Intelligent Data Analysis 10 (2006) 301–324
IOS Press


# Classification of symbolic objects: A lazy learning approach

Annalisa Appice, Claudia D'Amato, Floriana Esposito and Donato Malerba
*Dipartimento di Informatica, Università degli Studi, via Orabona, 4, 70125 Bari, Italy*
*E-mail: {appice, claudia.damato, esposito, malerba}@di.uniba.it*

**Abstract.** Symbolic data analysis aims at generalizing some standard statistical data mining methods, such as those developed for classification tasks, to the case of symbolic objects (SOs). These objects synthesize information concerning a group of individuals of a population, eventually stored in a relational database, and ensure confidentiality of original data. Classifying SOs is an important task in symbolic data analysis. In this paper a lazy-learning approach that extends a traditional distance weighted k-Nearest Neighbor classification algorithm to SOs, is presented. The proposed method has been implemented in the system SO-NN (Symbolic Objects Nearest Neighbor) and evaluated on symbolic datasets.

## 1. Introduction

National Statistics Institutes (NSIs), as well as government departments and local authorities, regularly collect a large amount of official data through censuses, statistical surveys and administrative records. Generally, official data are processed by NSIs to produce official statistics, such as inflation rate and gross national product (GNP), which are used to inform the general public and to support governments in their functions. More recently, there has been an increasing interest in using official data also to support the decision-making of either private companies (e.g., by keeping labor/product/capital market analyzers up to date) or single individuals (e.g., by informing them on specific educational and occupational choices) [33]. This different use of official data frequently requires more detailed analyses than are presently published by statistical organizations, as well as specific data mining activities that can reveal new patterns buried in official datasets. However, the idea of exploring a database with the objective of finding unexpected patterns is not familiar to official statisticians who have to answer precise questions and make forecasts. In NSIs, statistical analyses are done generally if they can be repeated in a production framework [40].

Since the main task of NSIs remains official data production, data analysis is often performed by academic institutions or independent research institutions. This leads to increasing pressure on NSIs and other statistical organizations to provide detailed data (*micro-data*) on a wide range of topics. However, there are problems in providing micro-datasets to researchers – the main one being the confidential nature of the data themselves. Indeed, general Data Protection laws prohibit NSIs from releasing individual responses of censuses and surveys to any other government agency or to any individual or business.

A solution to confidentiality problems is that of creating datasets for access by researchers through the *aggregation* of micro-data. An appropriate choice of the aggregation unit may provide researchers with datasets still useful for the intended data mining tasks but whose risk of identification of a record or of disclosure is acceptably small. For instance, UK census data are aggregated by Enumeration Districts

Table 1
Symbolic data table describing groups of residents per ED (Enumeration District)

| ED | # Inhabitants | Annual income (pounds) | Communal Establishments | Ethnicity % (W,B,A,O) |
|---|---|---|---|---|
| 03BS004 | 555 | [1500–25000] | {Kindergarten, Primary School, Guesthouse} | (60, 15, 15, 10) |
| 03BS005 | 307 | [20000–115000] | {College, Hotel, Hospital} | (85, 5, 5, 5) |
| 03BS006 | 805 | [0–12000] | {Prison, Hospital} | (25, 45, 25, 5) |

(ED) before being published. Data aggregated at an ED level do not allow data analysts to identify an individual or a single business establishment, but are detailed enough to investigate social problems, such as transportation [30] and accessibility [3].

In classical statistics, aggregation refers to the computation of some descriptive statistics, such as mode, mean or standard deviation, each of which returns a single (continuous or categorical) value. An alternative approach that equally prevents the disclosure of confidential micro-data is that of making only *generalizations of groups of individuals* available to external agencies and institutes. This approach is at the base of *symbolic data analysis* (SDA) [5], which is mainly concerned with the analysis of *second-order objects*, that is, generalizations of groups of individuals or classes, rather than single individuals (first-order objects).

In SDA, generalizations are typically represented by means of *set-valued* and *modal* variables [5]. A variable $X$ defined for all elements of a set $E$ is termed *set-valued* with domain $\mathcal{X}$ if it takes its values in $\boldsymbol{P}(\mathcal{X}) = \{U|U \subseteq \mathcal{X}\}$, that is, the power set of $\mathcal{X}$. When $X(k)$ is finite for each $k$, then $X$ is called *multi-valued*. A single-valued variable is a special case of set-valued variables for which $|X(k)|$ = 1 for each $k$. When an order relation $<$ is defined on $\mathcal{X}$, then the value returned by a set-valued variable can be expressed by an interval $[\alpha, \beta]$, and $X$ is termed an *interval* variable. A modal variable $X$ is a multi-valued variable with *weights*, such that $X(k)$ describes both multi-valued data $U(k)$ and associated weights $\pi(k)$, that is, $X(k) = (U(k), \pi(k))$. Notice that single-valued variables can also be considered special cases of modal variables whose weight equals either to 1 or 0.

Typically, each generalization is represented by a set of $m$ variables $X_i$. Moreover, generalizations of different groups of individuals from the same population are described by the same set of symbolic variables. This leads to data tables, named *symbolic data tables*, more complex than those typically used in classical statistics. Rows of the table correspond to distinct generalizations (or *symbolic descriptions*), while columns of a symbolic data table are called *symbolic variables*. Each item at the intersection of a row and a column does not necessarily contain, as usual, just a single continuous or categorical value, but several values with possibly an associated modality (frequency, probability or weight). An example of a symbolic data table describing groups of residents per ED (Enumeration District) is given in Table 1. In this case, the first symbolic variable is single-valued and reports the number of residents. The second symbolic variable is interval-valued and reports the range of values of annual income of the residents. The third symbolic variable is multi-valued and lists the communal establishments in the ED. The four modalities of the fourth symbolic variable refer to the percentages of White (W), Black (B), Asian (A) and other (O) residents in the ED.

Symbolic data tables can be generated from relational databases storing original micro-data, by applying both generalization and specialization operators in order to obtain a homogeneous description of group [41].

The symbolic description corresponding to a row of a symbolic data table describes a class of individuals, which are in turn the partial or complete extent of a given concept. Starting with this description, a

*symbolic object* (SO) models the underlying concept and provides a way to find at least the individuals of this class. Indeed, a symbolic object is formally defined in [5] as a triple $s = (a, R, d)$ where $R$ is a relation between descriptions (e.g., $R \in \{=, \equiv, \leqslant, \subseteq\}$ or $R$ is an implication, a kind of matching), $d$ is a description and $a$ is a mapping defined by a set of individuals $\Omega$ in a set $L$ (e.g., $L = \{\text{true, false}\}$ or $L = [0, 1]$) such that $a$ depends on $R$ and $d$.

The main goal of SDA is that of investigating new theoretically sound techniques by generalizing some standard statistical data mining methods, such as those developed for classification or clustering tasks to the case of symbolic objects. Such techniques usually concern classes of symbolic objects, where $R$ is fixed, "$d$" varies among a finite set of coherent descriptions and "$a$" is such that: $a(w) = [y(w)\ R\ d]$, which is by definition the result of the comparison of the description of the individual $w$ to $d$.

Many techniques for both the construction of symbolic objects from records of individuals and the analysis of symbolic objects have already been implemented in an integrated software environment SODAS [14].

In this paper, we investigate the classification of symbolic objects by means of a classification method named SO-NN (Symbolic Objects Nearest Neighbor) that extends the traditional distance weighted $k$-Nearest Neighbor ($k$-NN) classification algorithm to SOs. This work is the first attempt at extending a lazy learning method to deal with symbolic data tables. Lazy learning methods estimate the target function locally and differently for each instance to be classified and are considered quite effective when the target function is complex. Moreover, the main disadvantage of lazy learning, namely computational complexity, is mitigated by the lower dimensionality of a symbolic data table with respect to the original set of individuals.

The paper is organized as follows. In the next section we present related work on classification in SDA and we motivate the importance of a lazy learning approach. The definition of $k$-NN, and consequently of its extension to symbolic objects, is based on the notion of dissimilarity. For this reason, in Section 3 we introduce some dissimilarity measures defined for symbolic objects. The core of the SO-NN method is described in Section 4. Finally, some experimental results are reported in Section 5 and some conclusions are drawn.

## 2. Related work and motivation

The problem of classifying symbolic objects has been approached in many ways. Ciampi et al. [10] introduced a generalization of binary decision trees [7] to predict the class membership of a SO. Training observations are described by one or more explanatory symbolic variables $X_i$ and one single-valued symbolic target variable $Y$. The algorithm is based on a divide-and-conquer strategy starting with a root node that is associated with the entire set of training symbolic data. At each step, training data is recursively split until the tree is sufficiently accurate. The best split is chosen according to a splitting rule that consists in maximizing the Generalized Information Measure (GInf) over the set of candidate splits. Candidate splits are defined as in the classical case, that is, where each explanatory variable provides a single binary question (e.g. "age in [0, 18]" versus "age in [18, 130]"). In [9], the authors define fuzzy splits, which are probabilistic binary questions built on the basis of a logistic transformation.

Bravo [6] has proposed an alternative splitting criterion that maximizes the *Extended Information Content* (EIC) when building a Strata Decision Tree, which is a tree-based classifier, from a set of *individuals* $E$ partitioned into several strata. Each stratum corresponds to a SO since it is described by symbolic variables (categorical single-valued or modal, but not interval), which can be related by

hierarchical dependencies. EIC measure of a candidate split properly estimates internal entropy over strata $E$ with respect to both the left and right child of the split node.

The splitting criterion adopted for the construction of a Structural Bayesian Decision Tree (SBTREE) [37] minimizes the misclassification error when both the left and right child are labeled with the bayesian classification rule computed according to kernel density estimation. Similarly to the other tree-based classifiers for SOs, the result is a single class. However, the SBTREE induction method has been designed to deal only with interval symbolic variables.

Rossi and Conan-Guez [38] have generalized Multi-Layered Perceptrons [32] to work with interval-valued data. The output is an $l$-dimensional vector $(y_1, \ldots, y_l)$ of modalities, where $l$ is the number of distinct classes. The single-class output can be obtained by returning the class $c_i$ such that $y_j = \max(y_1, \ldots, y_l)$. Despite the robustness and flexibility of Multi-Layered Perceptrons, they appear clearly inadequate in problems where interpretability is a key factor, due to the difficulty of users in interpreting their predictions.

Finally, Lauro et al. [25] have proposed an extension of the Factorial Discriminant Analysis (FDA) to SOs. In FDA the original variables are linearly combined into a new set of features that describe the observations, and the classification is based on the computation of the proximity of the observation to the separating hyperplane. The extension of FDA to SOs is based on an initial conversion of SOs into observations described by a set of continuous explanatory variables, the application of standard FDA to such observations, and the final conversion of geometrical classification rules into SOs.

All these works refer to classification methods that eagerly learn a general explicit description (e.g. decision tree, neural network or rule set) of a discrete-valued target function (class label) when training data is provided. This is in contrast with *lazy learning* methods, which simply store training data and *delay* learning until a new instance must be classified. Each time a new test (query) instance is encountered, its relationship to the previously stored objects is examined in order to assign a target function value for the new instance. A key advantage of lazy learning methods is that instead of estimating the target function at once for the entire instance space, they can estimate it locally and differently for each instance to be classified [32]. This has significant advantages when either training data is noisy or the target function is very complex, but can still be described by a collection of less complex local approximation [1]. Indeed, since the test instance to be classified is known during the processing of training data, training a query-specific local model is possible with lazy learning. This means constructing only a local approximation of the target function that applies in the neighborhood of the new test instance and never constructing an approximation designed to perform well over the entire training space.

Obviously, this highly adaptive behavior may cause high cost of classifying new objects. This is due to the fact that nearly all computation takes place at classification time rather than when training objects are first encountered. However, this problem is less relevant in SDA, since the number of SOs generated from micro-data is generally much less than that of the original individuals.

A lazy learning method adopted in classification problems is the $k$-Nearest Neighbour ($k$-NN) algorithm [42] that approximates a discrete-valued variable by assigning the discrete-valued target function of the test instance $q$ with the most common value among the $k$ nearest training objects. Neighbors are here determined according to a distance measure $d$. An obvious refinement of the k-NN algorithm is to weight the contribution of each of the $k$ neighbors according to their distance from the query instance $q$, giving greater weight to closer neighbors [32].

In this work, we propose an extension of distance-weighted $k$-NN for SOs described by both a single-valued target variable $Y$ and $m$ explanatory variables (either modal or set valued). More precisely, given a set of SOs to be classified, the distance weighted k-NN method we propose produces, as output, a

matrix $P$ whose rows represent descriptions of SOs and whose columns represent all known classes. Each element $p_{ij}$ of the matrix $P$ is an estimation of the probability that $SO_i$ belongs to the class $j$ and $\sum_j p_{ij} = 1$. Estimate $p_{ij}$ is computed according to the *neighborhood* of $SO_i$ that is the $k$ training objects closest to $SO_i$ with respect to a dissimilarity measure $d$.

The basic k-NN method assumes that all training cases correspond to points in the $m$-dimensional space $\Re^m$ and the nearest neighbors of the instance to classify are defined in terms of the standard Euclidean distance. However, in our problem formulation, training observations are symbolic objects, which cannot be associated to points of $\Re^m$. Therefore, it is necessary to resort to a different notion of dissimilarity measure that applies to SOs. The next section surveys some of the dissimilarity measures defined on SOs implemented in the SODAS software.[1] A more detailed description is available in [15, 28,29].

## 3. Dissimilarity measures for symbolic objects

Henceforth, the term *dissimilarity measure* $d$ on a set of objects $O$ refers to a real valued function on $O \times O$ such that: $d_a^* = d(a,a) \leqslant d(a,b) = d(b,a) < \infty$ for all $a$, $b \in O$. Generally, $d_a^* = d^*$ for each object $a$ in $O$, and more specifically, $d^* = 0$. Several dissimilarity measures have been proposed for restricted classes of symbolic objects, namely *Boolean Symbolic Objects* (BSOs) and *Probabilistic Symbolic Objects* (PSOs). The former are described by set-valued variables only, while the latter are described by modal variables with a relative frequency distribution associated to each one.

### 3.1. Dissimilarity measure for BSOs

Let $a$ and $b$ be two BSOs described by $m$ symbolic variables $X_i$ with domain $\mathcal{X}_i$. Let $A_i$ ($B_i$) be the set of values (subset of $\mathcal{X}_i$) taken by $X_i$ in $a$ ($b$). A class of dissimilarity measure between $a$ and $b$ is defined by aggregating dissimilarity values computed independently at the level of single variables $X_i$ (*componentwise dissimilarities*). A classical aggregation function is the Minkowski metric (or $L_q$ distance) defined on $\Re^m$: Another class of dissimilarity measures is based on the notion of *description potential* $\pi(a)$ of a BSO $a$, which corresponds to the *volume* of the Cartesian product $A_1 \times A_2 \times \ldots \times A_p$. For this class of measures, no componentwise decomposition is necessary, so that no function is required to aggregate dissimilarities computed independently for each variable.

Dissimilarity measures considered in this study are reported in Table 2 together with their short identifier used in the SODAS software. They are:

- Gowda and Diday's dissimilarity measure (U1) [19],
- Ichino and Yaguchi's first formulation of a dissimilarity measure (U2) [20],
- Ichino and Yaguchi's dissimilarity measure normalized (U3) [20],
- Ichino and Yaguchi's normalized and weighted dissimilarity measure (U4) [20],

[1] SODAS (Symbolic Official Data Analysis System) was a three-year ESPRIT project concluded in November 1999. The SODAS software can be downloaded from: http://www.ceremade.dauphine.fr/~touati/sodas-pagegarde.htm. The three-year IST project ASSO (Analysis System of Symbolic Official Data) (http://www.info.fundp.ac.be/asso/) continued SODAS and produced an improved version of the SODAS software.

Table 2
Dissimilarity measures defined for BSOs

| Name | Componentwise dissimilarity measure | Objectwise dissimilarity measure |
|---|---|---|
| U1 | $D^{(i)}(A_i, B_i) = D_\pi(A_i, B_i) + D_s(A_i, B_i) + D_c(A_i, B_i)$ where $D_\pi(A_j, B_j)$ is due to *position*, $D_s(A_j, B_j)$ to *spanning* and $D_c(A_j, B_j)$ to *content*. | $\sum_{i=1}^{m} D^{(i)}(A_i, B_i)$ |
| U2 | $\phi(A_i, B_i) = \lvert A_i \oplus B_i\rvert - \lvert A_i \otimes B_i\rvert + \gamma(2\lvert A_i\rvert \otimes \lvert B_i\rvert - \lvert A_i\rvert - \lvert B_i\rvert)$ where *meet* ($\otimes$) and *join* ($\oplus$) are two Cartesian operators. | $\sqrt[q]{\sum_{i=1}^{m} [\phi(A_i, B_i)]^q}$ |
| U3 | $\psi(A_i, B_i) = \frac{\phi(A_i, B_i)}{\lvert X_i\rvert}$ | $\sqrt[q]{\sum_{i=1}^{m} [\psi(A_i, B_i)]^q}$ |
| U4 | $\psi(A_i, B_i) = \frac{\phi(A_i, B_i)}{\lvert X_i\rvert}$ | $\sqrt[q]{\sum_{i=1}^{m} w_i\,[\psi(A_i, B_i)]^q}$ |
| C1 | $D_1(A_i, B_i) = 1 - \alpha/(\alpha + \beta + \chi)$<br>$D_2(A_i, B_i) = 1 - 2\alpha/(2\alpha + \beta + \chi)$<br>$D_3(A_i, B_i) = 1 - \alpha/(\alpha + 2\beta + 2\chi)$<br>$D_4(A_i, B_i) = 1 - \frac{1}{2}(\frac{\alpha}{\alpha+\beta} + \frac{\alpha}{\alpha+\chi})$<br>$D_5(A_i, B_i) = 1 - \alpha/\sqrt{(\alpha+\beta)(\alpha+\chi)}$ | $\sqrt[q]{\frac{\sum_{i=1}^{m} [w_i D_r(A_i, B_i)]^q}{\sum_{i=1}^{m} \delta(i)}}$<br>where $\delta$(i) is the indicator function |
| SO1 | with: $\chi = \mu(c(A_i) \cap B_i)$<br>$\alpha = \mu(A_i \cap B_i)$;<br>$\beta = \mu(A_i \cap c(B_i))$<br>For each subset $U \subseteq \mathcal{X}$, $\mu(U) = \lvert U\rvert$ if $X$ is a set-valued variable, while $\mu(U) = \lvert a - b\rvert$ if $X$ is an interval variable with $U = [a - b]$. $c(U)$ is the complementary set of $U$ in the domain $\mathcal{X}$. | $\sqrt[q]{\sum_{i=1}^{m} [w_i D_r(A_i, B_i)]^q}$ |
| SO2 | $\psi'(A_i, B_i) = \frac{\phi(A_i, B_i)}{\mu(A_i \oplus B_i)}$ | $\sqrt[q]{\sum_{i=1}^{m} \frac{1}{m} [\psi'(A_i, B_i)]^q}$ |
| SO3 | none | $\pi(a \oplus b) - \pi(a \otimes b) + \gamma(2\pi(a \otimes b) - \pi(a) - \pi(b))$ where *meet* ($\otimes$) and *join* ($\oplus$) are Cartesian operators defined on BSO. |
| SO4 | none | $\frac{\pi(a\oplus b) - \pi(a\otimes b) + \gamma(2\pi(a\otimes b) - \pi(a) - \pi(b))}{\pi(a^E)}$ where $o$ is the BSO obtained by associating the domain set $\mathcal{X}_i$ to the symbolic variable $X_i$ |
| SO5 | none | $\frac{\pi(a\oplus b) - \pi(a\otimes b) + \gamma(2\pi(a\otimes b) - \pi(a) - \pi(b))}{\pi(a\oplus b)}$ |
| SO6 | none | $1 -$ [FlexMatch(a,b) + FlexMatch(b,a)]/2 |

– De Carvalho's normalized dissimilarity measure for constrained[2] BSOs (C1) [13],
– De Carvalho's dissimilarity measure (SO1) [12],

[2]The term *constrained BSO* refers to the fact that some dependencies are defined between two symbolic variables $X_i$ and $X_j$, namely *hierarchical dependencies* which establish conditions for some variables being not measurable (not-applicable values), or *logical dependencies* which establish the set of possible values for a variable $X_i$ conditioned by the set of values taken by the variable $X_j$. An investigation of the effect of constraints on the computation of dissimilarity measures is out of the scope of this paper, nevertheless it is always possible to apply the measures defined for constrained BSOs to unconstrained BSOs.

– De Carvalho's extension of Ichino and Yaguchi's dissimilarity (SO2) [12],
– De Carvalho's first dissimilarity measure based on description potential (SO3) [13],
– De Carvalho's second dissimilarity measure based on description potential (SO4) [13],
– De Carvalho's normalized dissimilarity measure based on description potential (SO5) [13],
– dissimilarity measure based on flexible matching among BSOs (SO6).

The last measure (SO6) differs from the others, since its definition is based on the notion of flexible matching [16], which is an asymmetric measure of similarity. The dissimilarity measure is obtained by means of a symmetrization method that is common to measures defined for PSOs.

### *3.2. Dissimilarity measure for PSOs*

Let $a$ and $b$ be two PSOs of a symbolic data table and $X$ a multi-valued modal variable describing the two PSOs. The sets of probabilistically weighted values taken by $\mathcal{X}$ in $a$ and $b$ define two discrete probability distributions $P$ and $Q$, whose comparison allows us to assess the dissimilarity between $a$ and $b$ on the basis of $X$ only. For instance, we may have: $P =$ (red:$0.\bar{3}$, white:$0.\bar{3}$, black:$0.\bar{3}$) and $Q =$ (red:0.1, white:0.2, black:0.7) when the domain of $X$ is $=\{$*red*, *white*, *black*$\}$. Therefore, the dissimilarity between two PSOs described by $m$ symbolic variables can be obtained by aggregating the dissimilarities defined on as many pairs of discrete probability distributions (componentwise dissimilarities). Before explaining how to aggregate them, some comparison functions $m(P, Q)$ for probability distributions, are introduced.

Most of the comparison functions for probability distributions belong to the large family of "convex likelihood-ratio expectations" introduced by both Csiszàr [11] and Ali and Silvey [2]. Some well-known exemplars of this family are:

– The *KL-divergence*, which is a measure of the difference between two probability distributions [24]. It is defined as $m_{KL}(P, Q) := \Sigma_{x\in\mathcal{X}} q(x)\log(q(x)/p(x))$ and measures to which extent the distribution$P$ is an approximation of the distribution $Q$. It is asymmetric, that is $m_{KL}(P, Q) \neq m_{KL}(Q, P)$ in general, and it is not defined when $p(x) = 0$. The KL-divergence is generally greater than zero, and it is zero only when the two probability distributions are equal.
– The $\chi^2$*-divergence* defined as $m^2_\chi(P, Q) := \Sigma_{x\in\mathcal{X}}|p(x) - q(x)|^2/p(x)$, is strictly topologically stronger then KL-divergence since the inequality $m_{KL}(P, Q) \leqslant m^2_\chi(P, Q)$ holds, i.e. the convergence in $\chi^2$-divergence implies convergence in KL-divergence, but the converse is not true [4]. Similarly to the KL-divergence, it is asymmetric and is not defined when $p(x) = 0$.
– The Hellinger Coefficient is a similarity-like measure given by $m^{(s)}(P, Q) := \Sigma_{x\in X} q^s(x)p^{1-s}(x)$, where $s$ is a positive exponent with $0 < s < 1$. From this similarity-like measure *Chernoff's distance of the order* $s$ is derived as follows: $m^{(s)}_C(P, Q) := -\log m^{(s)}(P, Q)$. This distance diverges only when the two distributions have zero overlap, that is, the intersection of their support is empty [21].
– *Rènyi's divergence* (or *information gain*) of order $s$ between two probability distributions $P$ and $Q$ is given by $m^{(s)}_R(P, Q) := -\log m^{(s)}(P, Q)/(s - 1)$. It is noteworthy that, as $s \to 1$, Rènyi's divergence approaches the KL-divergence [36].
– The *variation distance*, given by $m_1(P, Q) := \Sigma_{x\in\mathcal{X}}|p(x) - q(x)|$, is also known as *Manhattan distance* for the probability functions $p(x)$ and $q(x)$ and coincides with the *Hamming distance* when all features are binary. Similarly, it is possible to use *Minkowski's* $L_2$ (or *Euclidean*) *distance* given by $m_2(P, Q) := \Sigma_{x\in\mathcal{X}}|p(x) - q(x)|^2$ and, more in general, the Minkowski's $L_p$ distance

with $p \in \{1, 2, 3, \ldots\}$. All measures $m_p(P, Q)$ satisfy the metric properties and in particular the symmetry property. The main difference between $m_1$ and $m_p$, $p > 1$, is that the former does not amplify the effect of single large differences (outliers). This property can be important when the distributions $P$ and $Q$ are estimated from noisy data.

– The *K-divergence* is given by $m_K(P, Q) := \Sigma_{x \in \mathcal{X}} q(x) \log(q(x)/(\frac{1}{2}p(x) + \frac{1}{c}q(x)))$ [26], which is an asymmetric measure. It has been proved that K-divergence is upper bounded by the variation distance $m_1(P, Q)$: $m_K(P, Q) \leqslant m_1(P, Q) \leqslant 2$.

Some of the divergence coefficients defined above do not obey all the fundamental axioms that dissimilarities must satisfy. For instance, the KL-divergence does not satisfy the symmetric property. Nevertheless, a symmetrized version, termed *J-coefficient* (or *J-divergence*), can be defined as follows $J(P, Q) := m_{KL}(P, Q) + m_{KL}(Q, P)$. Alternatively, many authors have defined the *J-divergence* as the average rather than the sum $J(P, Q) := (m_{KL}(P, Q) + m_{KL}(Q, P))/2$. Generally speaking, for any (*possible*) non-symmetric divergence coefficient $m$ there exists a symmetrized version $\underline{m}(P, Q) = m(Q, P) + m(P, Q)$ which fulfils all axioms for a dissimilarity measure, but typically not the triangle inequality. Obviously, in the case of Minkowski's $L_p$ coefficient, which satisfies the properties of a dissimilarity measure and, more precisely of a metric (triangular inequality), no symmetrization is required.

Given these componentwise dissimilarity measures, we can define the dissimilarity measure between two PSOs $a$ and $b$ by aggregation through the generalized and weighted Minkowski's metric:

$$d_p(a, b) = \sqrt[p]{\sum_{i=1}^{m} [c_i m\,(A_i, B_i)]^p}$$

where, $\forall k \in \{1, \ldots, n\}$, $c_k > 0$ are weights with $\Sigma_{k=1\ldots m} c_k = 1$ and $m(A_i, B_i)$ is either Minkowski $L_p$ distance (LP) or a symmetrized version of J-coefficient (J), $\chi^2$ – divergence (CHI2), Rènyi's distance (REN), and Chernoff's distance (CHER). These are all variants of the dissimilarity measure denoted as P1 in the SODAS software.

Alternatively, the dissimilarity coefficients can be aggregated through the product. Therefore, by adopting appropriate precautions and considering only Minkowski's $L_p$ distance, we obtain the following normalized dissimilarity measure between PSOs:

$$d'_p(a, b) = 1 - \frac{\prod_{i=1}^{m} \left( \sqrt[p]{2} - \sqrt[p]{\sum_{y_i} |p(x_i) - q(x_i)|^p} \right)}{\left(\sqrt[p]{2}\right)^m} = 1 - \frac{\prod_{i=1}^{m} \left( \sqrt[p]{2} - \sqrt[p]{L_p} \right)}{\left(\sqrt[p]{2}\right)^m}$$

where each $x_i$ corresponds to a value of the $i$-th variable domain.

Note that this dissimilarity measure, denoted as P2 in the SODAS software, is symmetric and normalized in [0, 1]. Obviously $d'_p(a, b) = 0$ if $a$ and $b$ are identical and $d'_p(a, b) = 1$ if the two objects are completely different.

Finally, the dissimilarity measure between two PSOs $a$ and $b$, can be computed by estimating both the matching degree between $a$ and $b$ and vice-versa. The measure denoted as P3 in the SODAS software, extends the measure SO6 defined for BSOs.

A summary of the three dissimilarity measures defined on PSOs is reported in Table 3.

Table 3
Dissimilarity measures defined for PSOs

| Name | Componentwise dissimilarity measure | Objectwise dissimilarity measure |
|---|---|---|
| P1 | Either $m_p(P,Q)$ or a symmetrized version of $m_{KL}(P,Q), m^2_\chi(P,Q)$, $m^{(s)}_C(P,Q)$, $m^{(s)}_R(P,Q)$ | $\sqrt[p]{\sum_{i=1}^{m}[c_i m(A_i,B_i)]^p}$ |
| P2 | $m_p(P,Q)$ | $1-\frac{\prod_{i=1}^{m}\left(\sqrt[p]{2}-\sqrt[p]{m_p(A_i,B_i)}\right)}{\left(\sqrt[p]{2}\right)^m}$ |
| P3 | none | $1-$ [FlexMatch(a,b) + FlexMatch(b,a)]/2 |

## 4. Symbolic objects nearest neighbour classification

In a quite general formulation, the *classification problem* in SDA can be defined as follows:

*Given* a set $O$ of $n$ training cases $(\boldsymbol{x}, y) \in \boldsymbol{X} \times Y$, such that $\boldsymbol{x}$ is a vector denoting a symbolic description on the space of symbolic variables $\boldsymbol{X} = B_1 \times \ldots \times B_m$, while $y$ represents the category value (class) that belongs to a finite set $\mathcal{Y} = \{c_1, \ldots, c_l\}$; the goal is to *predict* the value of the class variable $Y$ for a SO (testing case) $q$ described by the same symbolic variables $B_1, \ldots, B_m$ as the training data.

A method that solves the problem stated above is SO-NN that extends the distance weighted $k$-NN classifier towards SDA. The $k$-NN algorithm is a simple, well-known lazy learning technique that requires only a dissimilarity measure $d$, a positive integer $k$ and a set of labeled training examples $O$, named prototypes. A new (unlabelled) example $q$ is assigned to the label most frequently represented among its $k$ nearest neighbors, that is, the set of $k$ prototypes which are most similar to $q$ with respect to $d$. This means that those training examples very different from $q$, are completely ignored. In the standard $k$-NN method, each training example in the neighborhood contributes to the classification with the same "weight", independently of how dissimilar it is from the test example $q$. A natural extension of k-NN is to weight the contribution of each neighbor on the basis of its dissimilarity to test case $q$, giving greater weight to more similar neighbors. This extension is known as distance-weighted k-NN [32].

The main difference between classical distance-weighted $k$-NN and lazy classification performed by SO-NN is the dissimilarity $d(q,o)$ between $q$ and its potential neighbors $o \in O$. Indeed, in classical $k$-NN, training examples are real-valued in an m-dimensional Euclidean space, while in SDA training examples are SOs that cannot be simply associated to points of $\Re^m$.

In Section 3 several dissimilarity measures have been reported for BSOs and PSOs. However, in general a SO can be described by both modal and non-modal variables. For mixed SOs, no new measures are defined, since it is possible to separate the Boolean from the Probabilistic part of the SO, and then to compute two dissimilarity measures separately. The combination of the two measures can be either additive or multiplicative, although special care must be taken when a choice is made.

A top-level description of SO-NN is reported in Fig. 1. SO-NN returns a modal variable $Y'$ rather than a single-valued class label. Modalities correspond to the probabilities of class membership for each of the $l$ classes. Details on the estimation of the probabilities are reported in the next subsections.

### 4.1. Estimating class probabilities

Let us consider a test SO $q$ to be classified according to the $k$-sized neighborhood $O_k(q) = \{o_1, \ldots, o_k\}$ determined on the training data $O$ by considering the $k$ SOs closest to $q$ with respect to a dissimilarity measure $d$. SO-NN returns the value $y'$ of an $l$-dimensional class probability vector $y' = (y_1(q), \ldots, y_l(q))$

**Training step**

Store the set $O$ of $n$ training SOs described by both $m$ symbolic variables $X_i$ (predictor variables) and single-valued target variable $Y$ (class variable) with range in the finite set $\mathcal{Y}=\{c_1,...,c_l\}$;

**Classification step**

Given a test SO $q = (x_1(q), \ldots, x_m(q))$ to be classified

Determine $O_k(q)$, that is, the set of $k$ SOs in $O$ closest to $q$ according to a dissimilarity measure $d$

Compute $y' = (y_1,\ldots,y_l)$ that is the vector of probabilities of class labels $(c_1, \ldots, c_l)$ on $O_k(q)$

return $(x_1(q), \ldots, x_m(q), y')$

Fig. 1. SO-NN algorithm for approximating unknown value of single-value categorical.

associated to $q$, such that $l$ is the number of distinct classes on $Y$. Each $y_i = P(Y(q) = c_1)$ is estimated on $O_k(q)$.

Intuitively, $P(Y(q) = c_1)$ can be simply estimated as:

$$P(Y(q) = c_i) = \frac{|\{o_j \in O_k(q)|Y(o_j) = c_i\}|}{k},$$

such that $P(Y(q) = c_i) \geqslant 0$ for each $i = 1, \ldots, l$ and $\sum_i P(Y(q) = c_i) = 1$. This means that all the $k$ nearest neighbors of a test SO $q$ equivalently contribute to estimate the class probability vector $y'$.

The alternative is to weight the contribution of the $k$ neighbors on the basis of the dissimilarity value with respect to $q$, giving greater weight to closer neighbors [32]. In particular, the class probability vector can be estimated by weighting the contribution of each neighbor $o_j$ according to the inverse of its dissimilarity value with respect to $q$. Indeed, this weighting strategy has been proved to perform better than bo weighted $k$-NN for finite samples [44].

Let $w_j = \frac{1}{d(q,o_j)}$ be the weight associated to each neighbour $o_j$, we denote:

$$W_i = \sum_{j=1}^{k} w_j \delta(c_i, Y(o_j)), \forall i = 1, \ldots, l,$$

where $\delta(c_i, Y(q_j)) = 1$ if $c_i = \hat{Y}(o_j)$, $\delta(c_i, Y(o_j)) = 0$, otherwise. Hence, the output class probabilities are estimated as follows:

$$P(Y(q) = c_i) = \frac{\frac{|\{o_j \in O_k(q)|Y(o_j)=c_i\}|}{k} \times W_i}{\sum_{i=1}^{l} \frac{|\{o_j \in O_k(q)|Y(o_j)=c_i\}|}{k} \times W_i}, \forall j = 1, \ldots, l.$$

The normalization by the summarization of all weighted class probabilities is required to guarantee that each probability class value is between 0 and 1 and the sum of the probabilities of all possible outcomes is 1.

Finally, the single-class output $Y(q)$ is obtained by returning the class $c_i$ such that $[P(Y(q) = c_i)] = \max(P(Y(q) = c_1), \ldots, P(Y(q) = c_l))$.

This weight-based estimation of class probabilities poses some problems due to the possible presence of one or more neighbors $o_i \in O_k(q)$ with $d(q, o_i) = 0$. We denote by $O_k^0(q)$ the proper sub-set of $O_k(q)$ such that $O_k^0(q) = \{o_j \in O_k(q)|d(q,o_j) = 0\}$. When $O_k^0(q)$ is not empty (i.e., $|O_k^0(q)| \neq \emptyset$),

$P(Y(q)$=c$_l)$ is an indeterminate form. To face this problem, we extend the solution discussed in [32] to the symbolic classification, that is, when there is exactly one neighbor $o_i$ with $d(q, o_i) = 0$, the same class of $o_i$ is assigned to $q$. More in general, if there are several neighbors $o_i \in O_k(q)$ with $d(q, o_i) = 0$ (i.e., $|O_k^0(q)| > 1$), the majority classification among them is assigned to $q$. By following this suggestion, SO-NN determines the class probability vector $y_1(q), \ldots, y_l(q)$ according to only neighbors $o_i \in O_k^0(q)$ when this set is not empty.

Two cases can be distinguished. In the first case, that is, $O_k^0(q) \neq \emptyset \wedge Y(q) = Y(o_j) = c, \forall o_j \in O_k^0(q)$, the class probability vector is computed as follows:

$$P(Y(q) = c) = 1 \text{ and } P(Y(q) = c_i) = 0 \; \forall c_i \neq c.$$

In the second case, that is, $\exists o_i, o_j \in O_k^0(q) \; s.t. \; o_i \neq o_j \wedge Y(o_i) \neq Y(o_j)$, the class probability vector is estimated as follows:

$$P(Y(q) = c_i) = \frac{\#\left\{o_j \in O_k^0(q) | Y(o_j) = c_i\right\}}{\#O_k^0(q)}, \forall i = 1, \ldots, l,$$

by considering only class values taken by neighbors of $q$ falling in $O_k^0(q)$.

### *4.2. Selecting the optimal $k$ value*

The performance of a $k$-NN classifier significantly depends on the size ($k$ value) of the neighborhood used to predict the unknown class value of a test case $q$ and a different size is appropriate for different problem domains.

In general, a $k$-NN classifier may work either as a *global* method using all training cases ($k = n$) to classify a new test case or as a *local* method using only the $k$ ($k < n$) nearest training cases. In the latter case, only data local to the region around $q$ actually contributes to estimating class probabilities. Henceforth, we will make the assumption that a single value of $k$ suffices to classify all testing cases $q$. This means that, in this work, we will consider only local methods with global determination of $k$.

Local methods have significant advantages when the probability distribution defined on the space of SOs for each class value is very complex, but it can still be described by a collection of less complex local approximations. Consequently, the choice of $k$ is critical, since it represents a trade-off between local and global approximations of the probability measures.

The choice of an appropriate value of $k$ can be based on a $v$-fold cross-validation approach. More precisely, the original training set $O$ is partitioned into $v$ blocks (or folds) of near-equal size, then, for every block, SO-NN is tested on it by using all the other blocks as a training set. In this way, the accuracy of SO-NN on each hold-out block is estimated and the average accuracy on the $v$ blocks can be considered as an approximation of SO-NN when all observations in $O$ are used as training examples. In our experimentation, we followed the recommendation of choosing a value of $v$ equals to 10 (ten-fold cross validation) [22].

The estimated accuracy of the SO-NN classifier for different values of $k$ enables the selection of the optimal $k$. Theoretically, we should try for different values of $k$ ranging in the interval [1, $n$]. Luckily, as observed in [44] it is not necessary to consider all possible values of $k$ during cross-validation to obtain the best performance: best performances are obtained by means of cross-validation on no more than approximately ten values of $k$. A similar consideration has also been reported in [18], where it is shown that the search for the optimal $k$ can be substantially reduced from [1, $n$] to [1, $\sqrt{n}$], without loosing too much accuracy in the approximation. Following this suggestion, the best $k$ is found on the sample [1, $\sqrt{n}$] and optimal $k$ can be approximated with this value without compromising accuracy in the approximation.

## 5. Experimental evaluation

SO-NN, whose implementation is publicly available,[3] has been empirically evaluated on symbolic data extracted from both real-world data and artificially generated data. These datasets collect information on individuals of fixed populations described by both a discrete target variable and one or more (continuous and discrete) explanatory variables resulting just in a single value. We used the DB2SO tool[4] [41] to aggregate original data on the basis of the values taken by some grouping variables. Grouping variables of each dataset include both the target variable and a subset of the explanatory variables describing the original data. The choice of the explanatory variables was performed on a case-by-case basis, after having examined the documentation provided with each dataset.

Both BSOs and PSOs have been generated in our experimental setting. In the case of PSOs, continuous variables have been a-priori discretized to generate modal symbolic variables.

To investigate the performances of SO-NN for distinct dissimilarity measures we had to solve a technical problem, due to the fact the computation of some dissimilarity measure for PSOs is indeterminate when a distribution has a zero-valued probability for some categories. To overcome this limitation, we used the *KT-estimate* [23] to estimate the probability distribution of modal variables. This estimate is based on the idea that no category of a modal symbolic variable in a PSO can be associated with a zero probability. The KT-estimate is computed as:

$$p(y) = \frac{(\text{No. times } y \text{ occursin } \{R_1, \ldots, R_M\}) + 1/2}{M + (K/2)},$$

where $y$ is the category of the modal symbolic variable, $\{R_1, \ldots, R_M\}$ are sets of aggregated individuals, $M$ is the number of individuals in the class, and $K$ is the number of categories of the modal symbolic variable.

Each symbolic dataset was partitioned into training set (70%) and testing set (30%) and accuracy performances were computed as follows:

$$A = \frac{1}{|TestingSet|} \sum_{o_j \in TestingSet} \delta(o_j(Y), class(o_j(Y))$$

where *class*$(o_j(Y))$ is the predicted class for a test SO $o_j$ and $\delta(o_j(Y), class(o_j(Y))$ is the indicator function equal to 1 if $o_j(Y) = class(o_j(Y))$, 0 otherwise.

Accuracy performed by SO-NN in classifying test SOs was evaluated by varying the dissimilarity measure $d$ used to determine the neighborhood of a testing case. Let $o_i$ be a test SO, SO-NN computes the probability vector $p_1(o_i), \ldots, p_l(o_i)$, where $p_j(o_i)$ denotes the probability that $o_i$ belongs to the class $c_j$ on the basis of the class values taken by the $k$ nearest neighbors of $o_i$ in the training set.

The predictive accuracy of SO-NN was also compared to that of TREE,[5] which is a state of the art classification system for symbolic data, whose theoretical foundations are found in [10]. TREE induces a binary decision tree from symbolic training data described by both set-valued and modal symbolic variables. Similarly to SO-NN, the target variable is a single-valued categorical variable.

[3] http://www.di.uniba.it/∼malerba/software/SONN/index.htm.

[4] We have actually implemented a version of DB2SO that supports the extraction of SOs by a process involving the querying of a relational database.

[5] The version of TREE considered in this work is that available in the ASSO workbench.

Candidate splits are evaluated by means of either the Gini measure or the Generalized Information (GInf) measure while leaves are associated with the most frequent class value over the training data falling in the corresponding leaf partition. TREE also allows the construction of decision trees with fuzzy tests associated with internal nodes.

TREE requires users to specify the minimal number of SOs falling into a split node. The choice of this threshold strongly affects the size of the induced tree (i.e. number of leaves). Indeed, a low value may cause the so-called overfitting of training data leading to a loss of accuracy in several applications [35]. In all experiments reported in this study we followed the empirical suggestion given in [31] and imposed that the minimal number of SOs falling into a split node must be greater than the root square number of SOs falling in the entire training set.

### *5.1. Experimental results on benchmarks for classification task*

SO-NN was tested on symbolic data extracted from twelve datasets taken from the UCI Machine Learning Repository (URL: http://www.ics.uci.edu/∼mlearn/MLRepository.html). Details on these datasets are reported in Table 4.

For each dataset, both BSOs and PSOs have been generated. In the case of PSOs, continuous variables have been a-priori discretized to generate modal symbolic variables. To this aim, we used the Relative Unsupervised DiscrEtization (RUDE) algorithm, which discretizes a continuous variable in the context defined by remaining continuous variables [27].

However, the high computational complexity of contextual discretization, prevents the application of RUDE to the one hundred and sixty-six continuous variables describing both "Musk 1" and "Musk 2" data. In this case, we used an equal-width discretization [8] that divides the range of each continuous variable into a constant number (i.e., 10) of intervals of equal width.

In addition to symbolic data extracted from UCI datasets, we considered the dataset "Taxonomy" produced in the context of the European project ASSO.[6] This dataset collects forty BSOs described according to sixteen symbolic variables (one interval variable, eight single-valued categorical variables and seven single-valued continuous variables).

For this data, we identified two different experimental settings where the goal of symbolic classification is significant. In the first setting, namely "Taxonomy 1", the goal is to predict the symbolic categorical variable "color" with domain {"white", "color"}, while in the second setting, namely "Taxonomy 2", the goal is to predict the symbolic categorical variable "k3k2" with domain {"11", "12"}. In both settings ten missing values occur on some explanatory variables.

In all experiments reported in this section, SO-NN performances are described in terms of estimated $k$ value, classification accuracy and running time by varying the dissimilarity measure $d$. It is noteworthy that all these results concern the use of $\mathrm{D}_1$ as componentwise function in the case of C1 and SO1 dissimilarity measures.

This depends by the fact that experiments on real data have enlighten no significant variation when varying the componentwise function $\mathrm{D}_i$.

Results on the number of neighbors ($k$) are reported in Table 5. It is noteworthy that the $k$ value estimated according to a 10-fold cross validation of training data varies only when SO-NN is tested on BSOs and PSOs extracted from Adult, Dermatology and Pima datasets. However such variations do not show any clear dependence between the estimated $k$ value and the dissimilarity measure $d$. The only

[6]The dataset is distributed with the ASSO workbench available at http://www.info.fundp.ac.be/asso/.

Table 4
UCI datasets used in the empirical evaluation of SO-NN

| Data sets | No. Cases | No. Variables | | Grouping variables | No. SOs | Class |
|---|---|---|---|---|---|---|
| | | Cont | Discr | | | |
| Adult | 32561 | 6 | 9 | Category, education, occupation, work class | 1134 | Income ("\$50K"or "$\leqslant$ \$50K") |
| Dermatology | 366 | 1 | 33 | Category, borders, erythema, family history, follicular papules, follicular horn plug, itching, knee and elbow involvement, koebner phenomenon, oral mucosal involvement, polygonal papules, scaling, scalp | 302 | Type of erythematic squamous disease ("cronic dermatitis", "lichen planus", "pityriasis rosea", "pityriasis rubra pilaris", "psoriasis" or "seboreic dermatitis") |
| Flare 1C | 323 | 0 | 11 | C-category flares production, code for class, code for largest spot size, code for spot distribution | 59 | Count of solar flares of C-class occuring in a 24 hour period ("0", "1" or "2") |
| Flare 1M | 323 | 0 | 11 | M-category flares production, code for class, code for largest spot size, code for spot distribution | 57 | Count of solar flares of M-class occuring in a 24 hour period ("0", "1", "2" or"4") |
| Flare 1X | 323 | 0 | 11 | X-category flares production, code for class, code for largest spot size, code for spot distribution | 43 | Count of solar flares of X-class occurring in a 24 hour period ("0" or "1") |
| Flare 2C | 1066 | 0 | 11 | C-category flares production, code for class, code for largest spot size, code for spot distribution | 122 | Count of solar flares of C-class occurring in a 24 hour period ("0", "1", "2", "3", "4", "5", "6" or "8") |
| Flare 2M | 1066 | 0 | 11 | M-category flares production, code for class, code for largest spot size, code for spot distribution | 70 | Count of solar flares of M-class occurring in a 24 hour period ("0", "1", "2", "4" or "5") |
| Flare 2X | 1066 | 0 | 11 | X-category flares, code for class, code for largest spot size, code for spot distribution | 51 | Count of solar flares of X-class occurring in a 24 hour period ("0", "1" or "2") |
| Musk1 | 476 | 166 | 3 | Category, molecule name, conformation name | 92 | Molecule type ("musk" or "non musk") |
| Musk2 | 6598 | 166 | 3 | Category, molecule name, conformation name | 102 | Molecule type ("musk" or "non musk") |
| Mushroom | 8124 | 0 | 23 | Category, cap colour, cap shape, cap surface | 133 | Mushroom type ("poisonous" or "edible") |
| Pima | 768 | 8 | 1 | Category, age, number of times pregnant | 385 | Tested positive for diabetes ("true" or "false") |

clear regularity observable in Table 5 is that $k$ tends to be close to the highest permitted value ($\sqrt{n}$). This means that in the trade-off between bias and variance, the increase of bias due to larger size of $O_k(q)$ when estimating probabilities is well compensated by the decrease of variance (distribution of cases in $O_k(q)$) between the $l$ classes [17].

Results on accuracy performed by both SO-NN and TREE are reported in Table 6 for BSOs, while in Table 7 for PSOs. These results concern both accuracy performed by SO-NN when varying the dissimilarity measure and accuracy performed by TREE when varying the splitting measure.

Mean value and standard deviation of accuracy performed by both methods on such datasets are reported in Table 8. Several conclusions can be drawn from these experimental results.

Table 5
The values of $k$ estimated by SO-NN according to a ten-fold cross validation on training data by varying the dissimilarity measure (DM)

| DM | Adult | Derma tology | Flare 1C | Flare 1M | Flare 1X | Flare 2C | Flare 2M | Flare 2X | Musk1 k1 | Musk2 k2 | Mush room | Pima | Taxon omy1 | Taxon omy2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| U1 | 13 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 17 | 5 | 5 |
| U2 | 25 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 16 | 5 | 5 |
| U3 | 28 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 17 | 5 | 5 |
| U4 | 28 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 17 | 5 | 5 |
| C1 | 2 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 8 | 5 | 5 |
| SO1 | 2 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 16 | 5 | 5 |
| SO2 | 2 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 10 | 5 | 5 |
| SO3 | 2 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 17 | 5 | 5 |
| SO4 | 2 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 15 | 5 | 5 |
| SO5 | 2 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 8 | 5 | 5 |
| SO6 | 2 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 16 | 5 | 5 |
| P1-J | 28 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 2 | – | – |
| P1-CHI2 | 28 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 2 | – | – |
| P1-REN | 2 | 4 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 17 | – | – |
| P1-CHER | 2 | 4 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 17 | – | – |
| P1-LP | 14 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 2 | – | – |
| P2 | 28 | 15 | 7 | 6 | 6 | 9 | 7 | 6 | 8 | 9 | 10 | 2 | – | – |
| P3 | 22 | 13 | 7 | 6 | 6 | 4 | 7 | 6 | 8 | 9 | 10 | 16 | – | – |

Table 6
SO-NN vs. TREE on BSOs: accuracy on testing set. SO-NN is evaluated by varying dissimilarity measure, while TREE is evaluated by varying the nature of splitting test, i.e. pure splits with either Generalized Information (GInf) measure or Gini measure as splitting criterion or fuzzy splits. Best accuracy is in italics

| SYSTEM | | Adult | Derma tology | Flare 1C | Flare 1M | Flare 1X | Flare 2C | Flare 2M | Flare 2X | Musk1 k1 | Musk2 k2 | Mush room | Pima | Taxon omy1 | Taxon omy2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SO-NN | U1 | 0.78 | 0.94 | 0.65 | 0.81 | 0.92 | 0.37 | 0.52 | 0.87 | 0.67 | 0.70 | 0.85 | 0.58 | 0.83 | 0.83 |
| | U2 | 0.64 | 0.61 | 0.65 | 0.81 | 0.92 | 0.37 | 0.52 | 0.87 | 0.59 | 0.77 | 0.90 | 0.65 | 0.83 | 0.5 |
| | U3 | 0.77 | 0.97 | 0.65 | 0.81 | 0.92 | 0.34 | 0.52 | 0.87 | 0.56 | 0.73 | 0.87 | 0.71 | 0.83 | 0.67 |
| | U4 | 0.77 | 0.97 | 0.65 | 0.81 | 0.92 | 0.34 | 0.52 | 0.87 | 0.56 | 0.73 | 0.87 | 0.71 | 0.83 | 0.67 |
| | C1 | 0.64 | 0.91 | 0.65 | 0.81 | 0.92 | 0.37 | 0.52 | 0.87 | 0.7 | 0.63 | 0.92 | 0.61 | 0.83 | 0.92 |
| | SO1 | 0.64 | 0.91 | 0.65 | 0.81 | 0.92 | 0.37 | 0.52 | 0.87 | 0.7 | 0.63 | 0.92 | 0.61 | 0.83 | 0.92 |
| | SO2 | 0.7 | 0.91 | 0.65 | 0.81 | 0.92 | 0.37 | 0.52 | 0.87 | 0.7 | 0.63 | 0.92 | 0.62 | 0.83 | 0.75 |
| | SO3 | 0.6 | 0.43 | 0.59 | 0.75 | 0.75 | 0.37 | 0.52 | 0.87 | 0.56 | 0.73 | 0.90 | 0.61 | 0.83 | 0.67 |
| | SO4 | 0.6 | 0.43 | 0.59 | 0.75 | 0.75 | 0.37 | 0.52 | 0.87 | 0.56 | 0.73 | 0.90 | 0.61 | 0.83 | 0.58 |
| | SO5 | 0.6 | 0.43 | 0.59 | 0.81 | 0.83 | 0.37 | 0.52 | 0.87 | 0.56 | 0.6 | 0.97 | 0.59 | 0.83 | 0.67 |
| | SO6 | 0.73 | 0.95 | 0.65 | 0.81 | 0.92 | 0.37 | 0.52 | 0.87 | 0.56 | 0.73 | 0.97 | 0.63 | 0.92 | 0.67 |
| | Best SONN | 0.78 | *0.97* | 0.65 | *0.81* | 0.92 | 0.37 | 0.52 | *0.87* | *0.7* | 0.77 | *0.97* | *0.71* | *0.92* | *0.92* |
| TREE | GInf | 0.79 | 0.89 | 0.7 | 0.62 | 1 | 0.38 | 0.56 | error | 0.48 | 0.8 | 0.94 | 0.68 | 0.83 | 0.58 |
| | Gini | 0.79 | 0.88 | 0.7 | 0.68 | 1 | 0.33 | 0.56 | error | 0.48 | 0.8 | 0.89 | 0.66 | 0.83 | 0.58 |
| | Fuzzy | 0.59 | 0.39 | 0.7 | 0.75 | 1 | 0.33 | 0.56 | 0.86 | 0.59 | 0.73 | 0.89 | 0.7 | 0.83 | 0.58 |
| | Best TREE | *0.79* | 0.89 | *0.7* | 0.75 | *1* | *0.38* | *0.56* | 0.86 | 0.59 | *0.8* | 0.94 | 0.7 | 0.83 | 0.58 |

First of all, SO-NN appears able to take advantage of the highly adaptive behavior of the lazy learning approach to locally approximate the class value of a test SO without compromising the accuracy of prediction. Indeed, SO-NN generally classifies test SOs better than or, at worst, approximately equally

Table 7
SO-NN vs. TREE on PSOs: accuracy on testing set. SO-NN is evaluated by varying dissimilarity measure, while TREE is evaluated by varying the nature of splitting test, i.e. pure splits with either Generalized Information (GInf) measure or Gini measure as splitting criterion or fuzzy splits. Best accuracy is in italics

| SYSTEM | | | Adult | Derma tology | Flare 1C | Flare 1M | Flare 1X | Flare 2C | Flare 2M | Flare 2X | Musk1 k1 | Musk2 k2 | Mush room | Pima |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SO-NN | P1 | J | 0.62 | 0.94 | 0.59 | 0.69 | 0.75 | 0.24 | 0.57 | 0.87 | 0.7 | 0.63 | 0.82 | 0.63 |
| | | CHI2 | 0.58 | 0.93 | 0.59 | 0.69 | 0.75 | 0.24 | 0.57 | 0.87 | 0.67 | 0.63 | 0.59 | 0.63 |
| | | REN | 0.32 | 0.78 | 0.59 | 0.69 | 1 | 0.21 | 0.57 | 0.87 | 0.74 | 0.63 | 0.56 | 0.61 |
| | | CHER | 0.32 | 0.78 | 0.59 | 0.69 | 1 | 0.21 | 0.57 | 0.87 | 0.74 | 0.63 | 0.56 | 0.61 |
| | | LP | 0.59 | 0.93 | 0.59 | 0.63 | 0.83 | 0.32 | 0.57 | 0.87 | 0.67 | 0.6 | 0.85 | 0.63 |
| | | P2 | 0.73 | 0.93 | 0.53 | 0.81 | 0.92 | 0.24 | 0.61 | 0.87 | 0.7 | 0.57 | 0.79 | 0.60 |
| | | P3 | 0.62 | 0.1 | 0.71 | 0.88 | 1 | 0.26 | 0.61 | 0.87 | 0.44 | 0.63 | 0.62 | 0.64 |
| | Best SONN | | *0.73* | *0.94* | *0.71* | *0.88* | *1* | *0.32* | *0.61* | *0.87* | *0.74* | 0.63 | 0.82 | 0.64 |
| TREE | GInf | | error | 0.44 | 0.70 | 0.87 | error | error | error | error | 0.74 | 0.73 | error | Error |
| | Gini | | 0.73 | 0.90 | 0.70 | 0.87 | 1 | 0.30 | 0.56 | error | 0.74 | 0.7 | 0.94 | 0.66 |
| | Fuzzy | | 0.59 | 0.39 | 0.70 | 0.75 | 1 | error | error | 0.86 | 0.55 | 0.7 | error | 0.66 |
| | Best TREE | | 0.73 | 0.9 | 0.7 | 0.87 | 1 | 0.3 | 0.5 | 0.86 | 0.74 | *0.73* | *0.94* | *0.66* |

Table 8
The mean and standard deviation of the accuracy performed for each symbolic dataset

| Dataset | BSOs | | | | PSOs | | | |
|---|---|---|---|---|---|---|---|---|
| | SO-NN | | TREE | | SO-NN | | TREE | |
| | Mean | St. Dev. | Mean | St. Dev. | Mean | St. Dev. | Mean | St. Dev. |
| Adult | 0.68 | 0.07 | 0.69 | 0.12 | 0.54 | 0.16 | 0.67 | 0.10 |
| Dermatology | 0.77 | 0.24 | 0.62 | 0.31 | 0.77 | 0.30 | 0.58 | 0.28 |
| Flare1C | 0.63 | 0.03 | 0.71 | 0 | 0.60 | 0.05 | 0.71 | 0 |
| Flare1M | 0.80 | 0.02 | 0.69 | 0.06 | 0.73 | 0.09 | 0.83 | 0.07 |
| Flare1X | 0.88 | 0.07 | 1.00 | 0 | 0.89 | 0.12 | 1.00 | 0 |
| Flare2C | 0.36 | 0.01 | 0.35 | 0.03 | 0.25 | 0.04 | 0.31 | 0 |
| Flare2M | 0.52 | 0 | 0.56 | 0 | 0.58 | 0.02 | 0.57 | 0 |
| Flare2X | 0.87 | 0 | 0.87 | 0 | 0.87 | 0.00 | 0.87 | 0 |
| Musk 1 | 0.61 | 0.07 | 0.52 | 0.06 | 0.67 | 0.10 | 0.68 | 0.11 |
| Musk 2 | 0.69 | 0.06 | 0.78 | 0.04 | 0.62 | 0.02 | 0.71 | 0.02 |
| Mushroom | 0.77 | 0.24 | 0.91 | 0.03 | 0.68 | 0.13 | 0.95 | 0 |
| Pima | 0.63 | 0.04 | 0.69 | 0.02 | 0.62 | 0.01 | 0.67 | 0 |
| Taxonomy1 | 0.71 | 0.13 | 0.58 | 0 | – | – | – | – |
| Taxonomy2 | 0.84 | 0.03 | 0.83 | 0 | – | – | – | – |

to TREE.[7] This is confirmed by results of the pairwise comparison between SO-NN and TREE when considering for each dataset the best accuracy obtained by varying the dissimilarity measure and the splitting measure, respectively.

The pairwise comparison is performed according to the non-parametric Wilcoxon test [34] by assuming that the experimental results (i.e., accuracy) of the two methods compared are independent pairs of sample data $\{(u_1, v_1), (u_2, v_2\}), \ldots, (u_n, v_n)\}$. We then rank the absolute value of the differences $|u_1 - v_1|$. The Wilcoxon statistics $W^+$ and $W^-$ are the sum of the ranks from the positive and negative differences, respectively.

We test the null hypothesis $H_0$: "no difference in distributions" against the two-sided alternative $H_1$:

[7] TREE performs better than SO-NN only on the BSOs extracted from Adult, Flare 1-C, Flare 1-X, Flare 2-C, Flare 2-M and Musk 2 and the PSOs extracted from Musk2, Mushroom and Pima. However, differences in accuracy are of the order of $10^{-2}$ with respect to the best result reported for SO-NN.

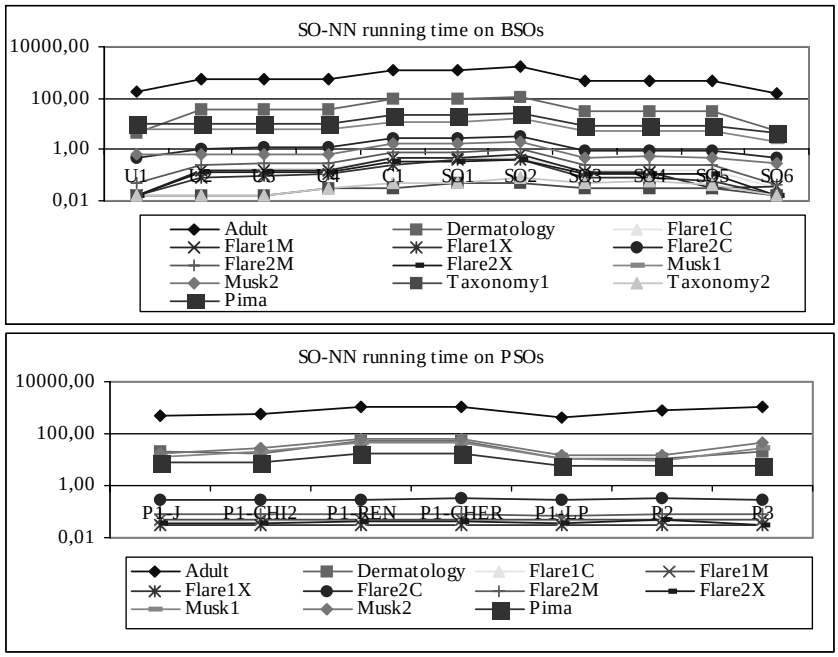


Fig. 2. SO-NN running time by varying dissimilarity measure. For each symbolic dataset, running time is represented in a logarithmic scale.

"there is a difference in distributions". More formally, the hypotheses are: $H_0$: "$\mu_u = \mu_v$" against $H_1$: "$\mu_u \neq \mu_v$". Intuitively, when $W^+ > W^-$ and vice-versa, $H_0$ is rejected. Whether $W^+$ should be considered "much greater than" $W^-$ depends on the significance level $p$ ($p \leqslant \alpha$). The basic assumption of the statistical test is that the two populations have the same continuous distribution (and no ties occur). Since, in our experiments $u_i$ and $v_i$ are the classification accuracy on the same testing sets, $W^+ > W^-$ implies that the first method (U) is better than the second (V). In particular, when we compared the best accuracy performed by SO-NN with best accuracy performed by TREE on each dataset we obtained that $W^+ = 69$, $W^- = 36$ and $p = 0.32$ in the boolean case and $W^+ = 22.5$, $W^- = 22.5$ and $p = 1$ in the probabilistic one. This confirms that SO-NN and TREE are statistically equivalent when considering the best accuracy returned by both methods.

A second observation concerns the effect of missing values. In both "Taxonomy 1" and "Taxonomy 2" ten missing values occur on some explanatory variables. In both cases SO-NN performs better than TREE (0.92 vs. 0.83 on "Taxonomy 1" and 0.92 vs. 0.58 on "Taxonomy 2").

A third observation concerns the dissimilarity measure $d$ used to both determine and weight neighbors in training data for each test SO to be classified. By varying $d$, we observe a variation on not only the accuracy of classification, but also the running time to classify the set of test SOs. SO-NN running time is shown in Fig. 2. For each symbolic dataset (either BSOs or PSOs), SO-NN running time (in seconds) to classify the entire test set is reported along the vertical axis in a logarithmic scale, while dissimilarity measures are listed along the horizontal axis.

By analyzing the accuracy of SO-NN classification on BSOs, we observe that only in the case of BSOs extracted from either "Flare-2M" or "Flare-2X", SO-NN accuracy on the test set is persistently 0.52 and 0.87 respectively, also when varying the dissimilarity measure. Consequently, in both cases the standard deviation of SO-NN accuracy when varying dissimilarity measure is 0. On the other hand,

Table 9
Results of the Wilcoxon test (p-value) on the accuracy of the classification performed by SO-NN on BSOs when comparing the dissimilarity measure reported on each row vs. the dissimilarity measure reported on each column. The statistically significant values p-value $\leqslant 0.07$ are in italics. The sign $+$ $(-)$ indicates $W+ > W-$ $(W + < W-)$, that is, the dissimilarity measure on the corresponding row outperforms the dissimilarity measure on the corresponding column (or vice-versa)

| p value | SO-NN | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | U1 | U2 | U3 | U4 | C1 | SO1 | SO2 | SO3 | SO4 | SO5 | SO6 |
| U1 | 1 | 0.21 | 1 | 1 | 0.81 | 0.81 | 0.46 | (+)*0.02* | (+)*0.02* | (+)*0.07* | 0.94 |
| U2 | 0.21 | 1 | 0.31 | 0.31 | 0.43 | 0.43 | 0.29 | (+)*0.07* | (+)*0.07* | 0.25 | 0.10 |
| U3 | 1 | 0.31 | 1 | 1 | 1 | 0.57 | (+)*0.05* | (+)*0.03* | (+)*0.01* | (+)*0.07* | 0.68 |
| U4 | 1 | 0.31 | 1 | 1 | 1 | 0.57 | 0.84 | (+)*0.03* | (+)*0.01* | (+)*0.07* | 0.68 |
| C1 | 0.81 | 0.43 | 1 | 1 | 1 | 1 | 1 | (+)*0.03* | (+)*0.03* | (+)*0.02* | 0.74 |
| SO1 | 0.81 | 0.43 | 0.57 | 0.57 | 1 | 1 | 1 | (+)*0.03* | (+)*0.03* | (+)*0.02* | 0.74 |
| SO2 | 0.46 | 0.29 | *0.05* | 0.84 | 1 | 1 | 1 | (+)*0.02* | (+)*0.01* | (+)*0.01* | 0.54 |
| SO3 | (−)*0.02* | (−)*0.07* | (−)*0.03* | (−)*0.03* | (−)*0.03* | (−)*0.03* | (−)*0.02* | 1 | 1 | 0.81 | 0.81 |
| SO4 | (−)*0.02* | (−)*0.07* | (−)*0.01* | (−)*0.01* | (−)*0.03* | (−)*0.03* | (−)*0.01* | 1 | 1 | 0.81 | 0.81 |
| SO5 | (−)*0.07* | 0.25 | (−)*0.07* | (−)*0.07* | (−)*0.02* | (−)*0.02* | (−)*0.01* | 0.81 | 0.81 | 1 | (−)*0.01* |
| SO6 | 0.94 | 0.10 | *0.04* | 0.68 | 0.74 | 0.74 | 0.54 | 0.81 | 0.81 | (+)*0.01* | 1 |

in the remaining datasets, the choice of the dissimilarity measure evidently affects the accuracy of the SO-NN classification. For instance, SO-NN accuracy is 0.95 when classifying test BSOs extracted from "Dermatology" using neighbors defined and weighted according to the SO6 dissimilarity measure, but the same accuracy decreases to 0.43 when neighbors are defined according to the SO5 dissimilarity measure. This is confirmed by the corresponding high value of standard deviation (0.24) of SO-NN accuracy on the Dermatology boolean dataset.

To complete the analysis of the dissimilarity measures for BSOs we discuss the results of Wilcoxon test to evaluate the statistical differences on the accuracy performed by SO-NN for each pair of dissimilarity measures. These results are reported in Table 9 and underline the worst performance of SO3, SO4 and SO5 when adopted in k-NN classification on real data, while no significant behavior is observed for the remaining dissimilarity measures.

On the other hand, when we analyze the running time, we observe that lower running time in the classification step is reached only when SO-NN uses either U1 or SO6 dissimilarity measure to estimate the distance between SOs in the neighborhood definition. It is noteworthy that, in this case, lower running time often coincides with better accuracy in classification.

Different considerations are suggested when we analyze the accuracy of SO-NN classification and neighbors are determined according to C1, SO1 or SO2 dissimilarity measures. In this case, higher complexity in defining neighbors does not necessarily correspond to better classification accuracy. Similar considerations are suggested when we jointly analyze the accuracy and running time of SO-NN classification on PSOs. Higher running time is performed when the distance between two PSOs is evaluated with a P1 dissimilarity measure by using either Rènyi's distance (REN) or Chernoff's distance (CHER) to estimate the dissimilarity coefficient between probability distributions, but it is quite difficult to identify a dissimilarity measure that guarantees better accuracy on classifying unknown testing SOs in each domain. This is confirmed by results of Wilcoxon test reported in Table 10. A solution can be to automatically determine the best dissimilarity measure according to the accuracy of the SO-NN classification on a cross validation of training data.

We conclude by observing that this study is quite different from previous work on dissimilarity measures for BSOs [28], where the authors discussed the results with respect to the Monotonic Increasing Dissimilarity (MID) property of dissimilarity measures. This kind of analysis is based on the assumption that it is sensible to define a dissimilarity on all the variables used for grouping. In this case, indeed,

Table 10
Results of the Wilcoxon test (p-value) on the accuracy of the classification performed by SO-NN on PSOs when comparing the dissimilarity measure reported on each row vs. the dissimilarity measure reported on each column

| p value | SO-NN | | | | | | |
|---|---|---|---|---|---|---|---|
| | P1+J | P1+CHI2 | P1+REN | P1+CHER | P1+LP | P2 | P3 |
| P1+J | 1.00 | 0.13 | 0.38 | 0.38 | 0.95 | 0.57 | 1 |
| P1+CHI2 | 0.13 | 1.00 | 0.58 | 0.58 | 0.31 | 0.20 | 0.38 |
| P1+REN | 0.38 | 0.58 | 1.00 | 1 | 0.50 | 0.37 | 0.50 |
| P1+CHER | 0.38 | 0.58 | 1.00 | 1 | 0.50 | 0.37 | 1 |
| P1+LP | 0.95 | 0.31 | 0.50 | 0.50 | 1 | 0.70 | 1 |
| P2 | 0.57 | 0.20 | 0.37 | 0.37 | 0.70 | 1 | 0.70 |
| P3 | 1.00 | 0.38 | 0.50 | 1 | 1 | 0.70 | 1 |

it is possible to check whether the degree of dissimilarity between SOs, computed on the independent variables, is proportional to the dissimilarity computed on the grouping variables. For instance, in the case of the Abalone dataset investigated in [28], the variable "number of rings" can be used to group the original set of 4,177 individuals (a kind of marine crustacean). This integer variable ranges between 1 and 29 so that nine distinct BSO can be generated by considering intervals of length 3 (e.g., [1, 3], [4, 5], and so on). Since it is sensible to assume that two abalones with the same number of rings should also present similar values for the other variables (e.g., sex, length, diameter, height) used to describe the crustaceans, we expect to observe a direct proportionality between the degree of dissimilarity between BSOs computed on the independent variables and the difference in the number of rings (MID property). The analysis of the MID property on the dataset reported in [28] showed that this property does not hold when the dissimilarity measure is computed according to the U1 measure. Indeed, U1 surprisingly discovers that old crustaceans with a high number of rings (25–29) are considered more similar to very young ones with a low number of rings (1–3) than to middle-aged abalones with 16–18 rings. However, this may depend on the fact that when BSOs are generated from unequally distributed individuals, with respect to a given class variable, a distance measure based on the spanning factor (e.g. U1) may lead to unexpected results [28].

### *5.2. Experimental results on artificial datasets*

SO-NN was also tested on artificial data generated for the synthetic waveform recognition problem described in [7, pp. 49–55]. This is a three-class problem that assumes an a-priori known model for data generation. This model is based on the waveforms $h_1$, $h_2$ and $h_3$, which represent a shifted triangular distribution defined as follows:

- $h_1(i) = \max\{6 - |i - 7|, 0\}$,
- $h_2(i) = h_1(i - 4)$ and
- $h_3(i) = h_1(i - 8)$.

Data have been generated for three classes of wave, namely $c_1$, $c_2$ and $c_3$ such that 1000 individuals have been built for each class $c_k$. Each individual $i$ is described according to 21 continuous variables $X_j$ ($j = 1, \ldots, 21$) defined as follows:

- $x_j(i) = u(i)h_1(j) + (1 - u(i))h_2(j) + e_j(i), j = 1, \ldots, 21$ for $i$ labelled with $c_1$,
- $x_j(i) = u(i)h_1(j) + (1 - u(i))h_3(j) + e_j(i), j = 1, \ldots, 21$ for $i$ labelled with $c_2$,
- $x_j(i) = u(i)h_2(j) + (1 - u(i))h_3(j) + e_j(i), j = 1, \ldots, 21$ for $i$ labelled with $c_3$,

where $U$ is an uniform random variable on the interval [0, 1] and $E_j$ ($j = 1, \ldots, 21$) are independent Gaussian random variables with zero mean and unit variance.

For each class, data have been aggregated according to the uniform variable $U$. To this aim, we considered four grouping modalities. The first one (L1) corresponds to build 10 SOs for each class $c_k$. In particular, the description of each SO *wave_k_m* ($k = 1, 2, 3$ and $m = 1, \ldots, 10$) is obtained with DB2SO by generalizing the group of individuals $i$ of class $c_k$ with $u(i) \in [0.1(m-1), 0.1m]$. The second one (L2) generates 20 SOs for each class $c_k$ such that the description of *wave_k_m* ($k = 1, 2, 3$ and $m = 1, \ldots, 20$) is derived from the group of individuals $i$ of class $c_k$ with $u(i) \in [0.05(m-1), 0.5m]$. The third one (L3) generates 50 SOs for each class $c_j$ such that the description of *wave_k_m* ($k = 1, 2, 3$ and $m = 1, \ldots, 50$) is derived from the group of individuals $i$ of class $c_k$ with $u(i) \in [0.02(m-1), 0.02m]$. Finally, the forth one (L4) generates 100 SOs for each class $c_j$ such that the description of *wave_k_m* ($k = 1, 2, 3$ and $m = 1, \ldots, 100$) is derived from the group of individuals $i$ of class $c_k$ such that $u(i) \in [0.01(m-1), 0.01m]$.

Both BSOs and PSOs have been generated. BSOs have been built in two different settings. In the first setting (BSOs-Interval variables), descriptions of BSOs have been derived from original data described by the continuous-valued variables $X_j$ ($j = 1, \ldots, 21$). In the second setting (BSOs-Multi-valued variables), continuous variables $X_j$ have been a-priori discretized according to an equal-width discretization [8] that divides the range of each continuous variable into a constant number (i.e., 10) of intervals of equal width. BSOs are then built from such discrete-valued data. The same discrete data have then been used to generate PSOs.

We used VDiss tool[8] to plot the SOs as points of a bi-dimensional plane. This scatterplot visualization is based on an extension of the Sammon's algorithm [39] that takes as input the dissimilarities among each pair of SOs and returns a collection of points such that Euclidean distances among points preserve original dissimilarity values. Scatterplot visualization of SOs built from this artificial data (see Fig. 3) confirms that SOs are distributed in three clusters of spatially close points where each cluster correspond to a different class label.

In Fig. 4 the results on accuracy performed by both SO-NN and TREE are reported. Results on such artificial data confirms that SO-NN takes advantage of the highly adaptive behavior of the lazy learning approach to locally approximate the class value of a test SO without compromising the accuracy of prediction. Since these artificial data have been generated in order to preserve a correspondence between the notion of proximity on the class variable and proximity on explanatory variables, some instructive considerations on dissimilarity measures are suggested from corresponding variation of SO-NN accuracy.

Firstly, we observe that accuracy of SO-NN classification decreases when the componentwise function $D_3$ is adopted to compute the dissimilarity measures $C_1$ or $SO_1$ between BSOs. This result is quite different from the corresponding result obtained on the BSOs extracted from real data where no significant variation was observed when varying the componentwise function.

Secondly, SO-NN accuracy on BSOs whose descriptions include interval variables confirms the worst performance of some dissimilarity measures such as SO3 and SO4.

## 6. Conclusions

Finding novel and interesting patterns in large databases without violating the inherent confidentiality of micro-data is a great challenge for the Data Mining community. A review of the issues and state-of-the-art of privacy-preserving data mining can be found in [43]. Basically, two approaches are identified:

[8] We have implemented VDiss tool that is actually available in the ASSO workbench.

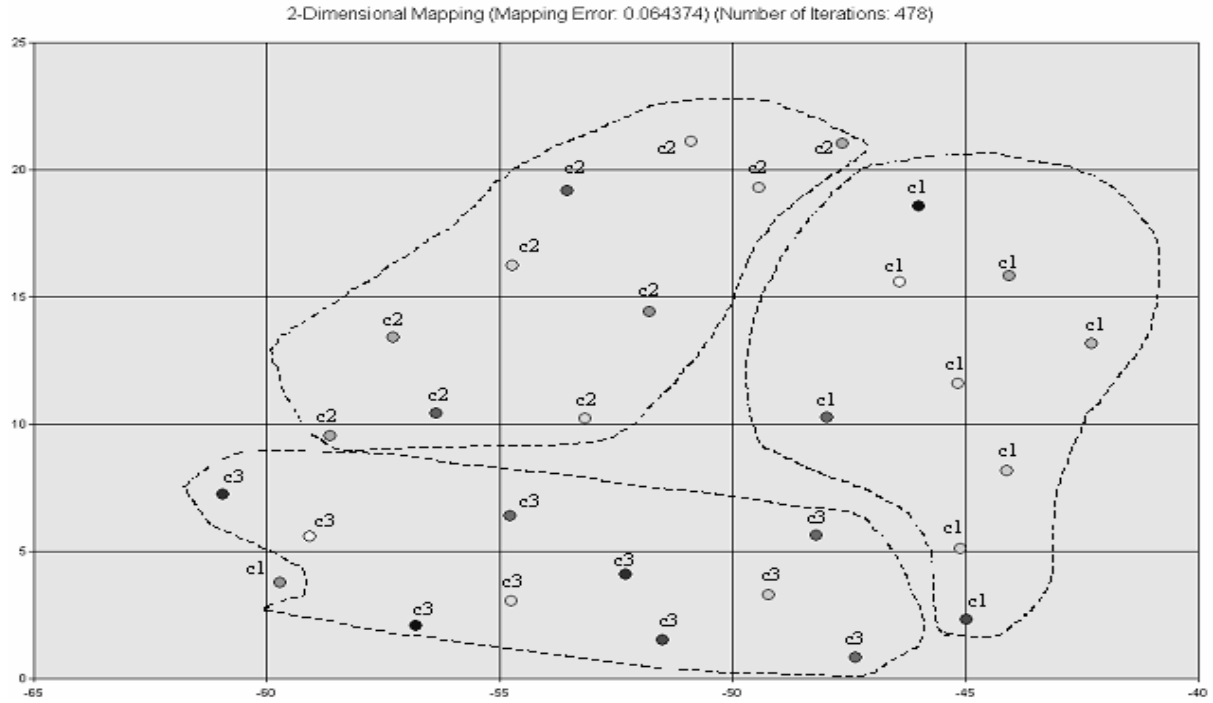


Fig. 3. Bi-dimensional plotting of the set of BSOs with interval variables built according to the L1 modality. Dissimilarity among BSOs are computed with the Gowda and Diday's dissimilarity measure (U1).

either perturbing the data by adding noise to it, or using cryptographic techniques to preserve privacy. Symbolic Data Analysis developed since the early '80s, offers a third approach based on the analysis of aggregated data, where aggregation is obtained through generalization over groups of individuals.

Several methods for the analysis of the symbolic data table have been reported in the literature. In this paper, we considered only those conceived for classification purposes. They are all based on an eager approach to learning, according to which a general explicit description (e.g. decision tree, neural network or rule set) of a discrete-valued target function (class label) is built when training data are provided. Alternatively, we proposed a lazy learning method, named SO-NN, based on the extension of the k-NN classifier. A key advantage of this method is that instead of estimating the target function at once for the entire instance space, it is possible to estimate it locally and differently for each observation to be classified.

A comparison with a state of the art symbolic classification system, namely TREE, has been reported on two restricted classes of symbolic objects (BSOs and PSOs), extracted from benchmark datasets. Results show that SO-NN is generally able to exploit the highly adaptive behavior of the lazy learning approach to locally approximate the unknown class value of a test SO without compromising the accuracy of prediction also in the presence of missing values.

Moreover, since the core of SO-NN classification is the dissimilarity measure used to determine and weight the neighbors of a test SO to be classified and the dissimilarity measure can be replaced without any side effects, we have exploited SO-NN framework to compare several dissimilarity measures proposed in the literature for both BSOs and PSOs.

One practical issue in applying k-NN is that the distance between observations is calculated based on *all* explanatory variables. When many of the variables are irrelevant, these will dominate in the computation

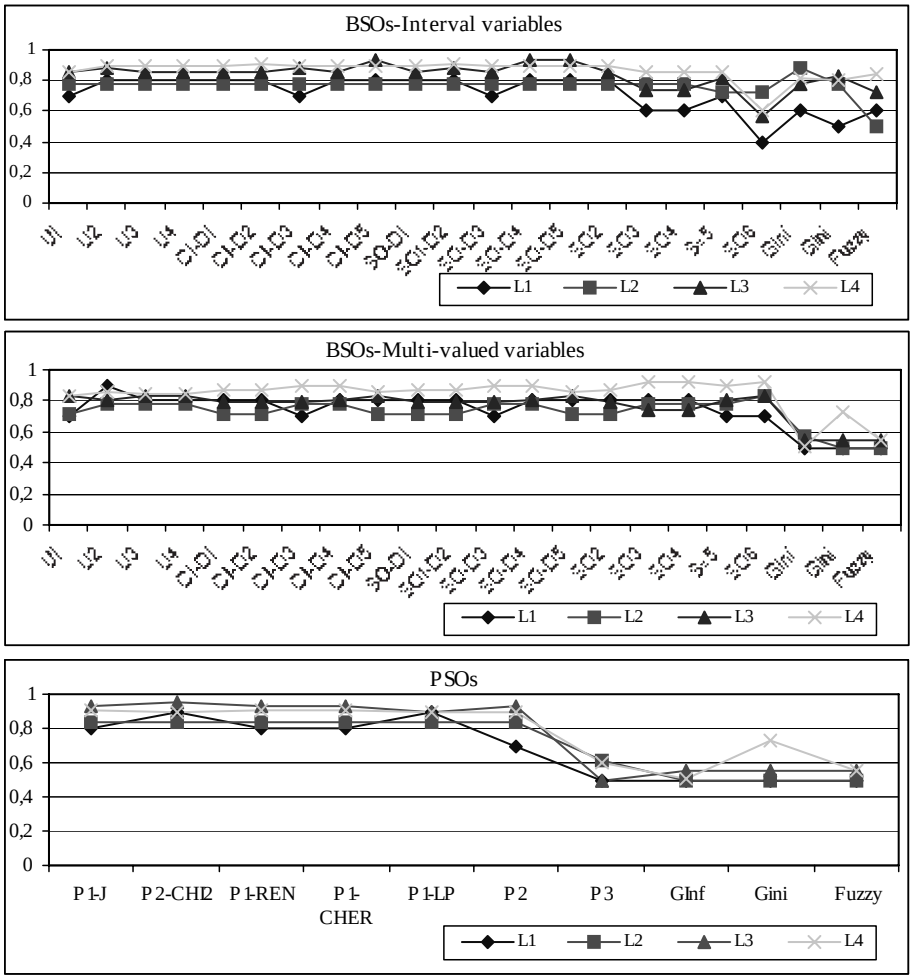


Fig. 4. SO-NN vs. TREE: accuracy on testing set. SO-NN is evaluated by varying the dissimilarity measure, while TREE is evaluated by varying the nature of splitting test.

of the dissimilarity, thus resulting in a low predictive accuracy. An approach to overcoming this problem is to weight each variable separately when computing the dissimilarity between two observations, in order to suppress the effect of irrelevant variables. SO-NN naturally supports weighting mechanisms for symbolic variables, since several dissimilarity measures proposed for both BSOs and PSOs allow symbolic variables to be weighted. As future work we intend to explore this possibility in order to improve the results already obtained with the proposed method.

## Acknowledgements

This work is partially supported by the COFIN 2001 project on “Methods of knowledge discovery, validation and representation of the statistical information in decision tasks” and the ATENEO-2004 project on “Methods of multi-relational data mining for knowledge discovery in databases”. The authors thank Yves Lechevallier for his support in using TREE and Edwin Diday for his constructive suggestions on this work.

## References


[1] D. Aha, D. Kibler and M. Albert, Instance-based learning algorithms, *Machine Learning* **6**(1) (1991), 37–66.

[2] S.M. Ali and S.D. Silvey, A general class of coefficient of divergence of one distribution from another, *Journal of the Royal Statistical Society* **B2** (1966), 131–142.

[3] A. Appice, M. Ceci, A. Lanza, F.A. Lisi and D. Malerba, Discovery of spatial association rules in geo-referenced census data: A relational mining approach, *Intelligent Data Analysis* **7**(6) (2003), 541–566.

[4] K.J. Beirlant, L. Devroye, L. Györfi and I. Vajda, Large deviations of divergence measures on partitions, *Journal of Statistical Planning and Inference* **93** (2001), 1–16.

[5] H.H. Bock and E. Diday, Symbolic Objects, in: *Analysis of Symbolic Data. Exploratory Methods for extracting Statistical Information from Complex Data*, H.H. Bock and E. Diday, eds, Series: Studies in Classification, Data Analysis, and Knowledge Organisation, 15, Springer-Verlag: Berlin, 2000.

[6] M.C. Bravo, Strata Decision Tree Symbolic Data Analysis Software, in: *Data Analysis, Classification and Related Methods*, H.A.L. Kiers, J.P. Rasson, P.J.F. Groenen and M. Shader, eds, Series: Studies in Classification, Data Analysis, and Knowledge Organization, Springer-Verlag: Berlin, 2000, 409–415.

[7] L. Breiman, J.H. Friedman, R.A. Olshen and C.J. Stone, *Classification and Regression Trees*, Wadsworth Inc., Belmont, California, 1984.

[8] J. Catlett, *On changing continuous attributes into ordered discrete attributes*, in Proceedings of the European Working Session on Learning, 1991, 164–178.

[9] A. Ciampi, E. Diday, J. Lebbe, E. Périnel and R. Vignes, Recursive partitioning with probabilistically imprecise data, in: *Ordinal and Symbolic Data analysis*, E. Diday and Y. Lechevallier, eds, Series: Studies in Classification Data Analysis and Knowledge Organization, Springer-Verlag: Berlin, 1996, pp. 201–212.

[10] A. Ciampi, E. Diday, J. Lebbe, E. Périnel and R. Vignes, Growing a tree classifier with imprecise data, *Pattern Recognition Letters* **21**(9) (2000), 787–803.

[11] I. Csiszár, Information-type measures of difference of probability distributions and indirect observatons, *Studia Scient Math Hung* **2** (1967), 299–318.

[12] F.A.T. de Carvalho, Proximity coefficients between Boolean symbolic objects, in: *New Approaches in Classification and Data Analysis*, E. Diday et al., eds, Series: Studies in Classification, Data Analysis, and Knowledge Organization, 5, Springer-Verlag: Berlin, 1994, pp. 387–394.

[13] F.A.T. de Carvalho, Extension based proximity coefficients between constrained Boolean symbolic objects, in: *Proceedings of the 5th Conference of the International Federation of Classification Societies*, C. Hayashi et al., eds, Springer-Verlag: Berlin, 1998, pp. 370–378.

[14] E. Diday and F. Esposito, An introduction to Symbolic Data Analysis and the SODAS software, *Intelligent Data Analysis* **7**(6) (2003), 583–602, IOS Press.

[15] F. Esposito, D. Malerba and V. Tamma, Dissimilarity Measures for symbolic objects, in: *Analysis of Symbolic Data. Exploratory Methods for extracting Statistical Information from Complex Data*, H.H. Bock and E. Diday, eds, Series Studies in Classification, Data Analysis and Knowledge Organisation, 15, Springer-Verlag: Berlin, 2000, pp. 165–185.

[16] F. Esposito, D. Malerba and F.A. Lisi, Matching Symbolic Objects. Chapter 8.4 in H.-H. Bock and E. Diday (Eds.), *Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data*, Series: Studies in Classification, Data Analysis, and Knowledge Organization, 15, Springer-Verlag: Berlin, 2000, pp. 186–197.

[17] S. German, E. Bienenenstock and R. Doursat, Neural networks and the bias/variance dilemma, *Neural Computation* **4** (1992), 1–58.

[18] G. Gora and A. Wojna, RIONA: A Classifier Combining Rule Induction and k-NN Method with Automated Selection of Optimal Neighbourhood, in: *Proceedings of the 13th European Conference on Machine Learning*, T. Elomaa, H. Mannila and H.T.T. Toivonen, eds, Springer-Verlag: Berlin, 2430, 2002, pp. 111–123.

[19] K.C. Gowda and E. Diday, Symbolic clustering using a new dissimilarity measure, *Pattern Recognition* **24**(6) (1991), 567–578.

[20] M. Ichino and H. Yaguchi, Generalized Minkowski Metrics for Mixed Feature-Type Data Analysis, *IEEE Transactions on Systems, Man, and Cybernetics* **24**(4) (1994), 698–707.

[21] K. Kang and H. Sompolinsky, Mutual Information of Population Codes and Distance Measures in Probability Space, *Physical Review Letters* **86** (2001), 4958–4961.

[22] R. Kohavi, *A study of cross-validation and bootstrap for Accuracy estimation and model selection*, Proceedings of the 14th International Joint Conference on Artificial Intelligence, Morgan Kaufmann, 1995, 1137–1143.

[23] R.E. Krichevsky and V.K. Trofimov, The performance of universal encoding, *IEEE Transaction Information Theory* **IT-27** (1981), 199–207.

[24] S. Kullback and R.A. Leibler, On information and sufficiency, *Annals of Mathematical Statistics* **22** (1951), 76–86.

[25] C. Lauro, R. Verde and F. Palombo, Factorial Discriminant Analysis on Symbolic Objects, in: *Analysis of Symbolic Data: Exploratory Methods for Extracting Statistical Information from Complex Data*, H.H. Boch and E. Diday, eds, Series Studies in Classification Data Analysis and Knowledge Organization, 15, Springer-Verlag: Berlin, 2000, pp. 212–233.
[26] J. Lin, Divergence Measures Based on the Shannon Entropy, *IEEE Transactions on Information Theory* **37**(1) (1991), 145–151.
[27] M.C. Ludl and G. Widmer, Relative Unsupervised Discretization for Association Rule Mining, in: *Proceedings of the 4*th *European Conference on Principles of Data Mining and Knowledge Discovery*, D.A., Zighed, H.J. Komorowski and J.M. Zytkow, eds, Springer-Verlag: Berlin, 1910, 2000, 148–158.
[28] D. Malerba, F. Esposito, V. Gioviale and V. Tamma, Comparing Dissimilarity Measures for Symbolic Data Analysis, *Proceedings of Exchange of Technology and Know-how and New Techniques and Technologies for Statistics* **1** (2001), 473–481.
[29] D. Malerba, F. Esposito and M. Monopoli, Comparing dissimilarity measures for probabilistic symbolic objects, in: *Data Mining III*, (Vol. 6), A. Zanasi, C.A. Brebbia, N.F.F. Ebecken and P. Melli, eds, Series Management Information Systems, WIT Press, Southampton, UK, 2002, 31–40.
[30] D. Malerba, F. Esposito, F.A. Lisi and A. Appice, Mining spatial association rules in census data, *Research in Official Statistics* **5**(1) (2002), 19–44.
[31] D. Malerba, F. Esposito, M. Ceci and A. Appice, Top-Down Induction of model Trees with Regression and Splitting Nodes, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **26**(5) (2004), 612–625.
[32] T. Mitchell, *Machine Learning*, McGraw Hill, New York, USA, 1997.
[33] B. Öhman, *Statistics as an Investment – Free for Users*, Proceedings of the 4th International Conference on Methodological Issues in Official Statistics, Stockholm October, 2000, 12–13.
[34] M. Orkin and R. Drogin, *Vital Statistics*, McGraw Hill, New York, 1990.
[35] J.R. Quinlan, Induction of Decision Trees, *Machine Learning* **1** (1986), 81–106.
[36] Z. Rached, F. Alajaji and L.L. Campbell, Rényi's Divergence and Entropy Rates for Finite Alphabet Markov Sources, *IEEE Transactions on Information Theory* **47**(4) (2001), 1553–1561.
[37] J.P. Rasson, P. Lallemand and S. Adans, Bayesian Decision Tree: Discriminant Analysis on interval data and the Bayesian tree visualization by using the modules SBTREE and VTREE, *A Short Tutorial for Users*, Version 1, December 15, 2003.
[38] F. Rossi and B. Conan-Guez, Multi-layer Perceptron on Interval Data, in: *Classification, Clustering, and Data Analysis* (*IFCS 2002*), K. Jajuga, A. Sokolowski and H.H. Bock, eds, 2002, pp. 427–434.
[39] J.J.W. Sammon, A nonlinear mapping for data structure analysis, *IEEE Transactions on Computers* **C-18** (1969), 401–409.
[40] G. Saporta, *Data mining and official statistics*, Proceedings of the 5th Italian National Conference of Statistics, ISTAT, 2000.
[41] V. Stéphan, G. Hébrail and Y. Lechevallier, Generation of Symbolic Objects from Relational Databases, in: *Analysis of Symbolic Data. Exploratory Methods for extracting Statistical Information from Complex Data*, H.H. Bock and E. Diday, eds, Series: Studies in Classification, Data Analysis and Knowledge Organisation, 15, Springer-Verlag: Berlin, 2000, pp. 78–105.
[42] C.W. Therrien, *Decision, estimation and classification*, John Wiley & Sons, 1989.
[43] V.S. Verykios, E. Bertino, I.N. Fovino, L.P. Provenza, Y. Saygin and Y. Theodoridis, State-of-the-art in privacy preserving data mining, *SIGMOD Record* **33**(1) (2004), 50–57.
[44] D. Wetteschereck, *A study of Distance-Based Machine Learning Algorithms*, Doctor of Philosophy dissertation in Computer Science, Oregon State University.