Extending the K-Nearest Neighbour Classification Algorithm to Symbolic Objects

Estensione agli oggetti simbolici dell'algoritmo di classificazione basato su K-Viciniori

Claudia D'Amato, Floriana Esposito, Donato Malerba, Marianna Monopoli
Dipartimento di Informatica, Università di Bari – Italy,
cldam@libero.it, {esposito, malerba,monopoli}@di.uniba.it

Riassunto: L'analisi di dati simbolici generalizza alcuni metodi statistici standard al caso di oggetti simbolici (SO). Questi oggetti, informalmente definiti "dati aggregati", poiché sintetizzano le informazioni relative ad un gruppo di individui, possono essere confrontati al fine di individuare dei cluster, di classificarli o ordinarli in base al loro grado di generalizzazione.
L'articolo propone un'estensione dell'algoritmo classico di classificazione K-Nearest Neighbor a tali oggetti. Il risultato di questo algoritmo è ancora un insieme di oggetti simbolici che possono essere studiati mediante altre tecniche di analisi di dati simbolici.

Keywords: Analisi dei dati simbolici, Data Mining, K-Nearest Neighbor

1. Introduction

The research activities in the field of data mining aim to study and develop techniques, methods and tools to extract useful information from large data sets. Statistical data analysis techniques have certainly influenced the growth of those tools that support intelligently and automatically the user during data treatment, but most of these techniques are designed for a relatively simple situation, where the observation unit is an individual (person, object) described by a well-defined set of random variables (qualitative or quantitative) each of which result in just one single value. However, in many situations, such as privacy, identity or data confidentiality protection, data analysts cannot access the single individuals (*first-order objects*).
A solution to this problem comes from Symbolic Data Analysis (SDA), which generalizes some standard statistical data mining methods, such as those developed for classification and clustering tasks, to a form of "aggregated data" named symbolic objects (SOs) (Bock and Diday, 2000). In SDA the observation unit is no more an individual but a class (*second-order object*), that is, a group of individuals described by a set-valued (interval or multi-valued) or modal variables also termed symbolic or descriptive variables. SOs can be divided in two main categories: boolean symbolic objects[2] (*BSO*) described only by set-valued variables, and probabilistic symbolic objects[3] (*PSO*), described also by modal variables. A set of SOs, which involves the same variables to describe different (possibly overlapping) classes of individuals, can be

[2] An example of BSO is: [hair color={white,black}] $\wedge$ [age $\in$ [20,29]] $\wedge$ [sex={M,F}]
[3] An example of PSO is: [hair color={white(0.6),black(0.4)}] $\wedge$ [age $\in$ [20,29]] $\wedge$ [sex={M(1)}]

described by a single table, called *symbolic data table*, where rows correspond to distinct symbolic data while columns correspond to descriptive variables.
In this context, there is a rapidly increasing need to extend standard data analysis methods (exploratory, graphical representations, clustering, classification) to these symbolic data.
This paper presents the extension of the distance weighted K-Nearest Neighbour (KNN) classification algorithm to SOs. The main novelties of the proposed extension are the use of a dissimilarity measure between SOs, the automated selection of K on the basis of cross-validation, and the output of a symbolic modal variable instead of a single class-value.

2. KNN for Symbolic Objects

The problem to be solved is the following: given a training set of SOs described by *p* modal or Boolean symbolic variables V_1, V_2, …, V_p and by a single-valued variable C, named target or class variable, with domain $\mathcal{C}$, we want to determine the value of a modal variable C' with domain $\mathcal{C}$ for a new SO (test case) described by the same set of symbolic variables used for the training set, namely V_1, V_2, …, V_p. The modality of the C' is probabilistic, meaning that we associate the new SO with a class probability vector whose dimension corresponds to the number of distinct values or classes in $\mathcal{C}$.
According to the KNN algorithm, the assignment of a value to C' can be based on the values taken by C for the K-nearest neighbors of the test case. However, the standard KNN algorithm assumes all training cases correspond to points in the p-dimensional space $\Re^p$, and the nearest neighbours of a new case are defined in terms of the standard Euclidean distance. Therefore, the extension of KNN to SOs requires the use of a dissimilarity measure *d* for SOs, which cannot be represented as points in $\Re^p$.
Many proposals of dissimilarity measures for BSOs have been reported in literature; an extensive review of their definitions is reported in (Esposito et al., 2000), while a preliminary comparative study on their suitability to real-world problems is reported in (Malerba et al., 2001). Recently, a set of dissimilarity measures has also been proposed for the case of PSOs defined by multi-valued variables (Malerba et al., 2002). Their definitions are based on different measures of divergence between two discrete probability distributions, which are associated to each SO for some multi-valued variable V_i. They all fulfil the classical conditions $0=d(a,a)\leq d(a,b)=d(b,a)<\infty$ for any pair of SOs *a,b*, while the triangle property holds only for some of them. Moreover, some dissimilarity measures are also defined for constrained SOs, that is, SOs where logical or taxonomical relations exist among variables. This means that our extended version of KNN can also properly work on data sets where some form of domain knowledge, expressed in the form of simple IF-THEN rules, is available.
In the classical KNN algorithm, all neighbours equally contribute to the determination of the value of the class variable. The proposed extension upgrades the distance-weighted KNN algorithm, which weights the contribution of each of the K neighbours according to their distance to the new case, giving greater weight to closer neighbours. The algorithm can be formally described as follows:

- Given a test object x_q to classify and a dissimilarity measure d, let $x_1, x_{2,\ldots,} x_k$ the k training objects most similar to the test object, that is, the k training objects so that $d(x_q, x_i)$ is minimal;
- Let n be the number of all the possible classes. It is possible to distinguish among tree different cases:
    1. For all nearest neighbours x_i such that $d(x_q, x_i) = 0$ (i.e., identical to the test case) each $C(x_i)$ is equal to the same class c. Then the output class probabilities are:
       $P(C'(x_q)=c)=1$ and $P(C'(x_q)=\nu_j)=0 \quad \forall j=1,\ldots,n$ such that $\nu_j \neq c$.
    2. Classes $C(x_i)$ for all nearest neighbours x_i such that $d(x_q, x_i) = 0$ can differ between them. Then the output class probabilities are estimated on the basis of the class values taken by nearest neighbour identical to the test case:
       $$P(C'(x_q)=\nu_j) = \frac{\#x_i : C'(x_i) = \nu_j}{K} \quad \forall j=1,\ldots,n;$$
    3. For all nearest neighbours x_i, $d(x_q, x_i) \neq 0$ for i=1..k. Let $\omega_i = 1/d(x_q, x_i)$ be the associated weight to the training object x_i. We define $\Omega_j = \sum_{i=1}^{k} \omega_i * \delta(\nu_j, C(x_i))$ for j=1,..n, where $\delta(\nu_j, C(x_i))=1$ if $C(x_i)=\nu_j$; $\delta(\nu_j, C(x_i))=0$ otherwise. Then the output class probabilities are estimated as follows:
       $$P(C'(x_q)=\nu_j) = \frac{\frac{\#x_i : C'(x_i) = \nu_j}{K} \cdot \Omega_j}{\sum_{j=1}^{n} \frac{\#x_i : C'(x_i) = \nu_j}{K} \Omega_j} \quad \forall j=1,\ldots,n.$$

In the third case, the class probability is weighted by the sum of weights associated to all neighbours of that class. The weight of a neighbour is computed as the inverse of its distance from x_q. Normalization by the summation of all weighted class probabilities is required to guarantee that the above measure satisfies properties of a probability measure.

By weighting distances in KNN, there is no harm in allowing all training examples to have an influence on the classification of the x_q, because very distant examples will have very little effect on the class probability estimates. When all training examples are considered when classifying a new test case, the algorithm works as a *global* method, while when the nearest training examples are considered, the algorithm works as a *local* method, since only data local to the area around x_q contribute to the class probabilities. Local methods have significant advantages when the probability measure defined on the space of symbolic objects for each class is very complex, but can still be described by a collection of less complex local approximations.

Therefore, it is clear that the choice of K is critical, since it represents a trade-off between local and global approximations of the probability measures. In order to support the user in the selection of the optimal K, a cross-validation approach is

adopted, where different values of K are considered. As proposed by Gora and Wojna (2002), the search for the optimal *K* can be reduced from the range [1,|*TrainingSet*|] to the range $[1, \sqrt{|TrainingSet|}]$, without loosing too much accuracy in the approximation. For this reason, the proposed algorithm is called *Optimal Local Distance-Weighted Symbolic K-NN* (OLD-SKNN).

Another typical problem of K-NN algorithms is that the distance between cases is calculated on all *p* variables. When most of variables are irrelevant for the task at hand, the distance between neighbours turns out to be dominated by them. One approach to overcoming this problem is to weight each variable differently when calculating the distance between two cases. Some dissimilarity measures between SOs that take into account the weights associated to symbolic variables have already been defined, therefore, they offer a natural solution to this problem, known as the curse of dimensionality.

Finally, it is noteworthy that during the testing phase the extended KNN takes several SOs described by p variables as input and returns as many SOs described by p+1 symbolic variables. This means that the output of the proposed algorithm can be subject to further analysis procedures developed in SDA. In particular, it is possible to apply an extension of Sammon's algorithm (1969) to SOs in order to plot in a bidimensional plane the testing cases for each class.

References

Bock, H.H., Diday, E. (eds. 2000): *Analysis of Symbolic Data. Exploratory Methods for Extracting Statistical Information from Complex Data*, Series: Studies in Classification, Data Analysis, and Knowledge Organisation, Vol. 15, Springer-Verlag, Berlin.

Esposito, F., Malerba, D., Tamma, V. (2000): Dissimilarity Measures for Symbolic Objects. Chapter 8.3 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, vol. 15, pp. 165-185, Springer-Verlag.

Gora, G., Wojna, A.(2002): RIONA: A Classifier Combining Rule Induction and k-NN Method with Automated Selection of Optimal Neighbourhood, *Proceedings of the Thirteenth European Conference on Machine Learning, ECML 2002*, Lecture Notes in Artificial Intelligence, 2430, pp. 111-123, Springer-Verlag.

Malerba, D., Esposito, F., Gioviale, V., Tamma, V. (2001): Comparing Dissimilarity Measures in Symbolic Data Analysis. *Pre-Proceedings of EKT-NTTS*, vol. 1, pp. 473-481.

Malerba, D., Esposito, F., Monopoli, M. (2002). Comparing dissimilarity measures for probabilistic symbolic objects. In A. Zanasi, C. A. Brebbia, N.F.F. Ebecken, P. Melli (Eds.) *Data Mining III*, Series Management Information Systems, Vol 6, pp. 31-40, WIT Press, Southampton, UK.

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