8

Dissimilarity and matching

Floriana Esposito, Donato Malerba and Annalisa Appice

8.1 Introduction

The aim of symbolic data analysis (SDA) is to investigate new theoretically sound techniques by generalizing some standard statistical data mining methods to the case 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 (Bock and Diday, 2000). A variable Y is termed *set-valued* with domain $\mathcal{Y}$ if it takes its values in $\boldsymbol{P}(\mathcal{X}) = \{U|U \subseteq \mathcal{Y}\}$, that is, the power set of $\mathcal{X}$. When $X(k)$ is finite for each k, then Y is called *multi-valued*, while when an order relation $\prec$ is defined on $\mathcal{Y}$, then the value returned by a set-valued variable is expressed by an interval $[\alpha, \beta]$ and Y is termed an *interval* variable. A modal variable Y is a multi-valued variable with *weights*, such that $Y(k)$ describes both multi-valued data $U(k)$ and associated weights $\pi(k)$.

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*, which are more complex than those typically used in classical statistics. Indeed, the columns of a symbolic data table are called *symbolic variables*, while the rows correspond to distinct generalizations (or *symbolic descriptions*) describing a class of individuals that are in turn the partial or complete extent of a given concept.

Starting with a symbolic description, a *symbolic object* (SO) models the underlying concepts and provides a way to find at least the individuals of this class. In Bock and Diday (2000) and Diday and Esposito (2003), an SO is formally defined as a triple $s = (a, R, d)$, where R is a relation between descriptions (e.g., $R \in \{=, \equiv, \leq, \subseteq\}$ or R is an implication, a kind of matching), d is a description and a is a membership function defined by a set of

Symbolic Data Analysis and the SODAS Software Edited by E. Diday and M. Noirhomme-Fraiture

individuals Ω in a set L (e.g., $L = \{\text{true, false}\}$ or $L = [0, 1]$), such that a depends on R and d.

Many techniques for both the construction of SOs from records of individuals and the analysis of SOs are actually implemented in the ASSO Workbench. They deal with special classes of SOs, called *assertions*, where R is defined as $[d'Rd] = \wedge_{j=1,\ldots,p}[d'_j R_j d_j]$, with $\wedge$ as the standard logical conjunction operator and a is defined as $a(w) = [y(w) \quad R\ d]$, with $y(w) = y_1(w), \ldots, y_p(w)$ being the vector of variables describing w.

Dissimilarity and matching constitute one of the methods available in the ASSO Workbench. Several dissimilarity measures (DISS module) and matching functions (MATCH module) are implemented, enabling the user to compare symbolic data in a symbolic data table.

The DISS module implements the dissimilarity measures presented in Malerba *et al.* (2001, 2002). Henceforth, the dissimilarity measure d on a set of individuals E refers to a real-valued function on $E \times E$, such that $d_a^* = d(a, a) \leq d(a, b) = d(b, a) < \infty$ for all a, $b \in E$. In contrast, a similarity measure s on a set of objects E is a real-valued function on $E \times E$ such that $s_a^* = s(a, a) \geq s(a, b) = s(b, a) \geq 0$ for all a, $b \in E$. Generally, $d_a^* = d^*$ and $s_a^* = s^*$ for each object a in E, and more specifically, $d^* = 1$ when $s^* = 0$ (Batagelj and Bren, 1995).

Since the similarity comparison can be derived by transforming a dissimilarity measure into a similarity one[1], only dissimilarity measures are actually implemented in the DISS module.

The MATCH module performs a *directional* (or asymmetric) comparison between the SOs underlying symbolic descriptions stored in a symbolic data table, in order to discover their linkedness or differences. Notice that, while the dissimilarity measures are defined for symbolic descriptions and do not consider how the extent of corresponding SOs is effectively computed according to R, the matching comparison is performed at the level of SOs by interpreting R as a matching operator.

This matching comparison has a *referent* and a *subject* (Patterson, 1990). The former represents either a prototype or a description of a class of individuals, while the latter is either a variant of the prototype or an instance (individual) of a class of objects. In its simplest form, matching compares referent and subject only for equality, returning false when they fail in at least one aspect, and true otherwise. In more complex cases, the matching process performs the comparison between the description of a class C (subject of matching) and the description of some unit u (referent of matching) in order to establish whether the individual can be considered an instance of the class (inclusion requirement). However, this requirement of equality (*canonical matching*), even in terms of inclusion requirement, is restrictive in real-world problems because of the presence of noise, the imprecision of measuring instruments or the variability of the phenomenon described by the referent of matching. This makes it necessary to rely on a relaxed definition of matching that aims to compare two SOs in order to identify their similarities rather than to establish whether they are equal. The result is a *flexible matching* function with a range in the interval [0,1] that indicates the probability of precisely matching the subject with the referent, provided that some change is possibly made in the description of the referent. It is interesting to note

[1] This transformation is made possible by preserving properties defined with the induced quasi-ordering and imposing $d := \phi(s)$ and $s := \psi(d)$, respectively, where $\phi(\cdot)$ and $\psi(\cdot)$ are strictly decreasing functions with boundary conditions (e.g. $\phi(0) = 1$ and $\phi(1) = 0$, or $\phi(0) = 1$ and $\phi(\infty) = 0$).

that even flexible matching is not a dissimilarity measure, due to the non-symmetry of the matching function.

Both DISS and MATCH input a symbolic data table stored in an ASSO file and output a new ASSO file that includes the same input data, in addition to the matrix of the results of dissimilarity (or matching) computation.

More precisely, the dissimilarity (or matching) value computed between the ith symbolic description (SO) and the jth symbolic description (SO) taken from the input symbolic data table is written in the (i, j)th cell (entry) of the output dissimilarity (matching) matrix. The main difference is that the dissimilarity matrix is stored as a lower triangular matrix, since the upper values $(i < j)$ can be derived from the symmetry property of dissimilarity. Conversely, the matching matrix is stored as a sparse square matrix, due to the non-symmetry of the matching function.

Finally, the ASSO Workbench supports users in choosing the list of symbolic variables forming the symbolic descriptions (SOs) to be compared, the dissimilarity measure or matching function to be computed as well as some related parameters.

In this chapter, the use of DISS for the computation of dissimilarity measures is illustrated together with the VDISS module for the visualization of the dissimilarities by means of two-dimensional scatterplots and line graphs. The explanation of the outputs and the results of MATCH for the computation of the matching functions are given at the end of the chapter.

8.2 Input data

The main input of the dissimilarity and matching method is an ASSO file describing a symbolic data table, whose columns correspond to either set-valued or probabilistic symbolic variables. Each row represents the symbolic description d of an individual of E. The ASSO Workbench allows users to select one or more symbolic variables associated with the columns of the symbolic data table stored in the input ASSO file.

Statistical metadata concerning the source agencies of symbolic data, collection information, statistical populations, original variables and standards, symbolic descriptions and symbolic variables, logistical metadata and symbolic analysis previously performed on the data may be available in the metadata file, meta<ASSO file name>.xml, stored in the same directory as the input ASSO file. Notice that metadata information is not manipulated by the dissimilarity and matching method, but it is simply updated by recording the last dissimilarity measure or matching function computed on such data and the list of symbolic variables involved in the comparison.

8.3 Dissimilarity measures

Several methods have been reported in the literature for computing the dissimilarity between two symbolic descriptions d_a and d_b (Malerba *et al.*, 2001, 2002). In the following, we briefly describe the dissimilarity measures implemented in the DISS module for two kinds of symbolic descriptions, named *boolean* and *probabilistic*. The former are described by set-valued variables, while the latter are described by probabilistic variables, that is, modal variables describing frequency distributions. SOs underlying boolean and probabilistic symbolic descriptions are referred to as *boolean symbolic objects* (BSOs) and *probabilistic symbolic objects* (PSOs), respectively.

Mixed symbolic descriptions, that is, symbolic descriptions described by both set-valued and probabilistic variables, are treated by first separating the boolean part from the probabilistic part and then computing dissimilarity values separately for these parts. Dissimilarity values obtained by comparing the boolean and probabilistic parts respectively are then combined by sum or product.

8.3.1 Dissimilarity measures for boolean symbolic descriptions

Let d_a and d_b be two symbolic descriptions described by m set-valued variables Y_i with domain Y_i. Let $A_i(B_i)$ be the set of values (subset of Y_i) taken from Y_i in d_a (d_b). A class of dissimilarity measures between d_a and d_b is defined by aggregating dissimilarity values computed independently at the level of single variables Y_i (*componentwise dissimilarities*). A classical aggregation function is the Minkowski metric (or L_q distance) defined on $\mathbb{R}^m$. Another class of dissimilarity measures is based on the notion of *description potential* $\pi(d_a)$ of a symbolic description d_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 implemented in DISS are reported in Table 8.1 together with their short identifier used in the ASSO Workbench. They are:

- Gowda and Diday's dissimilarity measure (U_1: Gowda and Diday, 1991);
- Ichino and Yaguchi's first formulation of a dissimilarity measure (U_2: Ichino and Yaguchi, 1994);
- Ichino and Yaguchi's normalized dissimilarity measure (U_3: Ichino and Yaguchi, 1994);
- Ichino and Yaguchi's normalized and weighted dissimilarity measure (U_4: Ichino and Yaguchi, 1994);
- de Carvalho's normalized dissimilarity measure for constrained[2] Boolean descriptions (C_1: de Carvalho, 1998);
- de Carvalho's dissimilarity measure (SO_1: de Carvalho, 1994);
- de Carvalho's extension of Ichino and Yaguchi's dissimilarity (SO_2: de Carvalho, 1994);
- de Carvalho's first dissimilarity measure based on description potential (SO_3: de Carvalho, 1998);
- de Carvalho's second dissimilarity measure based on description potential (SO_4: de Carvalho, 1998);

[2] The term *constrained Boolean descriptions* 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 which are 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 outside the scope of this paper, nevertheless it is always possible to apply the measures defined for constrained boolean descriptions to unconstrained boolean descriptions.

- de Carvalho's normalized dissimilarity measure based on description potential (SO_5: de Carvalho, 1998);
- a dissimilarity measure based on flexible matching between BSOs (SO_6).

The last measure (SO_6) differs from the others, since its definition is based on the notion of flexible matching (Esposito *et al.*, 2000), which is an asymmetric comparison. The dissimilarity measure is obtained by means of a symmetrization method that is common to measures defined for probabilistic symbolic descriptions.

The list of dissimilarity measures actually implemented in DISS cannot be considered complete. For instance, some clustering modules of the ASSO Workbench (e.g., NBCLUST and SCLUST; see Chapter 14) estimate the dissimilarity value between two boolean symbolic descriptions d_a and d_b as follows:

$$d(d_a, d_b) = \left(\sum_{i=1}^{m} \delta_i^2(A_i, B_i) \right)^{1/2},$$

where δ_i denotes a dissimilarity index computed on each pair (A_i, B_i). In particular, if Y_i is an interval variable, we have that $A_i = [a_{i,\inf}, a_{i,\sup}]$ and $B_i = [b_{i,\inf}, b_{i,\sup}]$. In this case, $\delta_i(A_i, B_i)$ is computed in terms of:

- the Hausdorff distance defined by

$$\delta_i([a_{i,\inf}, a_{i,\sup}], [b_{i,\inf}, b_{i,\sup}]) = \max\{|a_{i,\inf} - b_{i,\inf}|, |a_{i,\sup} - b_{i,\sup}|\};$$

- the L_1 distance defined by:

$$\delta_i([a_{i,\inf}, a_{i,\sup}], [b_{i,\inf}, b_{i,\sup}]) = |a_{i,inf} - b_{i,inf}| + |a_{i,sup} - b_{i,sup}|;$$

- the L_2 distance defined by:

$$\delta_i([a_{i,\inf}, a_{i,\sup}], [b_{i,\inf}, b_{i,\sup}]) = (a_{i,\inf} - b_{i,\inf})^2 + (a_{i,\sup} - b_{i,\sup})^2.$$

On the other hand, if Y_i is a multi-valued variable that takes its values in the power set of Y_i (i.e., $\boldsymbol{P}(\mathcal{Y}_i) = \{U | U \subseteq \mathcal{Y}_i\}$), the dissimilarity index δ_i is computed by estimating the difference in the frequencies of each category value taken by Y_i .

Denoting by p_i the number of categories in Y_i ($p_i = |U_i|$, with U_i the range of Y_i), the frequency value $q_{i,U_i}(c_s)$ associated with the category value c_s ($s = 1, \ldots, p_i$) of the variable $Y_i = U_i(c_s \in U_i)$ is given by

$$q_{i,U_i}(c_s) = \begin{cases} \frac{1}{|U_i|}, & \text{if } c_s \in U_i, \\ 0, & \text{otherwise.} \end{cases}$$

Therefore, the symbolic descriptions d_a and d_b can be transformed as follows:

$$d_a = ((q_{1,A_1}(c_1), \ldots, q_{1,A_1}(c_{p_1})), \ldots, (q_{m,A_m}(c_1), \ldots, q_{m,A_m}(c_{p_m}))),$$
$$d_b = ((q_{1,B_1}(c_1), \ldots, q_{1,B_1}(c_{p_1})), \ldots, (q_{m,B_m}(c_1), \ldots, q_{m,B_m}(c_{p_m}))),$$

Table 8.1 Dissimilarity measures defined for Boolean symbolic descriptions.

Name	Componentwise dissimilarity measure	Objectwise dissimilarity measure
U_1	$D^{(i)}(A_i, B_i) = D_\pi(A_i, B_i) + D_s(A_i, B_i) + D_c(A_i, B_i)$ 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*.	$\sum_{i=1}^{m} D^{(i)}(A_i, B_i)$
U_2	$\phi(A_i, B_i) = \|A_i \oplus B_i\| - \|A_i \otimes B_i\| + \gamma(2\|A_i \otimes B_i\| - \|A_i\| - \|B_i\|)$ with *meet* ($\otimes$) and *join* ($\oplus$)Cartesian operators.	$\sqrt[q]{\sum_{i=1}^{m} [\phi\,(A_i, B_i)]^q}$
U_3	$\psi(A_i, B_i) = \dfrac{\phi(A_i, B_i)}{\|X_i\|}$	$\sqrt[q]{\sum_{i=1}^{m} [\psi\,(A_i, B_i)]^q}$
U_4	$\psi(A_i, B_i) = \dfrac{\phi(A_i, B_i)}{\|X_i\|}$	$\sqrt[q]{\sum_{i=1}^{m} w_i\,[\psi\,(A_i, B_i)]^q}$
C_1	$D_1(A_\mathrm{i}, B_i) = 1 - \alpha/(\alpha + \beta + \chi)$ $D_2(A_\mathrm{i}, B_i) = 1 - 2\alpha/(2\alpha + \beta + \chi)$ $D_3(A_\mathrm{i}, B_i) = 1 - \alpha/(\alpha + 2\beta + 2\chi)$ $D_4(A_\mathrm{i}, B_i) = 1 - \frac{1}{2}\left(\frac{\alpha}{\alpha+\beta} + \frac{\alpha}{\alpha+\chi}\right)$ $D_5(A_\mathrm{i}, B_i) =$ $1 - \alpha/\sqrt{(\alpha + \beta)(\alpha + \chi)}$ with $\chi = \mu(c(A_i) \cap B_i)$ $\alpha = \mu(A_i \cap B_i);\ \beta = \mu(A_i \cap c(B_i))$	$\sqrt[q]{\dfrac{\sum_{i=1}^{m} [w_i D_r\,(A_i, B_i)]^q}{\sum_{i=1}^{m} \delta(i)}}$, where $\delta(i)$ is the indicator function

	For each subset $V_j \subseteq Y_i \mu(V_j) = \lvert V_j \rvert$ if Y_j is integer, nominal or ordinal and $\mu(V_j) = \lvert a - b \rvert$ if Y_j is continuous and $V_j = [a - b]$. $c(V_j)$ denotes the complementary set of V_j in the domain Y_i.	
SO_1		$\sqrt[q]{\sum_{i=1}^{m} [w_i D_r (A_i, B_i)]^q}$
SO_2	$\psi'(A_i, B_i) = \dfrac{\phi(A_i, B_i)}{\mu(A_i \oplus B_i)}$	$\sqrt[q]{\sum_{i=1}^{m} \frac{1}{m} [\psi'(A_i, B_i)]^q}$
SO_3	none	$\pi(d_a \oplus d_b) - \pi(d_a \otimes d_b)$ $+\gamma(2\pi(d_a \otimes d_b) - \pi(a) - \pi(b))$ where meet ($\otimes$) and join ($\oplus$) are Cartesian operators defined on BSOs.
SO_4	none	$\dfrac{\pi(d_a \oplus d_b) - \pi(d_a \otimes d_b) + \gamma(2\pi(d_a \otimes d_b) - \pi(d_a) - \pi(d_b))}{\pi(d_a^E)}$
SO_5	none	$\dfrac{\pi(d_a \oplus d_b) - \pi(d_a \otimes d_b) + \gamma(2\pi(d_a \otimes d_b) - \pi(d_a) - \pi(d_b))}{\pi(d_a \oplus d_b)}$
SO_6	none	$1 - [FlexMatch(a, b) + FlexMatch(b, a)]/2$ where *FlexMatch* denotes the flexible matching function, while a and b are the BSOs in the form of assertions underlying the descriptions d_a and d_b, respectively.

such that $\sum_{j=1}^{m_i} q_{i,A_i}(c_j)=1$ and $\sum_{j=1}^{m_i} q_{i,B_i}(c_j)=1$, for all $i\in\{1,\ldots,m\}$. Hence the dissimilarity index $\delta_i(A_i, B_i)$ is computed in terms of:

- the L_1 distance defined by

$$\delta_i(A_i, B_i)=\sum_{j=1}^{|Y_i|}|q_{i,A_i}(c_j)-q_{i,B_i}(c_j)|;$$

- the L_2 distance defined by

$$\delta_i(A_i, B_i)=\sum_{j=1}^{|Y_i|}(q_{i,A_i}(c_j)-q_{i,B_i}(c_j))^2;$$

- the de Carvalho distance defined by

$$\delta_i(A_i, B_i)=\sum_{j=1}^{|Y_i|}(\gamma q_{i,A_i}(c_j)+\gamma' q_{i,B_i}(c_j))^2,$$

  where

$$\gamma=\begin{cases}1, & \text{if } c_j\in A_i\wedge c_j\notin B_i,\\ 0, & \text{otherwise,}\end{cases}$$

$$\gamma'=\begin{cases}1, & \text{if } c_j\notin A_i\wedge c_j\in B_i,\\ 0, & \text{otherwise.}\end{cases}$$

These dissimilarity measures will be implemented in an extended version of the DISS module.

8.3.2 Dissimilarity measures for probabilistic symbolic descriptions

Let d_a and d_b be two probabilistic symbolic descriptions and Y a multi-valued modal variable describing them. The sets of probabilistically weighted values taken by Y in d_a and d_b define two discrete probability distributions P and Q, whose comparison allows us to assess the dissimilarity between d_a and d_b on the basis of Y 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 $\mathcal{Y}$ is $=$ {red, white, black}. Therefore, the dissimilarity between two probabilistic symbolic descriptions described by p symbolic probabilistic 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 Csiszár (1967) and Ali and Silvey (1996). Some well-known members of this family are as follows:

- The *Kullback–Leibler* (KL) *divergence*, which is a measure of the difference between two probability distributions (Kullback and Leibler, 1951). This is defined as $m_{\mathrm{KL}}(P, Q) := \Sigma_{x \in X} q(x) \log(q(x)/p(x))$ and measures to what extent the distribution P is an approximation of the distribution Q. It is asymmetric, that is, $m_{\mathrm{KL}}(P, Q) \neq m_{\mathrm{KL}}(Q, P)$ 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 χ^2 *divergence*, defined as $m_{\chi^2}(P, Q) := \Sigma_{x \in X} |p(x) - q(x)|^2/p(x)$, is strictly topologically stronger than the KL divergence, since the inequality $m_{\mathrm{KL}}(P, Q) \leq m_{\chi^2}(P, Q)$ holds, i.e. the convergence in χ^2 divergence implies convergence in the KL divergence, but the converse is not true (Beirlant *et al.*, 2001). 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 order s* is derived as follows:

  $$m_{\mathrm{C}}^{(s)}(P, Q) := -\log m^{(s)}(P, Q).$$

  This distance diverges only when the two distributions have zero overlap, that is, the intersection of their support is empty (Kang and Sompolinsky, 2001).

- *Rényi's divergence* (or *information gain*) of order s between two probability distributions P and Q is given by $m_{\mathrm{R}}{}^{(s)}(P, Q) := -\log \ m^{(s)}(P, Q)/(s-1)$. It is noteworthy that, as $s \to 1$, Rényi's divergence approaches the KL divergence (Rached *et al.*, 2001).

- The *variation distance*, given by $m_1(P, Q) := \Sigma_{x \in X} |p(x) - q(x)|$, is also known as the *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 $m_2(P, Q) := \Sigma_{x \in X} |p(x) - q(x)|^2$ and, more generally, 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 *Kullback divergence* is given by $m_{\mathrm{K}}(P, Q) := \Sigma_{x \in X} q(x) \log(q(x)/(1/2p(x) + 1/2q(x)))$ (Lin, 1991), which is an asymmetric measure. It has been proved that the Kullback divergence is upper bounded by the variation distance $m_1(P, Q) : m_{\mathrm{K}}(P, Q) \leq m_1(P, Q) \leq 2$.

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 the *J-coefficient* (or *J-divergence*), can be defined as follows:

$$J(P,Q) \ := m_{\mathrm{KL}}(P,Q) + m_{\mathrm{KL}}(Q,P).$$

Alternatively, many authors have defined the *J-divergence* as the average rather than the sum $J(P,Q) := (m_{\mathrm{KL}}(P,Q) + m_{\mathrm{KL}}(Q,P))/2$. 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 probabilistic symbolic descriptions d_a and d_b by aggregation through the generalized and weighted Minkowski metric:

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

where $\forall k \in \{1,\dots,m\}, c_k > 0$ are weights with $\Sigma_{k=1\dots m} c_k = 1$ and $m(A_i,B_i)$ is either the Minkowski L_p distance (LP) or a symmetrized version of the J-coefficient (J), χ^2 divergence (CHI2), Rényi's distance (REN), or Chernoff's distance (CHER). These are all variants of the dissimilarity measure denoted by P_1 in the ASSO Workbench. Notice that the Minkowski L_p distance, the J-coefficient, the χ^2 divergence, Rényi's distance and Chernoff's distance require no category of a probabilistic symbolic variable in a probabilistic symbolic description to be associated with a zero probability. To overcome these limitations, symbolic descriptions may be generated by using the *KT estimate* when estimating the probability distribution, in order to prevent the assignments of a zero probability to a category. 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:

$$p(x) = \frac{(\text{No. of times } x \text{occurs in } \{R_1,\dots,R_M\}) + 1/2}{M + (K/2)},$$

where x is the category of the modal symbolic variable, $\{R_1,\dots,R_M\}$ 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 (Krichevsky and Trofimov, 1981).

The dissimilarity coefficients can also 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 probabilistic symbolic descriptions:

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

where each x_i corresponds to a value of the ith variable domain.

Table 8.2 Dissimilarity measures defined for Probabilistic symbolic descriptions.

Name	Componentwise dissimilarity measure	Objectwise dissimilarity measure
P_1	Either $m_p(P, Q)$ or a symmetrized version of $m_{\mathrm{KL}}(P, Q), m_{\chi}^2(P, Q), m_{\mathrm{C}}^{(s)}(P, Q), m_{\mathrm{R}}^{(s)}(P, Q)$	$\sqrt[p]{\sum_{i=1}^{m}[c_i m(A_i, B_i)]^p}$
P_2	$m_p(P, Q)$	$1-\frac{\prod_{i=1}^{m}\left(\sqrt[p]{2}-\sqrt[p]{m_p(A_i, B_i)}\right)}{\left(\sqrt[p]{2}\right)^m}$
P_3	none	$1-[FlexMatch(a, b)+FlexMatch(b, a)]/2$, where *FlexMatch* denotes the flexible matching function, while a and b are the PSOs in the form of assertions representing the descriptions d_a and d_b, respectively.

Note that this dissimilarity measure, denoted as P_2 in the ASSO Workbench, is symmetric and normalized in [0,1]. Obviously $d'_p(d_a, d_b)=0$ if d_a and d_b are identical and $d'_p(d_a, d_b)=1$ if the two symbolic descriptions are completely different.

Alternatively, the dissimilarity measure between two probabilistic dissimilarity descriptions d_a and d_b can be computed by estimating both the matching degree between the corresponding PSOs a and b and vice versa. The measure denoted as P_3 in the ASSO Workbench extends the SO_6 measure defined for BSOs. A summary of the three dissimilarity measures, defined on probabilistic symbolic descriptions, is reported in Table 8.2.

As already observed for the boolean case, the list of dissimilarity measures implemented in DISS for PSOs is not exhaustive. Some clustering modules of the ASSO Workbench (e.g., NBCLUST and SCLUST; see Chapter 14) implement a further dissimilarity measure that estimates the dissimilarity between two probabilistic symbolic descriptions by composing the values of dissimilarity indices δ_i as follows:

$$d(d_a, d_b)=\left(\sum_{i=1}^{m}\delta_i^2((A_i, \pi_{A_i}), (B_i, \pi_{B_i}))\right)^{1/2}.$$

In this case, the dissimilarity index $\delta_i((A_i, \pi_{A_i}), (B_i, \pi_{B_i}))$ is computed in terms of:

- the L_1 distance defined by

$$\delta_i((A_i, \pi_{A_i}), (B_{\mathrm{i}}, \pi_{B_i}))=\sum_{j=1}^{|Y_i|}|\pi_{A_i}(c_j)-\pi_{B_i}(c_j)|;$$

- the L_2 distance defined by

$$\delta_i(A_i, B_i) = \sum_{j=1}^{|Y_i|} (\pi_{A_i}(c_j) - \pi_{B_i}(c_j))^2;$$

- the de Carvalho distance defined by

$$\delta_i(A_i, B_i) = \sum_{j=1}^{|Y_i|} (\gamma \pi_{A_i}(c_j) + \gamma' \pi_{B_i}(c_j))^2,$$

  where γ and γ' are defined as before.

Also in this case we plan to implement these additional dissimilarity measures for PSOs in a new release of the DISS module.

8.4 Output of DISS and its visualization

The DISS module outputs a new ASSO file that includes both the input symbolic data table D and the dissimilarity matrix M resulting from the computation of the dissimilarity between each pair of symbolic descriptions from D. This means that $M(i, j)$ corresponds to the dissimilarity value computed between the ith symbolic description and the jth symbolic description taken from D. Since dissimilarity measures are defined as symmetric functions, M is a symmetric matrix with $M(i, j) = M(j, i)$. However, due to computation issues, M is effectively computed as a lower triangular matrix, where dissimilarity values are undefined for upper values $(i < j)$ of $M(i, j)$. In fact, upper values can be obtained without effort by exploiting the symmetry property of dissimilarity measures. In addition, DISS produces a report file that is a printer-formatted file describing both the input parameters and the matching matrix. When a metadata file is associated with the input ASSO file, DISS updates the metadata by recording the dissimilarity measure and the list of symbolic variables considered when computing the dissimilarity matrix in question.

Both the dissimilarity matrix and the dissimilarity metadata can be obtained by means of the ASSO module VDISS. More precisely, VDISS outputs the matrix M in either a *table format*, a two-dimensional *scatterplot* or *graphic* representation.

The table format visualization shows the dissimilarity matrix M as a symmetric matrix, where both rows and columns are associated with the individuals whose symbolic descriptions are stored in the symbolic data table stored in the input ASSO file. Although M is computed as a lower triangular matrix, undefined upper values of M ($M(i, j)$ with $i < j$), are now explicitly stated by imposing $M(i, j) = M(j, i)$.

Moreover, several properties can be checked on the dissimilarity matrix: the *definiteness* property,

$$M(i, j) = 0 \Rightarrow i = j, \quad \forall i, j = 1, \ldots, n;$$

the *evenness* property,

$$M(i, j) = 0 \Rightarrow M(i, k) = M(j, k), \quad \forall k = 1, \ldots, n;$$

the *pseudo-metric* or *semi-distance*,

$$M(i,j) \leq M(i,k) + M(k,j), \quad \forall i,j,k = 1,\ldots,n;$$

the *Robinsonian* property, by which, for each $k = 1,\ldots,n$, we have that

$$\begin{aligned} M(k,k) &\leq M(k,k+1) \leq \ldots \leq M(k,n-1) \leq M(k,n) \wedge M(k,k) \\ &\leq M(k,k-1) \leq \ldots \leq M(k,2) \leq M(k,1), \\ M(k,k) &\leq M(k+1,k) \leq \ldots \leq M(n-1,k) \leq M(n,k) \wedge M(k,k) \\ &\leq M(k-1,k) \leq \ldots \leq M(2,k) \leq M(1,k); \end{aligned}$$

Buneman's inequality,

$$M(i,j) + M(h,k) \leq \max\{M(i,h) + M(j,k), M(i,k) + M(j,h)\} \quad \forall i,j,h,k = 1,\ldots,n;$$

and, finally, the *ultrametric* property,

$$M(i,j) \leq \max\{M(i,k), M(k,j)\} \quad \forall i,j,k = 1,\ldots,n.$$

The two-dimensional scatterplot visualization is based on the non-linear mapping of symbolic descriptions stored in the input ASSO file and points of a two-dimensional space. This non-linear mapping is based on an extension of Sammon's algorithm (Sammon, 1969) that takes as input the dissimilarity matrix M and returns a collection of points in the two-dimensional space (visualization area), such that their Euclidean distances preserve the 'structure' of the original dissimilarity matrix.

Scatterplot visualization supports both scrolling operations (left, right, up or down) as well as zooming operations over the scatterplot area. For each point in the scatterplot area, the user can also display the (X, Y) coordinates as well as the name (or label) of the individual represented by the point.

The dissimilarity matrix M can also be output graphically in the form of a partial or total line, bar and pie chart. In line chart based output, dissimilarity values are reported along the vertical axis, while individual identifiers (labels or names) are reported on the horizontal axis. For each column j of M, a different line is drawn by connecting the set of points $P(i,j)$ associated with the $M(i,j)$ value. In particular, the (X, Y) coordinates of the point $P(i,j)$ represent the individual on the ith row of M and the dissimilarity value stored in the $M(i,j)$ cell, respectively. The main difference between a partial line chart and a total line chart is that the former treats M as a lower triangular matrix and draws a line for each column j of M by ignoring points whose ordinate value is undefined in M (i.e. $i < j$), while the latter treats M as a symmetric matrix and derives undefined values by exploiting the symmetry property of the dissimilarity measures.

Both partial and total line charts can be visualized in a two- or three-dimensional space. Dissimilarity values described with total line charts can also be output as bar or pie charts.

Finally, a report including the list of variables and the dissimilarity measures adopted when computing M can be output in a text box.

8.5 An Application of DISS

In this section, we show the use of both the DISS module for the computation of a dissimilarity matrix from a symbolic data table stored in an input ASSO file and the VDISS module for the visualization of dissimilarities by means of two-dimensional scatterplots and line graphs. For this purpose, we present a case study of the analysis of the symbolic data table stored in the ASSO file enviro.xml that contains symbolic descriptions of 14 individuals generated by DB2SO.

Symbolic data are extracted by generalizing the data derived from a survey conducted by Statistics Finland. The population under analysis is a sample of 2500 Finnish residents aged between 15 and 74 in December 2000. Data are collected by interview and missing values are imputed by logistic regression. The survey contains 46 questions, but only 17 questions representing both continuous and categorical variables are selected as independent variables for symbolic data generation. Symbolic data are constructed by Cartesian product among three categorical variables (grouping variables): gender (M, F), age ([15–24], [25–44], [45–64], [65–74]) and urbanicity (very urban and quite urban).[3] Statistical metadata concerning information about the sample Finnish population analysed for the survey, the original variables, the symbolic descriptions and the symbolic variables are stored in metaenviro.xml.

Starting from the enviro symbolic data, a new ASSO chain named enviro is created by selecting Chaining from the main (top-level) menu bar and clicking on New chaining or typing Ctrl-N. The base node is associated with the ASSO file enviro.xml and a new empty block is added to the running chain by right-clicking on the Base block and choosing Insert method from the pop-up menu. The DISS module is selected from Dissimilarity and Matching in the Methods drop-down list and dragged onto the empty block (see Figure 8.1).

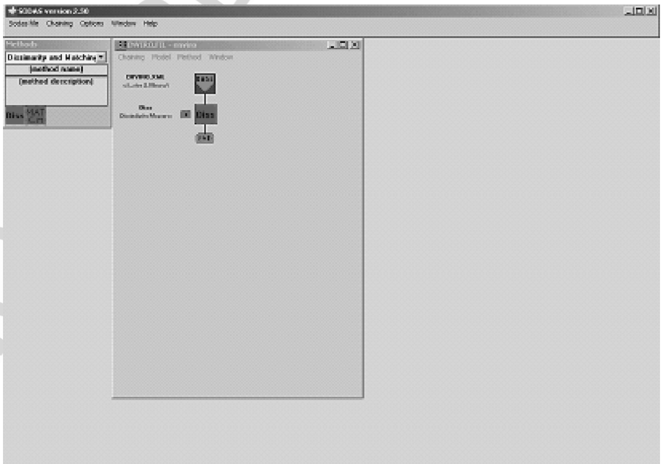
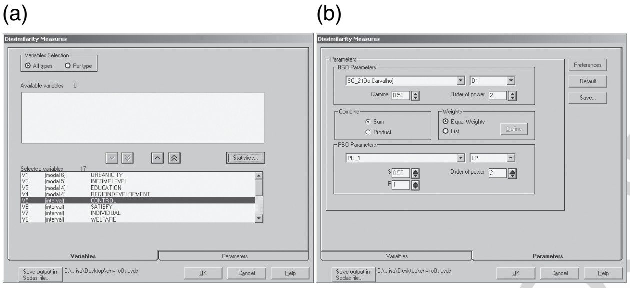


Figure 8.1 An example of the ASSO chain.

[3] The enviro.xml ASSO file contains the symbolic descriptions of only 14 individuals instead of 16. This is due to the fact that no enviro micro-data fall in two of the grouping sets obtained by DB2SO when aggregating enviro micro-data with respect to the 'gender–age–urbanicity' attributes.



(a)

Variables Selection: ⦿ All types ○ Per type

Available variables 0

Selected variables 17

V1	(modal 6)	URBANICITY
V2	(modal 5)	INCOMELEVEL
V3	(modal 4)	EDUCATION
V4	(modal 4)	REGIONDEVELOPMENT
V5	(interval)	CONTROL
V6	(interval)	SATISFY
V7	(interval)	INDIVIDUAL
V8	(interval)	WELFARE

C:\...isa\Desktop\enviroOut.sds

(b)

Parameters

BSO Parameters: SO_2 (De Carvalho) D1
Gamma 0.50 Order of power 2

Combine: ⦿ Sum ○ Product

Weights: ⦿ Equal Weights ○ List

PSO Parameters: PU_1 LP
S 0.50 Order of power 2
P 1

C:\...isa\Desktop\enviroOut.sds

Figure 8.2 (a) List of variables and (b) dissimilarity measures selected when computing the dissimilarity matrix from the symbolic descriptions stored in the enviro ASSO file.

This chain is now able to compute and output the dissimilarity matrix representing dissimilarity values between each pair of symbolic descriptions stored in the enviro symbolic data table.

Before computing the dissimilarity matrix, users must choose the list of symbolic variables to be involved in computing dissimilarities, the dissimilarity measure(s) to be used, the name of the output ASSO file, and so on. Both the list of symbolic variables and the dissimilarity measures are chosen by selecting Parameters . . . from the pop-up menu associated with the DISS block in the chain. The list of symbolic variables taken from the input symbolic data table is shown in a list box and users choose the variables to be considered when computing dissimilarity (see Figure 8.2(a)).

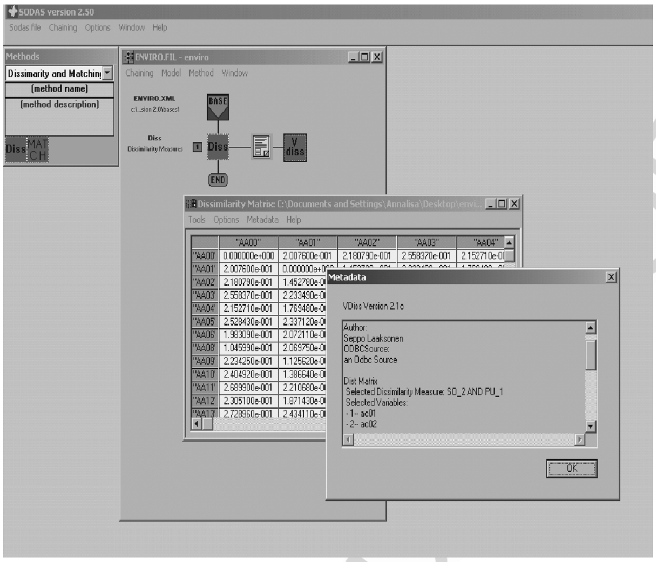
For each symbolic variable some statistics can be output, namely, the minimum and maximum for continuous (single-valued or interval) variables and the frequency distribution of values for categorical (single-valued or multi-valued) variables.

By default, all the symbolic variables are available for selection without any restriction on the type. However, users may decide to output only a subset of the variables taken from the input symbolic data by filtering on the basis of the variable type. Whenever users select only set-valued (probabilistic) variables, symbolic descriptions to be compared are treated as boolean (probabilistic) data. Conversely, when users select probabilistic variables in addition to set-valued variables, symbolic descriptions to be compared are treated as mixed data.

In this application, let us select all the symbolic variables (13 interval variables and four probabilistic variables) from the enviro data. This means that symbolic descriptions considered for dissimilarity computation are mixed data, where it is possible to separate the boolean part from the probabilistic part. Users have to choose a dissimilarity measure for the boolean part and a dissimilarity measure for the probabilistic part and to combine the result of computation by either sum or product (see Figure 8.2(b)).

In this example, we choose the dissimilarity measures SO_2 to compare the boolean parts and P_1(LP) to compare the probabilistic parts of the enviro symbolic descriptions. Equal weights are associated with the set-valued variables, while dissimilarity values obtained by comparing boolean parts and probabilistic parts are combined in an additive form.

When all these parameters are set, the dissimilarity matrix can be computed by choosing Run method from the pop-up menu associated with the DISS block in the running chain. DISS produces as output a new ASSO file that is stored in the user-defined path and includes



	"AA00"	"AA01"	"AA02"	"AA03"	"AA04"
"AA00'	0.000000e+000	2.007600e-001	2.180790e-001	2.558370e-001	2.152710e-00
"AA01'	2.007600e-001	0.000000e+0			
"AA02'	2.180790e-001	1.452780e-0			
"AA03'	2.558370e-001	2.233490e-0			
"AA04'	2.152710e-001	1.769480e-0			
"AA05'	2.528430e-001	2.337120e-0			
"AA06'	1.983090e-001	2.072110e-0			
"AA08'	1.045990e-001	2.069750e-0			
"AA09'	2.234250e-001	1.125620e-0			
"AA10'	2.404920e-001	1.386640e-0			
"AA11'	2.689900e-001	2.210680e-0			
"AA12'	2.305100e-001	1.871430e-0			
"AA13'	2.728960e-001	2.434110e-0			

VDiss Version 2.1c

Author:
Seppo Laaksonen
ODBCSource:
an Odbc Source

Dist Matrix
Selected Dissimilarity Measure: SO_2 AND PU_1
Selected Variables:
- 1-- ac01
- 2-- ac02

Figure 8.3 The table format output of the dissimilarity matrix and the dissimilarity metadata output of DISS on symbolic descriptions stored in the enviro.xml ASSO file.

both the symbolic data table stored in the input ASSO file and the dissimilarity matrix computed by DISS.

The metadata file is updated with information concerning the dissimilarity measures and the symbolic variables involved in the dissimilarity computation.

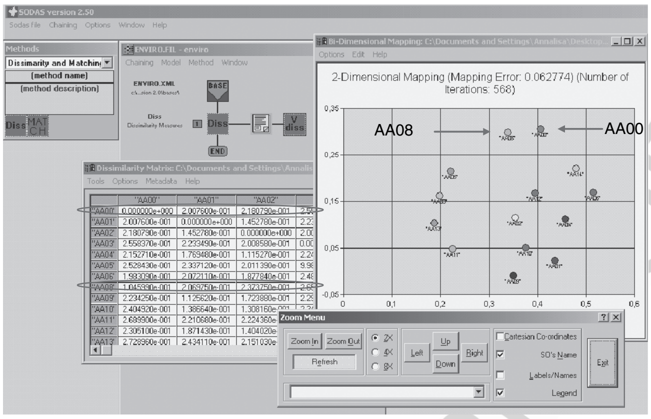
When the dissimilarity matrix is correctly constructed, the dissimilarity matrix is stored in the output ASSO file, in addition to the input symbolic data table. Moreover, the running chain is automatically extended with a yellow block (report block) that is directly connected to the DISS block. The report block allows users to output a report that describes the symbolic variables and the dissimilarity measures selected for the dissimilarity computation, as well as the lower triangular dissimilarity matrix computed by DISS. A pop-up menu associated with the report block allows users to either output this report as a printer-formatted file by selecting Open. . . and then View Result Report . . . or remove the results of the DISS computation from the running chain by selecting Delete results . . . from the menu in question.

Moreover, a red block connected to the yellow one is introduced in the running chaining. This block is automatically associated with the VDISS module and allows users to output both the dissimilarity matrix and the dissimilarity metadata (see Figure 8.3).

Table format output is shown by selecting Open. . . from the pop-up menu associated with the VDISS block of the running chain.

VDISS allows users to plot the symbolic descriptions taken from the enviro ASSO file as points on a two-dimensional scatterplot such that the Euclidean distance between the points preserves the dissimilarity values computed by DISS.

Notice that opting for a scatterplot highlights the existence of three clusters of similar symbolic descriptions (see Figure 8.4). In particular, symbolic descriptions labelled with AA00 and AA08 appear tightly close in the scatterplot area. This result is confirmed

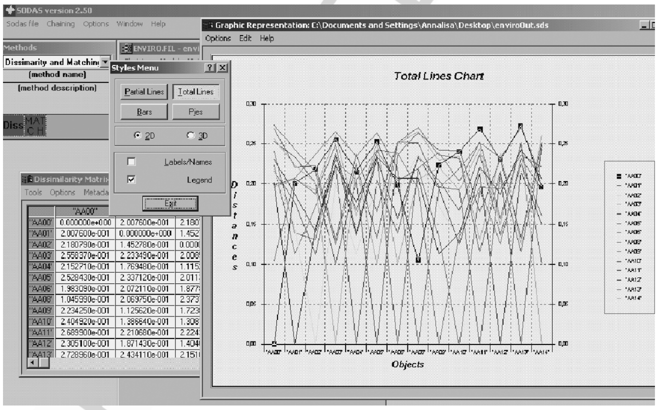


Dissimilarity Matrix: C:\Documents and Settings\Annalis

Tools Options Metadata Help

	"AA00"	"AA01"	"AA02"	
"AA00'	0.000000e+000	2.007600e-001	2.180790e-001	2.55
"AA01'	2.007600e-001	0.000000e+000	1.452780e-001	2.23
"AA02'	2.180790e-001	1.452780e-001	0.000000e+000	2.00
"AA03'	2.558370e-001	2.233490e-001	2.008580e-001	0.00
"AA04'	2.152710e-001	1.769480e-001	1.115270e-001	2.24
"AA05'	2.528430e-001	2.337120e-001	2.011390e-001	9.98
"AA06'	1.983090e-001	2.072110e-001	1.877840e-001	2.48
"AA08'	1.045990e-001	2.069750e-001	2.373750e-001	2.65
"AA09'	2.234250e-001	1.125620e-001	1.723880e-001	2.29
"AA10'	2.404920e-001	1.386640e-001	1.308160e-	
"AA11'	2.689900e-001	2.210680e-001	2.224360e-	
"AA12'	2.305100e-001	1.871430e-001	1.404020e-	
"AA13'	2.728960e-001	2.434110e-001	2.151030e-	

Figure 8.4 Scatterplot output of the symbolic descriptions stored in the enviro ASSO file, such that the Euclidean distance between the points preserves the dissimilarity values computed by the DISS module.



Dissimilarity Matrix

Tools Options Metada

	"AA00"		
"AA00'	0.000000e+000	2.007600e-001	2.180
"AA01'	2.007600e-001	0.000000e+000	1.452
"AA02'	2.180790e-001	1.452780e-001	0.000
"AA03'	2.558370e-001	2.233490e-001	2.008
"AA04'	2.152710e-001	1.769480e-001	1.115
"AA05'	2.528430e-001	2.337120e-001	2.011
"AA06'	1.983090e-001	2.072110e-001	1.877
"AA08'	1.045990e-001	2.069750e-001	2.373
"AA09'	2.234250e-001	1.125620e-001	1.723
"AA10'	2.404920e-001	1.386640e-001	1.308
"AA11'	2.689900e-001	2.210680e-001	2.224
"AA12'	2.305100e-001	1.871430e-001	1.404
"AA13'	2.728960e-001	2.434110e-001	2.151

Figure 8.5 Line chart output of the dissimilarity matrix computed on the mixed symbolic data stored in the enviro ASSO file.

when visualizing line charts of the dissimilarity matrix in question (see Figure 8.5). This result suggests that the SOs underlying the symbolic descriptions AA00 and AA08 have a small dissimilarity (i.e., large similarity), that is, they identify ‘homogeneous’ classes of objects.

8.6 The matching functions

Matching comparison is a directional judgement involving a referent and a subject. In SDA, the referent is an SO representing a class description, while the subject is an SO that typically corresponds to the description of an individual.

The matching problem consists of establishing whether the individual described by the subject can be considered an instance of the referent. For instance, the SO

$$r = [\text{colour} = \{\text{black}, \ \text{white}\}] \wedge [\text{height} = [170, 200]]$$

describes a group of individuals either black or white, whose height is in the interval [170, 200], while the SO

$$s_1 = [\text{colour} = \text{black}] \wedge [\text{height} = 180]$$

corresponds to an individual in the extent of $r(s_1 \in \text{Ext}_E(r))$, since it fulfils the requirements stated in r. Conversely, the SO

$$s_2 = [\text{colour} = \text{black}] \wedge [\text{height} = 160]$$

does not correspond to an individual in the extent of $r(s_2 \notin \text{Ext}_E(r))$, since $160 \notin [170, 200]$. Thus, we can say that r matches s_1 but not s_2.

More formally, given two SOs, r and s, the former describes a class of individuals (referent of matching), the latter an individual (subject of matching) and matching checks whether s is an individual in the class described by r. Canonical matching returns either 0 (failure) or 1 (success).

The occurrence of noise as well as the imprecision of measuring instruments makes canonical matching too restrictive in many real-world applications. This makes it necessary to rely on a flexible definition of matching that aims at comparing two descriptions and identifying their similarities rather than the equalities.

The result is a flexible matching function with ranges in the interval [0,1] that indicates the probability of a precisely matching the subject against a referent, provided that some change is possibly made in the description of the referent. Notice that both canonical matching and flexible matching are not a resemblance measure, due to the non-symmetry of the matching function.

In the following, we briefly describe the matching operators implemented in the MATCH module for BSOs and PSOs. In the case of mixed SOs, matching values obtained by comparing the boolean parts and the probabilistic parts are combined by product.

8.6.1 Matching functions for boolean symbolic objects

Let S be the space of BSOs in the form of assertions. The canonical matching between BSOs is defined as the function,

$$CanonicalMatch : S \times S \to \{0, 1\},$$

that assigns the value 1 or 0 to the matching of a referent $r \in S$ against a subject $s \in S$, where

$$r = [Y_1 \in w_1] \wedge [Y_2 \in w_2] \wedge \ldots \wedge [Y_p \in w_p],$$
$$s = [Y_1 \in v_1] \wedge [Y_2 \in v_2] \wedge \ldots \wedge [Y_p \in v_p].$$

More precisely, the canonical matching value is determined as follows:

$$CanonicalMatch(r, s) = \begin{cases} 1, & \text{if } v_j \subseteq w_j \forall j = 1, \ldots, p, \\ 0, & \text{otherwise.} \end{cases}$$

Conversely, the flexible matching between two BSOs is defined by

$$FlexMatch : S \times S \to [0, 1],$$

such that:

$$FlexMatch(r, s) = \begin{cases} 1, & \text{if } CanonicalMatch(r, s) = 1, \\ \in [0, 1], & \text{otherwise.} \end{cases}$$

Notice that the flexible matching yields 1 for an exact match.

In Esposito *et al.* (2000), the definition of flexible matching is based on probability theory in order to deal with chance and uncertainty. In this way, the result of flexible matching can be interpreted as the *probability* of r matching s, provided that a change is made in s.

More precisely, let

$$S(r) = \{s' \in S | CanonicalMatch(r, s') = 1\}$$

be the set of BSOs matched by r. Then the probabilistic view of flexible matching defines *FlexMatch* as the maximum conditional probability in $S(r)$, that is,

$$FlexMatch(r, s) = \max_{s' \in S(r)} P(s|s') \qquad \forall r, s \in S,$$

where $s(s')$ is the conjunction of simple BSOs (i.e., elementary events), that is, $s_1, \ldots, s_p(s'_1, \ldots, s'_p)$ such that each $s_i(s'_i)$ is in the form $[Y_i = v_i]([Y_i = v'_i])$. Then, under the assumption of conditional independence of the variables Y_j, the probability $\mathrm{P}(s|s')$ can be factorized as

$$P(s|s') = \prod_{i=1, \ldots, p} P(s_i|s') = \prod_{i=1, \ldots, p} P(s_i|s'_1 \wedge \ldots \wedge s'_p),$$

where $\mathrm{P}(s_i|s')$ denotes the probability of observing the event s_i given s'.

Suppose that s_i is an event in the form $[Y_i = v_i]$, that is, s describes an individual. If s' contains the event $[Y_i = v'_i]$, $P(s_i|s')$ is the probability that while we observed v_i, the true value was v'_i. By assuming that s_i depends exclusively on s'_i, we can write $P(s_i|s') = P(s_i|s'_i)$. This probability is interpreted as the similarity between the events $[Y_i = v_i]$ and $[Y_i = v'_i]$, in the sense that the more similar they are, the higher the probability:

$$P(s_i|s'_i) = P([Y_i = v_i] | [Y_i = v'_i]).$$

We denote by P the probability distribution of a random variable Y on the domain $\mathcal{Y}_i$ and δ_I a distance function such that $\delta_I : \mathcal{Y}_i \times \mathcal{Y}_i \to \mathfrak{R}$. We obtain that

$$P(s_i|s'_i) = P([Y_i = v_i] | [Y_i = v'_i]) = P(\delta_I(v'_j, Y) \geq \delta_I(v'_j, v_j)).$$

Henceforth, we make some tacit assumptions on the distance δ_I as well as on the probability distribution P when they are not specified (Esposito *et al.*, 1991). In particular, we assume that the distance function δ_I for continuous-valued variables is the L_1 norm,

$$\delta_I(v, w) = |v - w|;$$

for nominal variables it is the binary distance,

$$\delta(v, w) = \begin{cases} 0, & \text{if } v = w, \\ 1, & \text{otherwise;} \end{cases}$$

while for ordinal variables it is

$$\delta_I(v, w) = |\mathrm{ord}(v) - \mathrm{ord}(w)|,$$

where $\mathrm{ord}(v)$ denotes the ordinal number assigned to the value v.

The probability distribution of Y_i on the domain $\mathcal{Y}_i$ is assumed to be the uniform distribution.

Example 8.1. We assume a nominal variable Y_i with a set $\mathcal{Y}_i$ of categories and a uniform distribution of Y on the domain $\mathcal{Y}_i$. If we use the binary distance, then

$$P([Y_i = v_i]|[Y_i = v_i']) = \frac{|\mathcal{Y}_i| - 1}{|\mathcal{Y}_i|},$$

where $|\mathcal{Y}_i|$ denotes the cardinality of $\mathcal{Y}_i$.

The definition of flexible matching can be generalized to the case of comparing any pair of BSOs and not necessarily comparing a BSO describing a class with a BSO describing an individual. In this case, we have that:

$$FlexMatch(r, s) = \max_{s' \in S(r)} \prod_{i=1,\ldots,p} \sum_{j=1,\ldots,q} \frac{1}{q} P(s_{ij}|s_i'),$$

when q is the number of categories for the variable j in the symbolic object s.

Example 8.2. (Flexible matching between BSOs). Let us consider a pair of BSOs r (referent of matching) and s (subject of matching) in the form of assertions, such that:

$$r = [R_1 \in \{\text{yellow, green, white}\}] \wedge [R_2 \in \{\text{Ford, Fiat, Mercedes}\}],$$
$$s = [S_1 \in \{\text{yellow, black}\}] \wedge [S_2 \in \{\text{Fiat, Audi}\}],$$

such that $\mathcal{Y}_1 =$ {yellow, red, green, white, black} is the domain of both R_1 and S_1 while $|\mathcal{Y}_1|$ is the cardinality of $\mathcal{Y}_1$ with $|\mathcal{Y}_1 \quad| = 5$. Similarly $\mathcal{Y}_2 =$ { Ford, Fiat, Mercedes, Audi, Peugeot, Renault} is the domain of both R_2 and S_2 and $|\mathcal{Y}_2|$ is the cardinality of $\mathcal{Y}_2$ with $|\mathcal{Y}_2| = 6$. We build the set S_r as follows:

$$\begin{aligned}
S(r) = \{s' \in\ & S|CanonicalMatch\ (r, s') = 1\} = \{\\
& s'_1 = [S_1 = \text{yellow}] \wedge [S_2 = \text{Ford}];\\
& s'_2 = [S_1 = \text{yellow}] \wedge [S_2 = \text{Fiat}];\\
& s'_3 = [S_1 = \text{yellow}] \wedge [S_2 = \text{Mercedes}];\\
& s'_4 = [S_1 = \text{green}] \wedge [S_2 = \text{Ford}];\\
& s'_5 = [S_1 = \text{green}] \wedge [S_2 = \text{Fiat}];\\
& s'_6 = [S_1 = \text{green}] \wedge [S_2 = \text{Mercedes}];\\
& s'_7 = [S_1 = \text{white}] \wedge [S_2 = \text{Ford}];\\
& s'_8 = [S_1 = \text{white}] \wedge [S_2 = \text{Fiat}];\\
& s'_9 = [S_1 = \text{white}] \wedge [S_2 = \text{Mercedes}]\}.
\end{aligned}$$

When $s' = s_1$', we obtain that:

$$\begin{aligned}
P(s_{11}|s'_{11}) &= P(S_1 = \text{yellow}|S_1 = \text{yellow}) = 1,\\
P(s_{12}|s'_{11}) &= P(S_1 = \text{black}|S_1 = \text{yellow}) = \frac{|\mathcal{Y}_1| - 1}{|\mathcal{Y}_i|} = \frac{4}{5},\\
P(s_1|s'_1) &= 0.5(P(s_{11}|s'_{11}) + \text{P}(s_{12}|s'_{11})) = \frac{9}{10},\\
P(s_{21}|s'_{12}) &= P(S_2 = \text{Fiat}|S_2 = \text{Ford}) = \frac{|\mathcal{Y}_2| - 1}{|\mathcal{Y}_2|} = \frac{5}{6},\\
P(s_{22}|s'_{12}) &= \text{P}(S_2 = \text{Audi}|S_2 = \text{Ford}) = \frac{|\mathcal{Y}_2| - 1}{|\mathcal{Y}_2|} = \frac{5}{6},\\
P(s_2|s'_1) &= 0.5(\text{P}(s_{21}|s'_{12}) + \text{P}(s_{22}|s'_{12})) = \frac{5}{6}.
\end{aligned}$$

Consequently, we have that $P(s_1|s'_1) \times P(s_2|s'_1) = \frac{3}{4}$. This means that $FlexMatch(r, s) \geq 0.75$.

8.6.2 Matching functions for probabilistic symbolic objects

The definition of the flexible matching function given for BSOs can be extended to the case of PSOs. If $r$and s are two PSOs, the flexible matching of r (referent of matching) against s (subject of matching) can be computed as follows:

$$FlexMatch(r, s) = \max_{s' \in S(r)} \prod_{i=1,\ldots,p} P(s'_i) \sum_{j=1,\ldots,q} P(s_{ij}) P(s_{ij}|s'_i).$$

Example 8.3. (Flexible matching between PSOs). Let us consider a pair of PSOs r and s, such that:

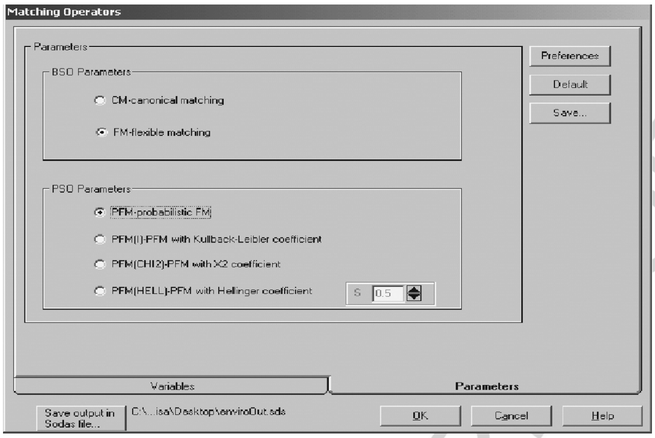
$$\begin{aligned} r = &[R_1 \in \{\text{yellow}(0.2), \text{green}(0.5), \text{white}(0.3)\}] \\ &\wedge [R_2 \in \{\text{Ford}(0.1), \text{Fiat}(0.5), \text{Mercedes}(0.4)\}] \\ s = &[S_1 \in \{\text{white}(0.6), \text{green}(0.4)\}] \wedge [S_2 \in \{\text{Fiat}(0.3), \text{Audi}(0.7)\}], \end{aligned}$$

such that $\mathcal{Y}_1 =$ {yellow, red, green, white, black} is the domain of both R_1 and S_1, while $\mathcal{Y}_2 =$ { Ford, Fiat, Mercedes, Audi, Peugeot, Renault} is the domain of both R_2 and S_2. We build the set S_r as follows:

$$\begin{aligned} S(r) = \{s' \in S | &CanonicalMatch(r, s') = 1\} = \{ \\ &s'_1 = [S_1 = \text{yellow}] \wedge [S_2 = \text{Ford}]; \\ &s'_2 = [S_1 = \text{yellow}] \wedge [S_2 = \text{Fiat}]; \\ &s'_3 = [S_1 = \text{yellow}] \wedge [S_2 = \text{Mercedes}]; \\ &s'_4 = [S_1 = \text{green}] \wedge [S_2 = \text{Ford}]; \\ &s'_5 = [S_1 = \text{green}] \wedge [S_2 = \text{Fiat}]; \\ &s'_6 = [S_1 = \text{green}] \wedge [S_2 = \text{Mercedes}]; \\ &s'_7 = [S_1 = \text{white}] \wedge [S_2 = \text{Ford}]; \\ &s'_8 = [S_1 = \text{white}] \wedge [S_2 = \text{Fiat}]; \\ &s'_9 = [S_1 = \text{white}] \wedge [S_2 = \text{Mercedes}]\}. \end{aligned}$$

When $s' = s_1$', we obtain that:

$$\begin{aligned} &P(s_{11}|s'_{11}) = \mathrm{P}(S_1 = \text{white}|S_1 = \text{yellow}) = \frac{4}{5}, \\ &P(s_{11}) \times P(s_{11}|s'_{11}) = \frac{3}{5} \times \frac{4}{5} = \frac{12}{25}, \\ &P(s_{12}|b'_{11}) = P(S_1 = \text{green}|S_1 = \text{yellow}) = \frac{4}{5}, \\ &P(s_{12}) \times P(s_{11}|s'_{11}) = \frac{2}{5} \times \frac{4}{5} = \frac{8}{25}, \\ &P(s_{11}) \times P(s_{11}|s'_{11}) + P(s_{12}) \times P(s_{11}|s'_{11}) = \frac{12}{25} + \frac{8}{25} = \frac{4}{5}, \\ &P(s_{21}|s'_{12}) = P(S_2 = \text{Fiat}|S_2 = \text{Ford}) = \frac{5}{6}, \\ &P(s_{21}) \times P(s_{21}|s'_{12}) = \frac{3}{10} \times \frac{5}{6} = \frac{1}{4}, \\ &P(s_{22}|s'_{12}) = P(S_1 = \text{Audi}|S_2 = \text{Ford}) = \frac{5}{6}, \end{aligned}$$



Matching Operators

Parameters

BSO Parameters

○ CM-canonical matching

● FM-flexible matching

PSO Parameters

● PFM-probabilistic FM

○ PFM(I)-PFM with Kullback-Leibler coefficient

○ PFM(CHI2)-PFM with X2 coefficient

○ PFM(HELL)-PFM with Hellinger coefficient S 0.5

C:\...isa\Desktop\enviroOut.sds

Figure 8.6 Setting the matching functions to compute the matching matrix.

AQ1

$$P(s_{22}) \times P(s_{22}|s'_{12}) = \frac{7}{10} \times \frac{5}{6} = \frac{35}{60},$$

$$P(s_{21}) \times P(s_{21}|s'_{12}) + P(s_{22}) \times P(s_{22}|s'_{12}) = \frac{1}{4} + \frac{35}{60} = \frac{5}{6}.$$

Consequently, we have that $(P(s'_{11}) \times (P(s_{11}) \times P(s_{11}|s'_{11}) + P(s_{12}) \times P(s_{11}|s'_{11}))) \times (P(s'_{12}) \times (P(s_{21}) \times P(s_{21}|s'_{12}) + P(s_{22}) \times P(s_{22}|s'_{12}))) = (\frac{1}{5} \times \frac{4}{5}) \times (\frac{1}{10} \times \frac{5}{6}) = \frac{1}{75}$. This means that $FlexMatch(r, s) \geq \frac{1}{75}$.

Alternatively, the flexible matching of r against s can be performed by comparing each pair of probabilistic variables R_i and S_i, which take values on the same range $\mathcal{Y}_i$, according to some non-symmetric function f and aggregating the results by product, that is:

$$FlexMatch(r, s) = \prod_{i=1,\ldots,p} f(R_i, S_i).$$

For this purpose, several comparison functions for probability distributions, such as the KL divergence, the χ^2 divergence and the Hellinger coefficient (see Section 8.3.2) have been implemented in the MATCH module. Notice that both the KL divergence and the χ^2 divergence are two dissimilarity coefficients. Therefore, they are not suitable for computing matching. However, they may be easily transformed into similarity coefficients by

$$f(P, Q) = \mathrm{e}^{-x},$$

where x denotes either the KL divergence value or the χ^2 divergence value.

8.7 Output of MATCH

The MATCH module outputs a new ASSO file that includes both the symbolic data table D stored in the input ASSO file and the matching matrix M such that $M(i, j)$ is the (canonical or flexible) matching value of the *i*th SO (referent) against the *j*th SO (subject) taken from the input data table. Matching values are computed for each pair of SOs whose descriptions are stored in the input ASSO file.

In addition, a report file is generated that is a printer-formatted file describing the input parameters and the matching matrix.

Finally, if a metadata file is associated with the input ASSO file, MATCH updates metadata by recording both the matching functions and the list of symbolic variables involved in the computation of the matching function.

8.8 An Application of the MATCH Module

In this section, we describe a case study involving the computation of the matching matrix from the SOs underlying as assertions the symbolic descriptions stored in enviro.xml. To this end, we create a new ASSO chain that includes a base block associated to the enviro.xml file. The running chain is then extended with a new block that is assigned to the MATCH module.

Before computing the matching matrix, users choose the list of symbolic variables to be involved in computing matching values, the matching functions to be computed, the name of the output ASSO file, and so on. Notice that in the current version of Match, users are not able to select a subset of SOs to be involved in matching computation. Conversely, all the SOs whose symbolic descriptions are stored in input ASSO file are processed to compute the matching matrix.

Both the list of variables and the matching functions are set by selecting Parameters. . . from the pop-up menu associated with the Match block in the running chain. The list of symbolic variables taken from the symbolic data table stored in the input ASSO file is shown in a list box and some statistics (e.g. minimum and maximum or frequency distribution) can be output for each variable.

Users choose symbolic variables to be considered when computing the matching matrix of the SOs taken from the enviro data. By default, all variables can be selected by users without any restriction on the type. However, users may decide to output only a subset of these variables (e.g. interval variables or probabilistic variables).

In this application, we decide to select all the symbolic variables (13 interval variables and four probabilistic variables) from the enviro data. This means that the SOs considered for matching computation are mixed SOs, where the boolean part is separated from the probabilistic part. The matching values computed when comparing both the boolean and probabilistic parts are then combined by product (see Figure 9.6).

When all the parameters are set, the matching matrix is built by choosing Run method from the pop-up menu associated with the MATCH block in the running chain.

If the matching matrix is correctly constructed, then *Match* produces as output a new ASSO file (e.g. enviroMatch.sds) that is associated with the current chain and is stored in the user-defined path and includes both the input symbolic data table and the matching matrix computed by MATCH.

A report file describing the input parameters (e.g. matching functions or symbolic variables) and the matching matrix is associated with a new block that is automatically introduced into the running chain and directly connected to the MATCH block (see Figure 8.7(a)).

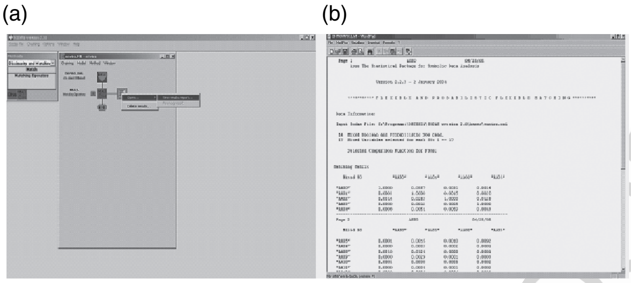
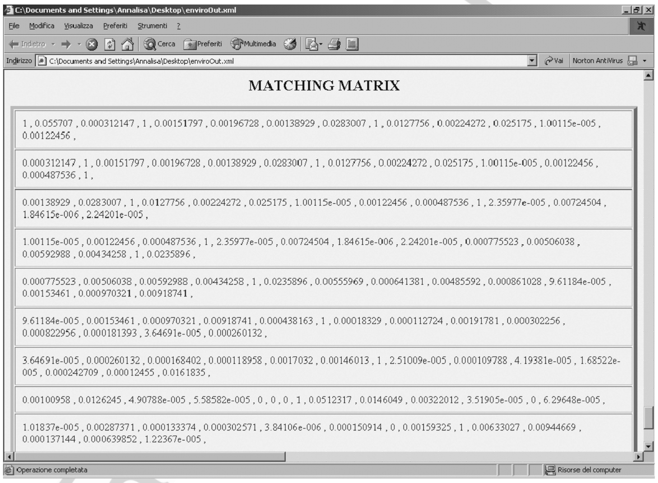


Figure 8.7 (a) Example of the ASSO chain including the Report block generated by MATCH and (b) the output of the report on the matching computation performed by MATCH.


C:\Documents and Settings\Annalisa\Desktop\enviroOut.xml
File Modifica Visualizza Preferiti Strumenti ?
Indietro Cerca Preferiti Multimedia
Indirizzo C:\Documents and Settings\Annalisa\Desktop\enviroOut.xml
Vai Norton AntiVirus
Operazione completata
Risorse del computer

MATCHING MATRIX

```
1 , 0.055707 , 0.000312147 , 1 , 0.00151797 , 0.00196728 , 0.00138929 , 0.0283007 , 1 , 0.0127756 , 0.00224272 , 0.025175 , 1.00115e-005 ,
0.00122456 ,

0.000312147 , 1 , 0.00151797 , 0.00196728 , 0.00138929 , 0.0283007 , 1 , 0.0127756 , 0.00224272 , 0.025175 , 1.00115e-005 , 0.00122456 ,
0.000487536 , 1 ,

0.00138929 , 0.0283007 , 1 , 0.0127756 , 0.00224272 , 0.025175 , 1.00115e-005 , 0.00122456 , 0.000487536 , 1 , 2.35977e-005 , 0.00724504 ,
1.84615e-006 , 2.24201e-005 ,

1.00115e-005 , 0.00122456 , 0.000487536 , 1 , 2.35977e-005 , 0.00724504 , 1.84615e-006 , 2.24201e-005 , 0.000775523 , 0.00506038 ,
0.00592988 , 0.00434258 , 1 , 0.0235896 ,

0.000775523 , 0.00506038 , 0.00592988 , 0.00434258 , 1 , 0.0235896 , 0.00555969 , 0.000641381 , 0.00485592 , 0.000861028 , 9.61184e-005 ,
0.00153461 , 0.000970321 , 0.00918741 ,

9.61184e-005 , 0.00153461 , 0.000970321 , 0.00918741 , 0.000438163 , 1 , 0.00018329 , 0.000112724 , 0.00191781 , 0.000302256 ,
0.000822956 , 0.000181393 , 3.64691e-005 , 0.000260132 ,

3.64691e-005 , 0.000260132 , 0.000168402 , 0.000118958 , 0.0017032 , 0.00146013 , 1 , 2.51009e-005 , 0.000109788 , 4.19381e-005 , 1.68522e-
005 , 0.000242709 , 0.00012455 , 0.0161835 ,

0.00100958 , 0.0126245 , 4.90788e-005 , 5.58582e-005 , 0 , 0 , 0 , 1 , 0.0512317 , 0.0146049 , 0.00322012 , 3.51905e-005 , 0 , 6.29648e-005 ,

1.01837e-005 , 0.00287371 , 0.000133374 , 0.000302571 , 3.84106e-006 , 0.000150914 , 0 , 0.00159325 , 1 , 0.00633027 , 0.00944669 ,
0.000137144 , 0.000639852 , 1.22367e-005 ,
```

Figure 8.8 Matching matrix computed from the MATCH module on enviro data.

This report can be output as a printer-formatted file by selecting Open. . . and then View Result Report. . . from the pop-up menu associated with the report block (see Figure 8.7(b)). Alternatively, the report block can be removed from the running chain by selecting Delete Results. . . from the pop-up menu.

The matching matrix is stored in the output ASSO file that is associated with the current chain. Both the enviro symbolic data table and matching matrix are stored in XML format (see Figure 8.8). Such matching values are involved in the directional comparison of the SOs extracted from the enviro data in order to identify the set of SOs matched (i.e., covered) from each fixed SO.

Finally, the metadata file is updated by recording matching measures and symbolic variables involved in the matching comparison.

References

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

Batagelj, V. and Bren, M. (1995) Comparing resemblance measures. *Journal of Classification*, **12**, 73–90.

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

Bock, H.-H. and Diday, E. (2000) Symbolic objects. In H.-H. Bock and E. Diday (eds), *Analysis of Symbolic Data: Exploratory Methods for Extracting Statistical Information from Complex Data*, pp. 54–77. Berlin: Springer-Verlag.

Csiszár, I. (1967) Information-type measures of difference of probability distributions and indirect observations. *Studia Scientiarum Mathematicarum Hungarica*, **2**, 299–318.

De Carvalho, F.A.T. (1994) Proximity coefficients between boolean symbolic objects. In E. Diday, Y. Lechevallier, M. Schader, P. Bertrand and B. Burtschy (eds), *New Approaches in Classification and Data Analysis*, pp. 387–394. Berlin: Springer-Verlag.

De Carvalho, F.A.T. (1998) Extension based proximity coefficients between constrained Boolean symbolic objects. In C. Hayashi, K. Yajima, H.-H. Bock, N. Ohsumi, Y. Tanaka and Y. Baba (eds),*Data Science, Classification, and Related Methods*, pp. 370–378. Tokyo: Springer-Verlag.

Diday, E. and Esposito F. (2003) An introduction to symbolic data analysis and the SODAS software. *Intelligent Data Analysis*, **7**, 583–602.

Esposito, F., Malerba, D., Semeraro, G. (1991) Flexible matching for noisy structural descriptions. In J. Mylopoulos and R. Reiter (eds), *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pp. 658–664. San Mateo, CA: Morgan Kaufmann.

Esposito, F., Malerba, D. and Lisi, F.A. (2000) Matching Symbolic Objects. In H.-H. Bock and E. Diday (eds), *Analysis of Symbolic Data: Exploratory Methods for Extracting Statistical Information from Complex Data*, pp. 186–197. Berlin: Springer-Verlag.

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

Ichino, M. and Yaguchi, H. (1994) Generalized Minkowski metrics for mixed feature-type data analysis. *IEEE Transactions on Systems, Man, and Cybernetics*, **24**, 698–707.

Kang, K. and Sompolinsky, H. (2001) Mutual information of population codes and distance measures in probability space. *Physical Review Letters*, **86**, 4958–4961.

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

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

Lin, J. (1991) Divergence measures based on the Shannon entropy. *IEEE Transactions on Information theory*, **37**, 145–151.

Malerba, D., Esposito, F., Gioviale, V. and Tamma, V. (2001) Comparing dissimilarity measures for symbolic data analysis. In *Proceedings of Techniques and Technologies for Statistics – Exchange of Technology and Know-How*, Crete, 1, pp. 473–481. http://www.csc.liv.ac.uk/∼valli/Papers/ntts-asso.pdf (accessed May 2007).

Malerba, D., Esposito, F. and Monopoli, M. (2002) Comparing dissimilarity measures for Probabilistic Symbolic Objects. In A. Zanasi, C.A. Brebbia, N.F.F. Ebecken and P. Melli (eds), *Data Mining III, Vol. 6: Series Management Information Systems*, pp. 31–40. Southampton: WIT Press.

Patterson, D.W. (1990) *Introduction to Artificial Intelligence and Expert Systems*. London: Prentice Hall.

Rached, Z., Alajaji, F. and Campbell, L.L. (2001) Rényi's divergence and entropy rates for finite alphabet Markov sources. *IEEE Transactions on Information Theory*, **47**, 1553–1561.

Sammon, J.J.W. (1969) A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers C*, **18**, 401–409.

QUERIES TO BE ANSWERED BY AUTHOR (SEE MARGINAL MARKS)

IMPORTANT NOTE: Please mark your corrections and answers to these queries directly onto the proof at the relevant place. Do NOT mark your corrections on this query sheet.

Chapter 08

Q. No.	Pg No.	Line No.	Query
AQ1	145	08	Please Provide citation for fig 8.6.