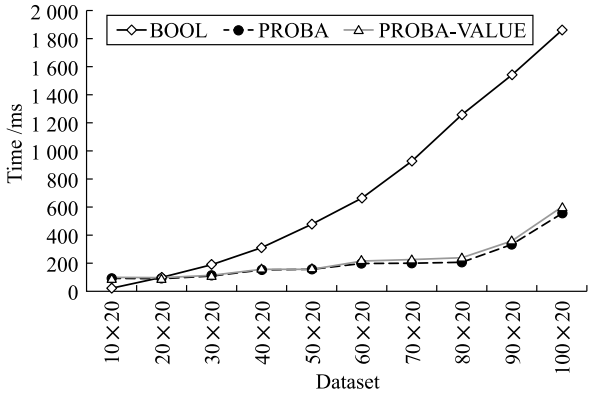

Fig. 6 Complexity test

We can notice in Fig. 6, that the time execution of Minset-Plus on the three categories of test data was good. Since for 100 symbolic objects, we had 550 milliseconds on PSOs using probabilistic criterion, 600 milliseconds on PSOs using value probabilistic criterion, and 2 000 milliseconds on BSOs. Also, you can notice that Minset-Plus algorithm is much faster when it is using PSOs than when using BSOs; this is due to the fact that the criterion used to select features on BSOs is based on the union and intersection between variable values (which is complex computation), but on PSOs the criterion is based on the calculation of value probabilities.

9 Conclusions

In this paper, we presented the feature selection on probabilistic symbolic objects, using the algorithm Minset-Plus. The probabilistic symbolic objects are a rich and complex representation of data, they represent clusters of individuals using multi-valuated attributes (set of values, intervals) with probabilities associated to the values; these probabilities can have different semantics. We have seen in this paper that the choice of strong and suitable feature selection criteria is very important to ensure the quality of the feature selection result, this is why we dedicated a whole section for this purpose, and we used many mathematics formulas to define the feature selection criteria.

Another aspect that we developed in this paper is the complexity; since the data are complex, a big effort has been done in order to improve the algorithm's complexity. Based on the application of some mathematical properties on *ODP* and *DP* functions, and based also on the use of the discrimination matrix, we did an important improvement on the algorithm's complexity.

In order to validate the result of feature selection, we developed an entire system including several programs such as: dataset simulator, symbolic object generator, symbolic object quality, and symbolic feature selection. We proposed a variety of parameters and validation criteria to help the experts to interact with the system and assess the feature selection result quality.

The experimentations done on real and simulated data showed and proved that Minset-Plus algorithm can reduce considerably the number of features without damaging the discrimination between symbolic objects. We also noticed that the feature selection using probabilistic symbolic objects obtains better results than the feature selection on boolean

symbolic objects. The experimentations also showed that the algorithm's time execution on probabilistic symbolic objects is far much better than the time execution on boolean symbolic objects.

Based on all these experimentation results, we can conclude that it is preferable to use feature selection on probabilistic symbolic objects instead of using feature selection on boolean symbolic objects. Because the probabilistic symbolic objects are rich objects, we obtain better results of feature selection, and we gain a lot in the complexity of the algorithm.

Acknowledgements The author would like to thank King Saud University, the College of Computer and Information Sciences, and the Research Center for their sponsorship.

References

1. Billard L, Diday E. Symbolic data analysis. John Wiley & Sons, Ltd., 2006
2. Diday E, Esposito F. An introduction to symbolic data analysis and the SODAS software. Intelligent Data Analysis, 2003, 7(6): 583–601
3. Diday E. Probabilist, possibilist and belief objects for knowledge analysis. Annals of Operations Research, 1995, 55(2): 227–276
4. Ziani D. Sélection de variables sur un ensemble d'objets symboliques: traitement des dépendances entre variables. Paris: University of Paris Dauphine, Dissertation for the Doctoral Degree 1996 (in French)
5. Lebbe J. Représentation des concepts en biologie et en médecine. Dissertation for the Doctoral Degree, 1991 (in French)
6. Bock H H, Diday E. Analysis of symbolic data: exploratory methods for extracting statistical information from complex data. Springer, 2000, 389–391
7. Ziani D. Feature selection on Boolean symbolic objects. International Journal of Computer Science & Information Technology, 2013, 5(6): 1
8. Malerba D, Esposito F, Monopoli M. Comparing dissimilarity measures for probabilistic symbolic objects. Data mining III, Series Management Information Systems, 2002, 6: 31–40
9. Rached Z, Alajaji F, Campbell L L. Rényi's divergence and entropy rates for finite alphabet Markov sources. IEEE Transactions on Information Theory, 2001, 47(4): 1553–1561
10. Kullback S, Leibler R A. On information and sufficiency. Annals of Mathematical Statistics, 1951, 22(1): 79–86
11. Beirlant J, Devroye L, Györfi L, Vajda I. Large deviations of divergence measures on partitions. Journal of Statistical Planning and Inference, 2001, 93(1): 1–16
12. Ziani D, Khalil Z, Vignes R. Recherche de sous-ensembles minimaux de variables à partir d'objets symboliques. In: Proceedings of the 5th èmes Journées "Symbolique-Numérique". 1994, 794–799 (in French)
13. Esposito F, Malerba D, Appice A. Dissimilarity and matching. Symbolic Data Analysis and the SODAS Software, 2008, 61–66
14. Frank A, Asuncion A. Uci machine learning repository [http://archive. ics. uci. edu/ml]. irvine, ca: University of california. School of Information and Computer Science, 2010, 213
15. Browne C, Düntsch I, Gediga G. Iris revisited: a comparison of discriminant and enhanced rough set data analysis. Rough Sets in Knowledge Discovery 2, 1998, 19: 345–368
16. Dash M, Choi K, Scheuermann P, Liu H. Feature selection for clustering — a filter solution. In: Proceedings of the 2002 IEEE International Conference on Data Mining. 2002, 115–122
17. Dy J G, Brodley C E. Feature selection for unsupervised learning. The Journal of Machine Learning Research, 2004, 5: 845–889

Djamal Ziani is an assistant professor in Computer Sciences and Information Systems College, King Saud University Saudi Arabia from 2009 until now. He is a researcher in ERP and in data management group of CCIS, King Saud University. He received his MS degree in computer sciences from University of Valenciennes, France in 1992, and his PhD degree in computer sciences from University of Paris Dauphine, France in 1996. Researcher in CLOREC project, INRIA Rocquencourt, France from 1992 to 1996. Post Doc in Department of Computer Sciences and Operational Research of University of Montreal, Canada from 1997 to 1998. Consultant and project manager in many companies in Canada from 1998 to 2009.