AN ALGORITHM TO COMPUTE THE TRANSITIVE CLOSURE, A TRANSITIVE APPROXIMATION AND A TRANSITIVE OPENING OF A PROXIMITY*

LUIS GARMENDIA
Universidad Complutense of Madrid, Spain
lgarmend@fdi.ucm.es

RAMÓN GONZALEZ
Universidad Complutense of Madrid, Spain
rgonzle@estad.ucm.es

JORDI RECASENS
Universitat Politècnica de Catalunya, Spain
j.recasens@upc.edu

A method to get the transitive closure, a transitive opening and a transitive approximation of a reflexive and symmetric fuzzy relation is presented. The method builds at the same time a binary partition tree for the output similarities.

1. Introduction

Equivalence relations are important in many branches of knowledge and especially in Classification theories and Cluster Analysis since they generate a partition on the universe of discourse and permit to classify their elements and make clusters. In many cases the relation we start with is not an equivalence relation but only a reflexive and symmetric one.

A very important family of fuzzy relations are T-indistinguishabilities (reflexive, symmetric and T-transitive fuzzy relations) since they generalize (fuzzify) the concepts of (crisp) equivalence relation and equality [Trillas and Valverde 1984] and are useful to represent the ideas of similarity and neighbourhood as well.

Among T-indistinguishabilities, the ones which are transitive with respect to the *Minimum* t-norm are called similarities and are especially interesting and widely used in Taxonomy since they generate indexed hierarchical trees.

How to obtain T-indistinguishabilities and especially similarities from a given proximity relation R, has become a very important task and there are many algorithms to do it. Many of them calculate the smallest T-indistinguishability

[*] Research partially supported by DGICYT projects number TIN2006-14311 and TIN2006-06190.

greater of equal than R, which is called its T-transitive closure. Other methods calculate T-transitive openings, which are T-indistinguishabilities smaller that R but maximal among all T-indistinguishabilities smaller than R. Though the T-transitive closure of a fuzzy proximity is unique, it is not the case of the T-transitive openings and though there are some algorithms to calculate some of them, there is still a very interesting open problem to find all T-transitive openings of a given fuzzy proximity. Less attention has been paid to the obtention of T-indistinguishability operators not comparable with R in the sense that some of the entries are greater while some smaller than the corresponding entries of R. These T-transitive relations will be called T-transitive approximations of R in this paper. Despite the little interest since now in these approximations, it is obvious their importance since if we must replace a given fuzzy proximity by a T-transitive one, it is clear that in most occasions they will be closer to our relation than its corresponding T-transitive closure or some of its T-transitive openings ([Garmendia and Recasens 2007]).
This paper provides a simple algorithm to produce similarities (and therefore indexed hierarchical trees) from a fuzzy proximity relation R. Several forms of defining weights allow to easily computing the transitive closure of R, a transitive opening and different transitive approximations of R. An interesting feature of it is that with the same algorithm the T-transitive closure, a T-transitive opening and several T-transitive approximations of R are generated.

2. Preliminaries

This section contains some definitions and properties of similarities and T-indistinguishabilities and some methods to build them from fuzzy proximity relations.
Definition 1: Let $E = \{e_1, ..., e_n\}$ be a finite set. A fuzzy relation R on E is a map $R: E\times E \to [0, 1]$. The relation degree value for elements e_i and e_j in E is called e_{ij}. So $e_{ij} = R(e_i, e_j)$.
A fuzzy relation R is **reflexive** if $e_{ii} = 1$ for all $1 \le i \le n$.
A fuzzy relation R is α-**reflexive** if $e_{ii} \ge \alpha$ for all $1 \le i \le n$.
The relation R is **symmetric** if $e_{ij} = e_{ji}$ for all $1 \le i, j \le n$.
A reflexive and symmetric fuzzy relation is called a fuzzy **proximity** relation.
Definition 2. Let T be a triangular norm [Schweizer, Sklar; 1984]. A fuzzy relation $R: E\times E \to [0, 1]$ is **T-transitive** if and only if $T(R(a, b), R(b, c)) \le R(a, c)$ for all a, b, c in E. In a fuzzy logic context it can be interpreted as 'The sentence "If a is related to b and b is related to c, then a is related to c" is true'.
Definition 4. [Zadeh 1971] A fuzzy **similarity** is a reflexive, symmetric and min-transitive fuzzy relation.
The T-transitive closure of a symmetric fuzzy relation is also symmetric. Also reflexivity and α-reflexivity are preserved by the T-transitive closure.

3. Algorithm to compute the transitive closure, a transitive opening and a transitive approximation of a fuzzy proximity generating a binary partition tree.

This section presents an algorithm that allows the computation of the transitive closure, a transitive opening and several other approximations of a give fuzzy proximity R. The fact that the same algorithm generates all these kind of approximations of R simplifies calculations and makes it a good tool to solve the problem of approximating fuzzy proximities, since the user can choose which kind of T-transitive approximation wants or needs.

Lemma 2. [Lee 2001] Let C and D be two fuzzy relations and

E (f; C, D) = $\begin{pmatrix} \boxed{C} & \boxed{F^T} \\ \boxed{F} & \boxed{D} \end{pmatrix}$ where all values in the box F are f.

If C and D are fuzzy similarities, then E(f; C, D) = E is also a fuzzy similarity, $\forall f \in [0, \min(\min(C), \min(D))]$.
The algorithm goes as follows.

Algorithm 1. Let R be a fuzzy proximity relation on a universe E = {e_1, ..., e_n} with values $e_{ij} = R(e_i, e_j)$. Lets call node to a subset of E (a node is an element of $\wp$(E)). In order to make an easier notation, we consider the elements of E by their natural number of their position.
Input: a proximity R
Output: Partition tree T (and matrix) of the transitive closure A = [aij], a transitive opening B = [bij] and a transitive approximation C = [cij] from R.
The given algorithm is the following:

1) Create a set of nodes N initially with a set of singletons $N_i = \{e_i\}$ for each element e_i in E.

2) Set a_{ii}=1, b_{ii}=1, and c_{ii}= 1 for all i from 1 to n.

3) n-1 times (while N is not the universe E) {

Compute $m(N_i, N_j) = \max_{i \in N_i\, j \in N_j} e_{i,j}$ for all pair of nodes NxN with $i \neq j$.

Record (i, j) where $m(N_i, N_j)$ is maximal.

Assign $a_{rs} = a_{sr} := \max_{i \in N_i\, j \in N_j} e_{i,j}$ for all $r \in N_i$ and $s \in N_j$.

Assign $b_{rs} = b_{sr} := \min(\min_{i \in N_i\, j \in N_j} e_{i,j}, \min_{k,l \in N_i} b_{k,l}, \min_{k,l \in N_{ji}} b_{k,l})$ for all $r \in N_i$ and $s \in N_j$.

Assign c_{rs}= c_{sr} := $\min(\operatorname*{avg}_{i\in N_i\, j\in N_j} e_{i,j}, \min_{k,l\in N_i} c_{k,l}, \min_{k,l\in N_{ji}} c_{k,l})$ **for all r∈N_i and s∈N_j.**

Delete nodes N_i and N_j from N.

Insert $N_i \cup N_j$ into N.

}

avg is an idempotent aggregation operator. In particular, avg can be any quasi-arithmetic mean such the arithmetic or geometric means, or an OWA operator, including the Minimum (corresponding to a T-transitive opening). Using different aggregator operators in this algorithm can provide different transitive approximations (or even a set of them computing several similarities Bi) taking the same time complexity.
The trees generated by A, B and C coincide. Only their weights differ.
The algorithm takes just n-1 steps, where n is the cardinality of the universe E. It takes O(n2) space complexity and O(n2log n) average time complexity.

2.1. *Example :*

Let R be the fuzzy proximity given by the following matrix:

$$R = \begin{pmatrix} 1 & 1 & 0.5 & 0.3 & 0.2 & 0.3 \\ 1 & 1 & 0.8 & 0.2 & 0.4 & 0.3 \\ 0.5 & 0.8 & 1 & 0.9 & 0.3 & 0.3 \\ 0.3 & 0.2 & 0.9 & 1 & 0.8 & 0.1 \\ 0.2 & 0.3 & 0.3 & 0.8 & 1 & 0.5 \\ 0.3 & 0.2 & 0.3 & 0.1 & 0.5 & 1 \end{pmatrix}$$

The first two loops of the part 3) of the algorithm records m(N1, N2) = 1 and m(N3, N4) = 0,9.
In the third loop a maximal value it is found with $m(N_3 \cup N_4, N_5) = 0.8$
The matrix construction of the transitive closure A, transitive opening B and transitive approximation C is in this step as follows

A	B	C
$\begin{pmatrix} 1 & 1 & & & & \\ 1 & 1 & & & & \\ & & 1 & 0.9 & 0.8 & \\ & & 0.9 & 1 & 0.8 & \\ & & 0.8 & 0.8 & 1 & \\ & & & & & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 & & & & \\ 1 & 1 & & & & \\ & & 1 & 0.9 & 0.3 & \\ & & 0.9 & 1 & 0.3 & \\ & & 0.3 & 0.3 & 1 & \\ & & & & & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 & & & & \\ 1 & 1 & & & & \\ & & 1 & 0.9 & 0.55 & \\ & & 0.9 & 1 & 0.55 & \\ & & 0.55 & 0.55 & 1 & \\ & & & & & 1 \end{pmatrix}$

Figure 1 Transitive closure A, transitive opening B and transitive approximation C in step 3.

In one more step, $N_1 \cup N_2$ and N_3

The matrix construction of the transitive closure A, transitive opening B and transitive approximation C is in this step as follows.

A

$$\begin{pmatrix}\begin{pmatrix}\begin{pmatrix}1&1\\1&1\end{pmatrix} & \begin{matrix}0.8&0.8&0.8\\0.8&0.8&0.8\end{matrix}\\ \begin{matrix}0.8&0.8\\0.8&0.8\\0.8&0.8\end{matrix} & \begin{pmatrix}\begin{pmatrix}1&0.9\\0.9&1\end{pmatrix}&\begin{matrix}0.8\\0.8\end{matrix}\\ \begin{matrix}0.8&0.8\end{matrix}&1\end{pmatrix}\end{pmatrix} & \\ & 1\end{pmatrix}$$

B

$$\begin{pmatrix}\begin{pmatrix}\begin{pmatrix}1&1\\1&1\end{pmatrix} & \begin{matrix}0.2&0.2&0.2\\0.2&0.2&0.2\end{matrix}\\ \begin{matrix}0.2&0.2\\0.2&0.2\\0.2&0.2\end{matrix} & \begin{pmatrix}\begin{pmatrix}1&0.9\\0.9&1\end{pmatrix}&\begin{matrix}0.3\\0.3\end{matrix}\\ \begin{matrix}0.3&0.3\end{matrix}&1\end{pmatrix}\end{pmatrix} & \\ & 1\end{pmatrix}$$

C

$$\begin{pmatrix}\begin{matrix}1&1&0.38&0.38&0.38\\1&1&0.38&0.38&0.38\\0.38&0.38&1&0.9&0.55\\0.38&0.38&0.9&1&0.55\\0.38&0.38&0.55&0.55&1\end{matrix} & \\ & 1\end{pmatrix}$$

Figure 2 Transitive closure A, transitive opening B and transitive approximation C is in loop 4.

Finally, the last node is linked in loop 5. Note that there are only five (n-1) loops because the universe E has 6 elements.

A

$$\begin{pmatrix}1&1&0.8&0.8&0.8&0.5\\1&1&0.8&0.8&0.8&0.5\\0.8&0.8&1&0.9&0.8&0.5\\0.8&0.8&0.9&1&0.8&0.5\\0.8&0.8&0.8&0.8&1&0.5\\0.5&0.5&0.5&0.5&0.5&1\end{pmatrix}$$

B

$$\begin{pmatrix}\begin{pmatrix}\begin{pmatrix}1&1\\1&1\end{pmatrix} & \begin{matrix}0.2&0.2&0.2\\0.2&0.2&0.2\end{matrix}\\ \begin{matrix}0.2&0.2\\0.2&0.2\\0.2&0.2\end{matrix} & \begin{pmatrix}\begin{pmatrix}1&0.9\\0.9&1\end{pmatrix}&\begin{matrix}0.3\\0.3\end{matrix}\\ \begin{matrix}0.3&0.3\end{matrix}&1\end{pmatrix}\end{pmatrix} & \begin{matrix}0.1\\0.1\\0.1\\0.1\\0.1\end{matrix}\\ \begin{matrix}0.1&0.1&0.1&0.1&0.1\end{matrix} & 1\end{pmatrix}$$

C

$$\begin{pmatrix}1&1&0.38&0.38&0.38&0.28\\1&1&0.38&0.38&0.38&0.28\\0.38&0.38&1&0.9&0.55&0.28\\0.38&0.38&0.9&1&0.55&0.28\\0.38&0.38&0.55&0.55&1&0.28\\0.28&0.28&0.28&0.28&0.28&1\end{pmatrix}$$

Figure 3 transitive opening B of the fuzzy proximity R, and its binary weighted tree (with the same shape that the T-transitive closure binary tree, but different values)

3. Conclusions

A method to get the transitive closure, a transitive opening and a transitive approximation of a reflexive and symmetric fuzzy relation at the same time is given.
The binary partition trees of the output similarities are the same.
Some examples are provided.

Acknowledgements

The author would like to thank the anonymous referees for the constructive comments and contributions.
Research partially supported by DGICYT projects number TIN2006-14311 and TIN2006-06190.

References

1. Bandler, W. Kohout, J. Special properties, closures and interiors of crisp and fuzzy relations. Fuzzy Sets and Systems 26 (1988) 317-331.
2. De Baets, B., De Meyer, H., Transitive approximation of fuzzy relations by alternating closures and openings, Soft Computing 7 (2003) 210-219.
3. Elorza, J., Burillo, P, On the relation of fuzzy preorders and fuzzy consequence operators, International Journal of Uncertainty, Fuzziness and Knoledge-based Systems 7 (3) (1999).
4. Esteva, F, Garcia, P., Godo, L., Rodriguez, R. O., Fuzzy approximation relations, modal structures and possibilistic logic, Mathware and Soft Computing 5 (2-3) (1998) 151-166.
5. Garmendia, L., Campo, C., Cubillo, S., Salvador, A. A Method to Make Some Fuzzy Relations T-Transitive. International Journal of Intelligence Systems. Vol. 14, Nº 9, (1999) 873 – 882.
6. Garmendia, L, Recasens, J. Finding close T-indistinguishability Operators to a given Proximity. Proceedings EUSFLAT2007. Ostrava. (2007). Vol 1, 127-133.
7. Jacas, J., Recasens, J., Fuzzy T-transitive relations: eigenvectors and generators, Fuzzy Sets and Systems 72 (1995) 147–154.
8. Jacas, J., Recasens, J., Decomposable indistinguishability operators, Proceedings of the Sixth IFSA Congress, Sao Paulo (1995).
9. Larsen H., R. Yager, “Efficient computation of transitive closures,” Fuzzy Sets Syst., vol. 38 (1990) 81–90.
10. Lee, H.-S. An optimal algorithm for computing the max–min transitive closure of a fuzzy similarity matrix , Fuzzy Sets and Systems 123 (2001) 129–136.
11. Naessens, H., De Meyer, H., De Baets, B., Algorithms for the Computation of T-Transitive Closures, IEEE Trans Fuzzy Systems 10:4 (2002) 541-551.
12. Ovchinnikov, S. Representations of Transitive Fuzzy Relations, in Aspects of Vagueness, H. J. Skala, S. Termini y E. Trillas (Eds.), Reidel Pubs. (1984) 105-118.
13. Schweizer, B., Sklar A. Probabilistic Metric Spaces, North-Holland, New York, (1984).
14. Trillas, E., Valverde, L., An inquiry into indistinguishability operators, in Aspects of Vagueness, H. J. Skala, S. Termini y E. Trillas (Eds.), Reidel Pubs. (1984) 231-256.
15. Valverde, L., On the structure of F-indistinguishability operators, Fuzzy Sets and Systems 17 (1985) 313–328.
16. Zadeh, L. A., Similarity relations and fuzzy orderings, Inform. Sci. 3 (1971) 177–200.