

# Transductive Learning for Spatial Data Classification

Michelangelo Ceci, Annalisa Appice, and Donato Malerba

Dipartimento di Informatica,  
Università degli Studi di Bari,  
Via Orabona 4, I-70126 Bari, Italy  
{ceci, appice, malerba}@di.uniba.it

**Abstract.** Learning classifiers of spatial data presents several issues, such as the heterogeneity of spatial objects, the implicit definition of spatial relationships among objects, the spatial autocorrelation and the abundance of unlabelled data which potentially convey a large amount of information. The first three issues are due to the inherent structure of spatial units of analysis, which can be easily accommodated if a (multi-)relational data mining approach is considered. The fourth issue demands for the adoption of a transductive setting, which aims to make predictions for a given set of unlabelled data. Transduction is also motivated by the contiguity of the concept of positive autocorrelation, which typically affect spatial phenomena, with the smoothness assumption which characterize the transductive setting. In this work, we investigate a relational approach to spatial classification in a transductive setting. Computational solutions to the main difficulties met in this approach are presented. In particular, a relational upgrade of the naïve Bayes classifier is proposed as discriminative model, an iterative algorithm is designed for the transductive classification of unlabelled data, and a distance measure between relational descriptions of spatial objects is defined in order to determine the  $k$ -nearest neighbors of each example in the dataset. Computational solutions have been tested on two real-world spatial datasets. The transformation of spatial data into a multi-relational representation and experimental results are reported and commented.

## 1 Introduction

Several applications, such as fleet management, environmental and ecological modeling, remote sensing, are the source of a huge amount of spatial data, which are stored in spatial databases of Geographic Information Systems (GISs). A GIS is a software system that provides the infrastructure for editing, storing, analyzing and displaying spatial objects. Popular GISs (e.g. ArcView, MapInfo and Open GIS) have been designed as a toolbox that allows planners to explore spatial data by zooming, overlaying, and thematic map coloring. They are provided with functionalities that make the spatial visualization of individual variables effective, but overlook complex multi-variate dependencies. GIS vendors and researchers now recognize this limitation and have begun to address it by adding spatial data interpretation capabilities to the systems.

Wessel [48] describes a prototypical GIS extended with a knowledge-base, expressed in Description Logic, and some reasoning capabilities which can be useful for the interpretation of stored maps. However, this system has a limited range of applicability for a

variety of reasons. First, providing the GIS with operational definitions of some spatial concepts (e.g. morphological environments) is not a trivial task. Generally only declarative and abstract definitions, which are difficult to compile into database queries, are available. Second, the operational definitions of some spatial objects are strongly dependent on the data model adopted for the GIS. Finding relationships between density of vegetation and climate is easier with a raster data model, while determining the usual orientation of some morphological elements is simpler in a topological data model [15]. Third, different applications of a GIS will require the recognition of different spatial objects in a map. Providing the system in advance with all the knowledge required for its various application domains is strenuous.

The solution to these difficulties can be found in *spatial data mining*, which investigates how interesting, but not explicitly available knowledge can be extracted from spatial data. Several works on spatial data mining have already been reported in the literature. They concern several tasks, both predictive (e.g., regression [32] and localization [42]) and descriptive (e.g., clustering [19,40] and discovery of association rules [24,1], co-location [44], subgroups [22], emerging patterns [5], and spatial trends [13]).

In this chapter, we focus on the classification task, where the goal is to produce a model that can predict the class of unlabeled examples, by training on examples whose label, or class, is supplied. Some examples of applications of classification models to spatial data are the recognition of complex geo-morphologies for topographic map interpretation [33] and the land use and land cover mapping on the basis of remotely sensed data [34].

In the literature there are already several works on spatial classification. Ester *et al.* [14] have proposed an extension of decision trees based on neighborhood graph to consider both thematic attributes of the classified objects and relations with neighboring objects. Koperski [23] has investigated a method which deals with both spatial and hierarchical relations between spatial objects, and takes thematic attributes of neighboring objects into account. Shekhar *et al.* [43] compares two major approaches for incorporating spatial dependence into classification/prediction problems, namely the Markov random fields and the spatial autoregression model.

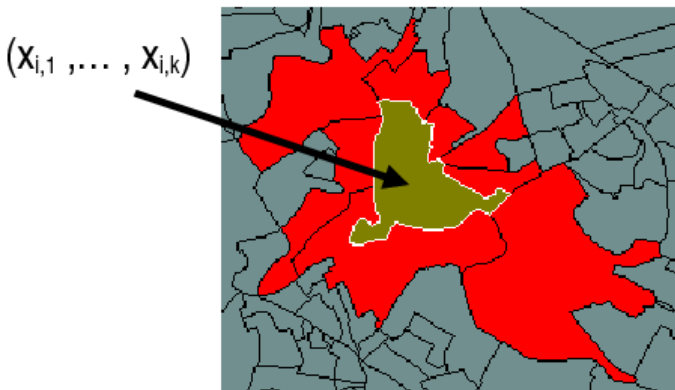
The formulation of a spatial classification method cannot leave out of consideration the following issues.

*Issue 1: Heterogeneity of spatial objects.* Each unit of analysis involves spatial objects of different types, such as a town and a highway. In spatial databases, objects of different types are organized in separate *layers*, each of which has a distinct set of attributes and possibly a geometry attribute represented in the vector mode. For instance, a town can be described in terms of economic and demographic factors, as well as a polygon corresponding to its administrative boundary, while a highway is described by the average speed limit, traffic and driving safety conditions, as well as a polyline corresponding to its path. To deal with object heterogeneity, the design of a spatial classification method should not be strictly bound to process objects in one specific layer.

*Issue 2: The implicit definition of spatial relationships among objects.* Spatial objects have a locational property which implicitly defines several spatial relationships between objects, such as topological, distance-based and directional. Topological relationships

(e.g. ‘intersects’) are invariant under homomorphisms, such as rotation, translation and scaling. Their semantics is precisely defined by means of the 9-intersection model [11]. Distance relationships between two spatial objects can be easily defined in case of punctual representation of the objects (e.g., Euclidean metric) while are more elaborated for linear or surfacic representations. Directional relations can be expressed by the angle formed by two points with respect to the origin of the reference system, or by an extension of Allen’s interval algebra which is based on projection lines [38]. In a spatial database, implicit binary spatial relationships correspond to *spatial joins* between layers. The relational nature of spatial rules makes the computation of these spatial joins crucial for the development of effective and efficient data analysis methods.

*Issue 3: Spatial autocorrelation.* By picturing the spatial variation of some observed variables in a map, we may observe regions where the distribution of values is smoothly continuous with some boundaries possibly marked by sharp discontinuities. In this case, a variable is correlated with itself through space. Formally, *spatial autocorrelation* (or *spatial dependence*, as it is typically called in statistics) is defined as the property of random variables taking values, at pairs of locations a certain distance apart, that are more similar (positive autocorrelation) or less similar (negative autocorrelation) than expected for randomly associated pairs of observations [29]. Informally, spatial positive (negative) autocorrelation occurs when the values of a given property are highly uniform (different) among similar spatial objects in the neighborhood (see Fig. 1). Spatial autocorrelation clearly indicates a violation of the independence assumption of observations usually made in statistics and data mining. Therefore, knowledge discovery techniques that ignore spatial autocorrelation typically perform poorly in the presence of spatial data. When properties of some spatial objects are investigated, attributes of spatially related objects must be taken into account as well, since they may have some kind of influence. This leads to distinguish between the *reference* (or target) *objects*, which are the main subject of analysis and identify the units of analysis, and the



**Fig. 1.** An example of positive autocorrelation. The classification of the brown area as ‘well served’ is based on some features measured in that area (e.g., number of communal establishments (schools, hospitals, etc.) and in the nearby areas.

*task-relevant objects*, which are spatial objects “in the neighborhood” that can help to account for the spatial variation.

*Issue 4: Using both labelled and unlabelled data.* Learning algorithms designed for spatial classification tasks may require large sets of labeled data. However, the common situation is that only few labeled training data are available since manual annotation of the many objects in a map is very demanding. In this situation, it is important to exploit the large amount of information potentially conveyed by unlabeled data to better estimate the data distribution and to build more accurate classification models.

The first three issues reported above are due to the inherent structure of the units of analysis, which cannot be easily accommodated into a classical double-entry table, whose columns correspond to elementary (nominal, ordinal or numeric) single-valued attributes. In fact, spatial datasets can be naturally modeled as a set of relations  $R_1, \dots, R_n$ , such that each relation  $R_i$  has a number of elementary attributes  $A_1^i, \dots, A_{m_i}^i$  and possibly a geometry attribute  $G_i$  (in which case it is a layer). Relationships are either explicitly modeled by means of foreign key constraints or implicitly represented by spatial joins.

(Multi-)relational data mining (MRDM) offers the most suitable setting for spatial data mining tasks. Indeed, MRDM tools can be applied directly to data distributed over several relations to find relational patterns which involve multiple relations [10]. Relational patterns can be expressed either in SQL or in first-order logic (or predicate calculus). The latter representation formalism explains why many MRDM algorithms originate from the field of inductive logic programming (ILP) [37, 8, 28].

The earliest proposal of a relational approach to spatial classification has been reported in [33], where authors propose to exploit the expressive power of first-order logic to represent not only spatial relations, but also background knowledge. Ceci and Appice [3] have subsequently proposed the combination of two relational methods, one for the extraction of spatial association rules and the other for the Bayesian classification. Both proposals, however, do not deal with the problem of learning from both labelled and unlabelled data.

To deal with the fourth issue, two learning settings have been proposed in the literature: the *semi-supervised* setting and the *transductive* setting [41]. The former is a type of inductive learning [35], since the learned function is used to make predictions on any possible example. The latter asks for less - it is only interested in making predictions for the given set of unlabeled data.

Transduction [47] seems to be the most suitable setting for spatial classification tasks, for at least two reasons. First, in spatial domains observations to be classified are already known in advance: they are spatial objects on maps already available in a GIS. Second, transduction is based on a (semi-supervised) smoothness assumption according to which if two points  $x_1$  and  $x_2$  in a high-density region are close, then the corresponding outputs  $y_1$  and  $y_2$  should also be close [6]. In spatial domains, where closeness of points corresponds to some spatial distance measure, this assumption is implied by (positive) spatial autocorrelation.

Several transductive learning methods have been proposed in the literature for support vector machines ([2] [16] [20] [7]), for k-NN classifiers ([21]) and even for general classifiers ([27]). However, all of these transductive learning algorithms assume (un)labeled input examples are represented as rows of a classical double-entry table (or

database relation). As previously said, this tabular representation of data, also known as *propositional* or *feature-vector* representation, turns out to be too restrictive for spatial classification tasks.

These considerations motivate the investigation of a transductive setting for spatial classification which is based on a relational approach. To devise an algorithm for transductive learning from spatial data represented in as a set of relations  $R_1, \dots, R_n$ , we face several problems. First, we have to choose a strategy for the classification of unlabeled data and make it suitable for relational data. Second, we have to choose a classifier which can handle relational data representing different spatial objects and their spatial relationships. Third, we have to define a distance measure between examples described by several relations in order to define a meaningful neighborhood of an object. Solutions to these problems are described in the following sections. They have been implemented in a new classification algorithm based on an extension of the naïve Bayes classifier. Information on the potential uncertainty of classification conveyed by naïve Bayes classifiers is especially useful to prevent drastic changes in the prediction due to small changes in the attribute values of a test case. It is also useful when missing (or imprecise) information may prevent a new object from being classified at all.

This paper is organized as follows. The next section presents both the background of this work and the most relevant related literature. Section 3 introduces a formalization of the transductive learning problem and describes the proposed solution, which is based on an iterative  $k$ -NN based re-classification of labelled and unlabelled examples in order to identify “borderline” cases, i.e. examples for which the classification is more uncertain. Experimental results on two real-world spatial datasets are reported and discussed in Section 4. Finally, Section 5 concludes and presents ideas for further work.

## 2 Background and Related Work

For spatial classification, two methods have been reported in the literature which resort to a relational approach. Malerba et al. [33] proposed to exploit the expressive power of predicate logic to represent both spatial relations and background knowledge. For this purpose, the ILP system ATRE [30] has been integrated in the data mining server of a prototypical GIS, named INGENS, which allows, among other things, to mine classification rules for geographical objects stored in an object-oriented database. Training is based on a set of examples and counterexamples of geographic concepts of interest to the user (e.g., ravine or steep slopes). The symbolic representation of the training examples is automatically extracted from maps, although it is still controlled by the user who can select a suitable level of abstraction and/or aggregation of data by means of a data mining query language [31].

The main weakness of this approach is that learned rules are quite brittle, since conditions in the body must perfectly match against the object description in order to classify it. Ceci and Appice [3] have investigated an associative classification framework, where association rules discovered from training datasets are used by a naïve Bayes classifier which operates on relational representations of spatial data.

Given an object  $E$  to be classified, a classical naïve Bayes classifier assigns  $E$  to the class  $C_i$  that maximizes the *posterior probability*  $P(C_i|E)$ . By applying the Bayes theorem,  $P(C_i|E)$  is expressed as follows:

$$P(C_i|E) = \frac{P(C_i) \cdot P(E|C_i)}{P(E)}. \quad (1)$$

In fact, the decision on the class that maximizes the posterior probability can be made only on the basis of the numerator, that is  $P(C_i) \cdot P(E|C_i)$ , since  $P(E)$  is independent of the class  $C_i$ .

To work on relational representations, Ceci and Appice proposed to consider a set  $\mathfrak{R}$  of association rules, expressed as first order definite clauses, which are mined on the training set and can be used to define a suitable decomposition of the likelihood  $P(E|C_i)$  *à la* naïve Bayes in order to simplify the probability estimation problem. In particular, if  $\mathfrak{R}(E) \subseteq \mathfrak{R}$  is the set of first order definite clauses whose body covers  $E$ , the probability  $P(E|C_i)$  is defined as follows:

$$P(E|C_i) = P\left(\bigwedge_{R_j \in \mathfrak{R}(E)} \text{antecedent}(R_j)|C_i\right). \quad (2)$$

The straightforward application of the naïve Bayes independence assumption to all literals in  $\bigwedge_{R_j \in \mathfrak{R}(E)} \text{antecedent}(R_j)$  is not correct, since it may lead to underestimating  $P(E|C_i)$  when several similar clauses in  $\mathfrak{R}(E)$  are considered for the class  $C_i$ . To prevent this problem the authors resort to the logical notion of factorization [39]. Details are reported in [3].

For relational classification in the transductive setting, two methods have been reported in the literature. Krogel and Scheffer ([26]) investigate a transformation (known as *propositionalization*) of a relational description of gene interaction data into a classical double-entry table, and then study transduction with the well-known transductive support vector machines. Therefore, transduction is not explicitly investigated on relational representations and it is based on propositionalization, which is fraught with many difficulties in practice ([9, 17]). Taskar *et al.* ([45]) build, on the framework of Probabilistic Relational Models, a *generative* probabilistic model which captures interactions between examples, either labelled or unlabelled. However, given sufficient data, a *discriminative* model generally provides significant improvements in classification accuracy over generative models ([46]). This motivates our interest in designing classifiers based on discriminative models.

For spatial classification, there is no work, at best of our knowledge, concerning a transductive setting. Only recently, some work on semi-supervised learning for spatial classification has been reported [43]. Therefore, we intend this contribution to be a further step toward the investigation of methods which originate from the intersection of these three promising research areas, namely transduction, relational data mining and spatial data mining.

### 3 Problem Formalization and Proposed Solution

The transductive spatial classification problem can be formalized as follows:

*Given*

- a spatial database (*SDB*),
- a set  $S$  of reference spatial objects

- a set  $TS \subseteq S$  of reference spatial objects tagged with a class label  $Y \in \{C_1, \dots, C_L\}$  (training reference objects),
- a set  $WS = S - TS$  of unlabelled reference spatial objects (working reference objects),
- some sets  $R_k$ ,  $1 \leq k \leq m$  of task-relevant spatial objects

*Find:* the most accurate prediction of  $Y$  for each reference spatial object in  $WS$ .

Informally, the learner receives full information (including labels) on the reference spatial object in  $TS$  and partial information (without labels) on the reference spatial object in  $WS$  and is required to predict the class values only of the examples in  $WS$ . The original formulation of the problem of function estimation in a transductive (*distribution-free*) setting requires  $TS$  to be sampled from  $S$  without replacement. This means that, unlike the standard inductive setting, the examples in the training (and working) set are supposed to be mutually dependent. Vapnik also introduced a second (*distributional*) transduction setting, in which the learner receives training and working sets, which are assumed to be drawn i.i.d. from some unknown distribution. As shown in [47](Theorem 8.1), error bounds for learning algorithms in the distribution-free setting also apply to the more popular distributional transductive setting. Therefore, also considering the spatial autocorrelation issue, in this work we focus our attention on the first setting.

In order to take advantage of results obtained in the area of MRDM, we boil down the spatial database  $SDB$  to a relational database  $D$  with schema  $\Sigma$ . This requires the materialization of spatial queries which compute complex and massive spatial joins (or operators) between reference objects and task relevant objects. For example, spatial joins operations are used to compute the “interact” (non disjoint) relationship between spatial objects (either reference or task-relevant) and the result is stored in database relations. Hence, the schema  $\Sigma$  includes additional relations with respect to those originally present in  $SDB$  in order to model these spatial relationships.

Reference objects, task-relevant objects and (spatial) interactions among them are tuples stored in separate relations of  $D$ . In particular,  $D$  includes a *target* relation  $T$  whose tuples represent the reference objects in  $S$ . Each set  $R_i$  is stored in a distinct relation of  $D$ . The interaction between different objects is materialized in new relations of  $\Sigma$  (e.g., the relation “Towns\_CrossedBy\_Roads” describes the instances of the crossing relation between towns and roads). Foreign key constraints ( $FK$ ) state how tuples in one relation relate to tuples in another and define the inherent *spatial structure* of data. For example, the relation “Towns\_CrossedBy\_Roads” has two foreign keys: one for the relation “Towns” and one for the relation “Roads”. Foreign keys can be used to retrieve all task-relevant objects of  $D$  which are (spatially) related to a reference object and are relevant to discriminate between the values of  $Y$  (target attribute). Indeed, an example in  $S$  is represented as one tuple  $t \in T$  and all tuples related to  $t$  in  $D$  according to  $FK$ .

### 3.1 The Transductive Learning Algorithm

The core of the transductive learning method is described in Algorithm 1. In keeping with the main idea expressed in [20], the algorithm iteratively refine the classification by changing the class of “borderline” training and working examples, i.e., of those

examples whose classification is more uncertain. It starts with a given classification and, at each iteration, alternates a step during which examples are reclassified and a step during which the class of borderline examples is changed.

---

**Algorithm 1.** Top level transductive algorithm description
 

---

```

1: transductiveClassifier(initialClassification, TS, WS)
2: classification1  $\leftarrow$  initialClassification;
3: changedExamples  $\leftarrow$   $\phi$ ;
4: i  $\leftarrow$  0;
5: repeat
6:   prevClassification  $\leftarrow$  classification1;
7:   prevChangedExamples  $\leftarrow$  changedExamples;
8:   classification2  $\leftarrow$  reclassifyExamplesKNN(classification1, TS, WS);
9:   (classification1, changedExamples)  $\leftarrow$  changeClass(classification2);
10: until ( ++i  $\geq$  MAX_ITER) OR
      (computeOverlap(prevChangedExamples, changedExamples)  $\geq$  MAXOVERLAP)
11: return prevClassification
  
```

---

The initial classification of  $E \in WS \cup TS$  is obtained according to the classification function *preclass* defined as follows:

$$preclass(E) = \begin{cases} class(E) & \text{if } E \in TS \\ BayesianClassification(E) & \text{if } E \in WS, \end{cases}$$

where *BayesianClassification*( $E$ ) is the initial probabilistic classifier built from the training set  $TS$  according to formula 2.

The examples are then reclassified by means of a version of the  $k$ -NN algorithm [36], tailored for transductive inference in MRDM. The idea is to classify each example  $E \in \{TS \cup WS\}$  on the basis of a  $k$ -sized neighborhood  $N_k(E) = \{E_1, \dots, E_k\}$ , consisting of the  $k$  examples of  $TS \cup WS$  closest to  $E$  with respect to a dissimilarity measure  $d$ . More precisely, each example  $E$  is associated with an  $L$ -dimensional class probability vector  $\mathbf{y}' = (y_1(E), \dots, y_L(E))$ , where  $y_i(E)$  estimates  $P(class(E) = C_i)$ . Estimation is based of the following formula:

$$P(class(E) = C_i) = \frac{|\{E_j \in N_k(E) | C_{E_j} = C_i\}|}{k}, \quad (3)$$

where  $C_{E_j}$  is the class value associated to  $E_j$  at the previous step (at the first step,  $C_{E_j}$  is the class label returned by *preclass*( $E_j$ )). The following properties hold for this estimate:

- i)  $P(class(E) = C_i) \geq 0$  for each  $i = 1, \dots, L$ ;
- ii)  $\sum_{i=1, \dots, L} P(class(E) = C_i) = 1$ ;
- iii)  $P(class(E) = C_i)$  is estimated according to the transductive inference principle, as both training and working examples are taken into account in its computation.



The *changeClass* procedure is in charge of changing the classification of the borderline examples. Unlike what was originally proposed in [20], where examples on the border are identified by means of support vectors, we consider the examples for which the entropy of the decision made by the classifier is maximum. The entropy of an example  $E$  is computed from the probabilities associated with each class  $C_i$ :

$$Entropy(E) = - \sum_{i=1, \dots, L} P(class(E) = C_i) \times \log(P(class(E) = C_i)). \quad (4)$$

Examples are ordered according to the entropy function and the class label of at most the first  $k$  examples, having  $Entropy(E) > MINENTROPY$ , is changed. In particular, each selected example  $E$  is assigned the most likely class  $C_i$  for  $E$  among those remaining after the the old class of  $E$  has been excluded. The threshold  $k$ , which is the same used for  $k$ -NN, is necessary in order to avoid changing the class of too many examples, in the extreme case, the class of all examples in a cluster.

Two distinct stopping criteria are used. The first criterion stops the execution of the algorithm when the maximum number of iterations (*MAX\_ITERES*) is reached. This guarantees the termination of the algorithm. In any case, our experiments showed that this criterion is rarely attained when the parameter *MAX\_ITERES* is as small as 10. The second criterion stops execution when a cycle processes the same examples as the previous one. For this purpose, the overlap between two sets of examples is determined. The *computeOverlap* function returns the ratio between the cardinality of the intersection between the sets of examples and that of their union.

### 3.2 Generating the First Order Definite Clauses

Differently from the original work in [3], the set  $\mathfrak{R}$  of first order definite clauses used in formula 2 do not correspond to association rules. They are directly constructed by means of a depth-bounded breadth-first search in the space of definite clauses built by navigating foreign key paths. A *foreign key path* is an ordered sequence of relations  $\vartheta = \{T_{i_1}, T_{i_2}, \dots, T_{i_s}\}$ , such that  $T_{i_j} \in \Sigma$  ( $j = 1 \dots s$ ) and each relation  $T_{i_j}$  has a foreign key to the relation  $T_{i_{j-1}}$  ( $j = 2, \dots, s$ ) or  $T_{i_{j-1}}$  has a foreign key to relation  $T_{i_j}$ . To formally define the set of first order definite clauses  $\mathfrak{R}$  in terms of foreign key paths, some definitions are introduced.

**Definition 1 (Structural predicate).** A binary predicate  $p$  is a structural predicate associated to the pair of relations  $\langle T_i, T_j \rangle \in S$  if a foreign key in  $T_i$  exists which references a relation  $T_j \in S$ , or vice-versa. The first argument of  $p$  represents the primary key of  $T_j$  and the second argument represents the primary key of  $T_i$ , or vice-versa.

**Definition 2 (Property predicate).** A binary predicate  $p$  is a property predicate associated to a relation  $T_i \in S$  and an attribute *Att* of  $T_i$  if the first argument of  $p$  represents the primary key of  $T_i$  and the second argument represents a value observed for *Att* in  $T_i$ . The attribute *Att* is neither the primary key of  $T_i$  nor a foreign key in  $T_i$ .

Hence, we can formally define a first order definite clause associated to a foreign key path.

**Definition 3 (First order definite clause associated to a foreign key path).** A first order definite clause associated to the foreign key path  $\vartheta = \{T_{i_1}, \dots, T_{i_{s-1}}\}$  is a clause in the form:

$$p_0(A_1, y) \leftarrow p_1(A_1, A_2), p_2(A_2, A_3), \dots, p_{s-1}(A_{s-1}, A_s), p_s(A_s, c).$$

or

$$p_0(A_1, y) \leftarrow p_1(A_1, A_2), p_2(A_2, A_3), \dots, p_{s-1}(A_{s-1}, A_s).$$

where:

1.  $p_0$  is a property predicate associated to both the target relation  $T$  and the target attribute  $Y$ .
2.  $p_1$  is a structural predicate associated to the target relation  $T$  and relation  $T_{i_1}$  of  $\vartheta$ .
3.  $p_k$ ,  $k = 2, \dots, s-1$ , is a structural predicate associated to the pair of relations  $\langle T_{i_{k-1}}, T_{i_k} \rangle$  of  $\vartheta$ .
4.  $p_s$  is an optional property predicate associated to both relation  $T_{i_{s-1}}$  and an attribute  $Att$  of  $T_{i_{s-1}}$ .

The set  $\mathfrak{R}$  is constructed by searching the first order definite clauses  $R_j$  associated with a foreign key path  $\vartheta$ , such that the antecedent of  $R_i$  covers at least one training example. The length of the foreign key path  $\vartheta$  is upper bounded by a user-defined constant `MAX_LENGTH_PATH`.

The property predicate  $p_s$  is associated to either a discrete attribute or a continuous attribute  $Att$  of the relation  $T_{i_{s-1}}$ . In the former case,  $p_s$  checks a condition in the form “ $Att = v$ ”, where  $v$  is a value in the range of  $Att$ , while in the latter case,  $p_s$  checks a condition in the form “ $Att \in [v_1, v_2]$ ”, where  $[v_1, v_2]$  denotes an interval.

### 3.3 The Relational Dissimilarity Measure

The re-classification of training and working examples is based on a dissimilarity measure  $d$ . The classical  $k$ -NN method assumes that examples correspond to points in the  $m$ -dimensional space  $\mathbb{R}^m$  and the nearest neighbors of the example to classify are defined in terms of the standard Euclidean distance. However, in the proposed relational algorithm, examples cannot be associated to points of  $\mathbb{R}^m$ . This motivates the need for a different notion of a distance (dissimilarity) measure that applies to relational data.

The dissimilarity measure between each pair of examples  $E_1$  and  $E_2$  is computed by first converting the first order definite clauses in  $\mathfrak{R}$  into a set of boolean features and then using these features as input of some propositional dissimilarity measure. Actually, the conversion into boolean features concerns only those first order definite clauses  $R_j \in \mathfrak{R}$  whose property predicate  $p_s$  is defined on a continuous attribute  $Att$ . In particular, the range of values of  $Att$  is discretized<sup>1</sup> into  $Nb$  disjoint intervals (or bins), where  $Nb$  is a user-defined parameter, and the predicate  $p_s$  is defined on the basis of the returned sets of bins.

<sup>1</sup> In this work, we use an equal-width discretization strategy.

Special care is needed for this conversion. Indeed, if we let  $p_s$  simply modeling the condition  $T_{i_{s-1}}.Att \in [v_1, v_2]$ , where  $[v_1, v_2]$  is a bin, we incur in a harmful information loss on the order relation of continuous values. To overcome this problem, we follow the idea formulated in [12] and transform the rule  $R_j$  into  $R'_j$ , such that:

1.  $p'_i = p_i$ , for each  $i = 0, \dots, s-1$ , while
2. the property predicate  $p'_s$  expresses the condition  $T_{i_{s-1}}.Att \leq v_2$ .

The advantage of  $R'_j$  with respect to  $R_j$  is that  $R'_j$  models as closer two examples  $E_1$  and  $E_2$ , whose  $Att$  values belong to two consecutive bins, rather than two examples, whose  $Att$  values belong to distant bins.

Once the new set  $\mathfrak{R}' = \{R'_j\}$  is constructed, boolean features are derived from  $\mathfrak{R}'$ , in order to represent examples by means of a single relation  $V$ . This is a form of propositionalization [25] which allows us to use dissimilarity measures defined for classical propositional representations. The schema of  $V$  includes  $|\mathfrak{R}'|$  attributes, that is one for each first order definite clause  $R'_j$ . Each row of  $V$  corresponds to an example  $E \in \{TS \cup WS\}$ . If the antecedent of the first order clause  $R'_j$  covers  $E$ , then the  $j$ -th value of the row in  $V$  corresponding to  $E$  is set to *true*, *false* otherwise.

The similarity between the pair of examples  $E_1$  and  $E_2$  can be computed by means of the Kendall, Sokal-Michener similarity measure [12], computed on their feature vector representation  $V(E_1)$  and  $V(E_2)$  stored in  $V$ , that is:

$$s(E_1, E_2) = \frac{\text{cardinality}(V(E_1) \text{ XNOR } V(E_2))}{|\mathfrak{R}'|}, \quad (5)$$

where  $\text{cardinality}(\bullet)$  returns the number of *true* values occurring in the input boolean vector. The similarity coefficient computed in Equation (5) takes values in the unit interval. In particular,  $s(E_1, E_2) = 1$ , if the two vectors match perfectly, while  $s(E_1, E_2) = 0$ , if the two vectors are orthogonal. The dissimilarity between the pair of examples is then computed as follows:

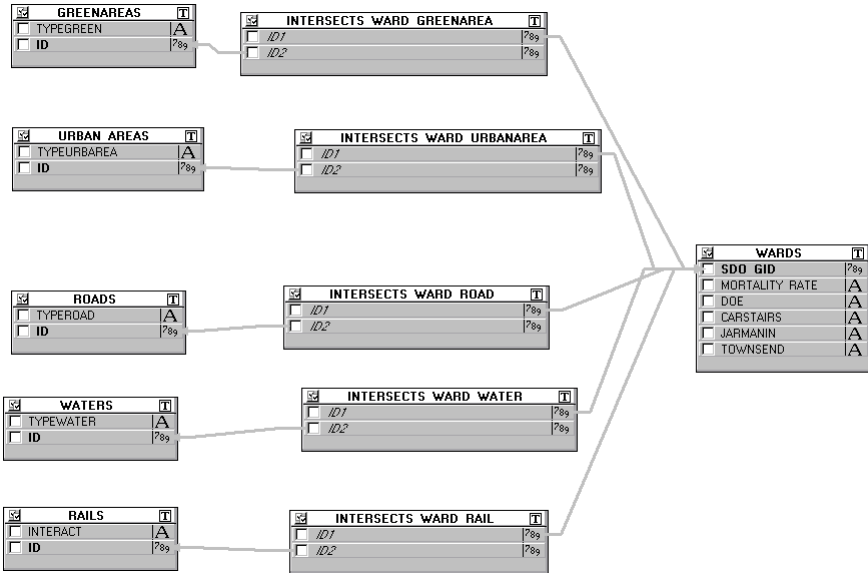
$$d(E_1, E_2) = 1 - s(E_1, E_2). \quad (6)$$

## 4 Experimental Results

In this section, we evaluate the proposed approach on two real-world spatial datasets concerning Greater Manchester (UK) and Munich (Germany). A description of the two datasets is reported below.

*Dataset 1: Greater Manchester.* Data have been collected within the context of the European project SPIN!<sup>2</sup> They concern Greater Manchester, one of the five counties of North West England. Greater Manchester is divided into 10 metropolitan districts, each of which is in turn decomposed into census sections (wards), for a total of 214 wards. Census data are available at ward level and provide socio-economic statistics (e.g. mortality rate) as well as some measures of the deprivation of each ward according to

<sup>2</sup> Web site of the project: <http://www.ais.fraunhofer.de/KD/SPIN/project.html>



**Fig. 2.** Schema of the relational database storing data on deprived areas of Greater Manchester

information provided by Census combined into single index scores. We consider the Jarman Underprivileged Area Score (designed to estimate the need for primary care), the indices developed by Townsend and Carstairs (used to perform health-related analyses), and the Department of the Environment’s (DoE) index (used in targeting urban regeneration funds). The higher the indices values the more deprived the ward. Layers available concern the boundaries of the 1998 census wards as well as the urban areas (115 lines), the green areas (9 lines), the road network (1,687 lines), the rail network (805 lines) and the water network (716 lines). The original spatial database boils down to a relational database where spatial objects of each layer are stored as single tuples of a database relation together with information on the object type. For instance, type information of an urban area can be either “large urban area” or “small urban area”. Topological relationships between wards and objects in all these layers are materialized as database relations. The number of instances of the “non disjoint” spatial relationship is 5,313. The schema of the relational database is reported in Fig. 2. In this work, the reference spatial objects are the wards, that is, *WARDS* is the target relation, the task-relevant objects are all spatial objects in the six layers described above, and the goal of the classification task is to predict a dichotomized value of the Jarman index (low=[0.001, 0.01] or high]=[0.01,0.18]), which defines the class label  $Y$ . Information used for prediction purposes is given by other deprivation indices (Townsend index, Carstairs index and DoE index), the mortality rate and spatial objects that geometrically intersects with wards. An assumption of positive spatial autocorrelation seems sensible for this dataset, since deprived wards tend to spatially cluster.

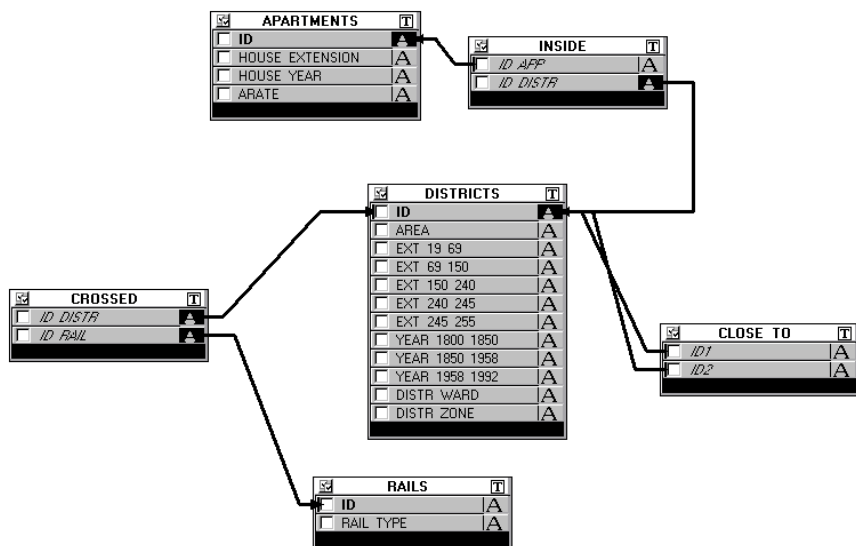


Fig. 3. Schema of the relational database storing rental data on Munich

*Dataset 2: Munich.* This dataset<sup>3</sup> has been collected in 1998 by Infratest Sozialforschung to develop the 1999 Munich rental guide. It concerns the monthly rent (expressed in German Marks) per square meter for flats in Munich. In particular, the data describe 2,180 geo-referenced flats located in the 446 subquarters of Munich obtained by first dividing the Munich metropolitan area up into three areal zones and then by decomposing each of these zones into 64 districts. The vectorized boundaries of districts as well as the map of public train stops (56 subway (U-Bahn) stops, 15 rapid train (S-Bahn) stops and 1 railway station) within Munich are available for this study. Trains stops are distinguished into three types, namely U-Bahn, S-Bahn or railway station, while flats are described by means of their “monthly rent per square meter”, “floor space in square meters” and “year of construction”. The schema of the relational database derived from the original spatial database is reported in Fig. 3. In this work, the reference spatial objects are the flats, that is, *APARTMENTS* is the target relation, the task relevant objects are both the districts and the train stops, and the goal of the classification task is to predict a dichotomized value of the ‘monthly rent per square meter’ (low = [2.0,14.0] or high=[14.0,35.0]), which defines the class label  $Y$ . An assumption of positive spatial autocorrelation is sensible for this dataset as well. The spatial arrangement of data is defined by both the “close.to” relation between Munich metropolitan districts, the “inside” relation between flats and districts, and the “inside” relation between public train stops and districts. All these topological relations are materialized into as many database relations.

The experiments aimed at validating the actual advantage of the transductive algorithm over the basic inductive algorithm when few labeled examples are available. The

<sup>3</sup> Available at [http://www.di.uniba.it/~ceci/mic Files/munich\\_db.tar.gz](http://www.di.uniba.it/~ceci/mic Files/munich_db.tar.gz)

basic inductive algorithm is that used for the initial classification of unlabelled examples (see line 2 of Algorithm 1). The empirical comparison is based on the misclassification error of the two algorithms. To estimate the misclassification error rate, a  $K$ -fold cross validation is performed. For each trial, both algorithms are trained on a single fold and tested on the hold-out  $K - 1$  folds, which form the working set. The comparative statistics is computed by averaging the misclassification error over the  $K$ -folds. The construction of the  $K$  folds is a two-stepped process. First, the target relation is partitioned into  $K$  blocks  $B_1, \dots, B_K$  of nearly-equal size. Then, for each block  $B_i$ , the subset of tuples in other non-target relations which are related, according to the foreign key constraints, to the tuples in  $B_i$  are extracted.

It is noteworthy that, unlike the standard cross-validation approach, here only one fold is used for the training set. In this way we can simulate datasets with a small percentage of labelled cases (the training set) and a large percentage of unlabelled data (the working set), which is the usual situation for a transductive setting. Therefore, high misclassification error rates with respect to those published in previous works [3, 4] should not be surprising, due to the small amount of labelled examples that we consider.

Since the performance of the transductive algorithm may vary significantly depending on the size ( $k$ ) of the neighborhood, experiments for different  $k$  are performed in order to set the optimal value. In theory, we should experiment with each value of  $k$  ranging in the interval  $[1, N]$ , where  $N$  is the number of reference objects in  $TS \cup WS$ . However, as observed in [49], it is not necessary to consider all possible values of  $k$  during cross-validation to obtain the best performance. This can be well approximated by means of cross-validation on no more than ten values of  $k$ . A similar consideration has also been reported in [18], where it is shown that the search for the optimal  $k$  can be substantially reduced from  $[1, N]$  to  $[1, \sqrt{N}]$ , without degrading accuracy of learning algorithm too much. Hence, we have decided to consider in our experiments only those values of  $k$  such that  $k \in \{\eta i | i = 1, \dots, q\}$ , where  $\eta = \sqrt{N}/q$  is the step value and  $q$  is the number of steps.

The classifiers mined in this study are obtained by setting  $MAX\_ITERS=10$ ,  $MAX\_LENGTH\_PATH=3$ ,  $MINENTROPY=0.65$  and  $MAXOVERLAP=0.5$ . The step value  $\eta$  differs for each dataset.

The error rates of the transductive algorithm and its inductive counterpart on the two datasets are reported in Tables 1 and 2. The results are obtained according to both a 10-fold cross validation (CV) of the data and a 20-fold CV of the same data. For experiments on Greater Manchester data, we set  $k \in \{4, 7, 9, 11, 14\}$ , while for experiments on Munich data we set  $k \in \{9, 18, 27, 36, 45\}$ . The percentage of error loss refers to the difference of the error rate of the transductive algorithm with respect to the inductive algorithm. A positive (negative) value is in favor of the transductive (inductive) algorithm.

In both datasets, results confirm a lower error rate for the transductive setting with respect to the inductive one. The improvement depends on the  $k$  value and this result is more evident in the case of 10-fold CV. In 20-fold CV, there is an error propagation through algorithm iterations due to the presence of few training examples. A deeper analysis of results of 10-fold CV reveals that the error curve is  $U$ -shaped: it first decreases (the minimum is obtained when  $k=11$  for dataset 1 and  $k=36$  for dataset 2), and

**Table 1.** Dataset 1: Transductive vs. inductive algorithm

Experiment	Transductive					Inductive
	k=4	k=7	k=9	k=11	k=14	
Avg 10-CV Error (%)	23.38	21.10	19.79	18.04	19.08	22.71
<i>error loss (%)</i>	-2.97	7.06	12.84	20.56	15.99	
Avg 20-CV Error (%)	33.87	34.41	33.82	33.20	33.28	34.31
<i>error loss (%)</i>	0.00	-1.60	0.15	1.96	1.75	

**Table 2.** Dataset 2: Transductive vs. inductive algorithm

Experiment	Transductive					Inductive
	k=9	k=18	k=27	k=36	k=45	
Avg 10-CV Error (%)	28.99	28.61	28.36	28.30	28.15	31.23
<i>error loss (%)</i>	7.17	8.41	9.19	9.40	9.86	
Avg 20-CV Error (%)	37.25	36.30	36.73	36.67	36.78	37.79
<i>error loss (%)</i>	1.44	3.94	2.81	2.98	2.68	

then slightly increases when  $k$  approximates  $\sqrt{|N|}$ . This poses the problem of automatically determining the best  $k$  value.

Finally, we observe that the error rate is also affected by the number of bins for the discretization of continuous values. In these experiments the number of bins ( $N_b$ ) for discretization is set to 10 for dataset 1, and 40 for dataset 2. Some results, which we omit here due to space limitations, empirically prove that the higher the error loss the lower the number of bins.

## 5 Conclusions

In this work we have investigated the main issues met when facing a spatial classification learning problem. The search for a solution to these issues has motivated the proposal of a relational approach to spatial classification in a transductive setting. Indeed, our proposal is to transform the original spatial database into a relational database with multiple relations, which represent both the reference objects (i.e., the main subject of analysis), and the related task-relevant objects and their spatial relationships. The target relation associated to the reference objects includes the class attribute, whose value is null for unlabelled objects. A distinct database relation is associated to each type of spatial object and to each type of spatial relationship (topological, distance-based or directional) explicitly computed on the original spatial database.

An initial probabilistic classifier is induced from these relational data by processing only training (i.e., labeled) examples. The inductive learning algorithm is a relational

upgrade of the naïve Bayes classifier and provides a discriminative model on the basis of which unlabelled reference objects (i.e., examples in the working set) can be preliminarily classified. This initial classification of the examples in the working set is then iteratively refined by a transductive learning algorithm. Each iteration consists in a k-NN classification of unlabeled examples and a subsequent reclassification of some “borderline” examples, i.e. examples for which the classification is more uncertain. Uncertainty is measured as the entropy of the class probabilities.

Neighbors are determined by computing a distance measure on a propositionalized representation of unlabelled examples. Propositional features are obtained by transforming a set of first order definite clauses into boolean features. Definite clauses are directly constructed by means of a depth-bounded breadth-first search in the space of definite clauses built by navigating foreign key paths.

The transductive classifier has been compared to its inductive counterpart on two spatial datasets. Experimental results are in favor of the transductive algorithm. The improvement in accuracy seems more significant than that observed by Joachims [21] who compared SVMs in both the inductive and transductive setting.

We believe that our investigation indicates that learning in a transductive setting is an important direction for further research in spatial classification, since there is a clear contiguity of the concept of positive autocorrelation, which typically affect spatial phenomena, with the smoothness assumption which characterize the transductive setting. In principle, we expect that a strong spatial autocorrelation should counterbalance the lack of labelled data, if a relational classifier which takes spatial autocorrelation into account is learned in a transductive setting. As future work, we intend to extend the empirical investigation in order to corroborate our intuition.

**Acknowledgements.** This work is in partial fulfillment of research objectives of the Strategic Project PS121 “Telecommunication Facilities and Wireless Sensor Networks in Emergency Management” funded by “Regione Puglia” as well as of the ATENEO-2008 project “Knowledge Discovery in Relational Domains” funded by the University of Bari, Italy.

## References

1. Appice, A., Ceci, M., Lanza, A., Lisi, F.A., Malerba, D.: Discovery of spatial association rules in georeferenced census data: A relational mining approach. *Intelligent Data Analysis* 7(6), 541–566 (2003)
2. Bennett, K.P.: Combining support vector and mathematical programming methods for classification. In: Schölkopf, B., Burges, C.J.C., Smola, A.J. (eds.) *Advances in kernel methods: support vector learning*, pp. 307–326. MIT Press, Cambridge (1999)
3. Ceci, M., Appice, A.: Spatial associative classification: propositional vs. structural approach. *Journal of Intelligent Information Systems* 27(3), 191–213 (2006)
4. Ceci, M., Appice, A., Malerba, D.: Mr-SBC: A multi-relational naïve bayes classifier. In: Lavrač, N., Gamberger, D., Todorovski, L., Blockeel, H. (eds.) *PKDD 2003. LNCS (LNAI)*, vol. 2838, pp. 95–106. Springer, Heidelberg (2003)



5. Ceci, M., Appice, A., Malerba, D.: Discovering emerging patterns in spatial databases: A multi-relational approach. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) PKDD 2007. LNCS (LNAI), vol. 4702, pp. 390–397. Springer, Heidelberg (2007)
6. Chapelle, O., Schölkopf, B., Zien, A. (eds.): Semi-supervised learning. MIT Press, Cambridge (2006)
7. Chen, Y., Wang, G., Dong, S.: Learning with progressive transductive support vector machines. *Pattern Recognition Letters* 24, 1845–1855 (2003)
8. De Raedt, L.: Interactive theory revision. Academic Press, London (1992)
9. De Raedt, L.: Attribute-value learning versus inductive logic programming: the missing links. In: Page, D.L. (ed.) ILP 1998. LNCS, vol. 1446, pp. 1–8. Springer, Heidelberg (1998)
10. Džeroski, L., Lavrač, N.: Relational data mining. Springer, Berlin (2001)
11. Egenhofer, M.J., Franzosa, R.: Point-set topological spatial relations. *International Journal of Geographical Information Systems* 5(2), 161–174 (1991)
12. Esposito, F., Malerba, D., Tamma, V., Bock, H.: Similarity and dissimilarity. In: Bock, H.H., Diday, E. (eds.) *Analysis of Symbolic Data: Exploratory Methods for Extracting Statistical Information from Complex Data*, pp. 139–152. Springer, Heidelberg (2000)
13. Ester, M., Gundlach, S., Kriegel, H., Sander, J.: Database primitives for spatial data mining. In: *Proceedings of the International Conference on Database in Office, Engineering and Science, BTW 1999*, Freiburg, Germany (1999)
14. Ester, M., Kriegel, H., Sander, J.: Spatial data mining: A database approach. In: Scholl, M.O., Voisard, A. (eds.) *SSD 1997*. LNCS, vol. 1262, pp. 47–66. Springer, Heidelberg (1997)
15. Frank, A.: Spatial concepts, geometric data models, and geometric data structures. *Computers and Geosciences* 18(4), 409–417 (1992)
16. Gamberman, A., Azoury, K., Vapnik, V.: Learning by transduction. In: *Proc. of the 14th Annual Conference on Uncertainty in Artificial Intelligence, UAI 1998*, pp. 148–155. Morgan Kaufmann, San Francisco (1998)
17. Getoor, L.: Multi-relational data mining using probabilistic relational models: research summary. In: Knobbe, A., Van der Wallen, D.M.G. (eds.) *Proceedings of the 1st Workshop on Multi-Relational Data Mining*, Freiburg, Germany (2001)
18. Góra, G., Wojna, A.: RIONA: A classifier combining rule induction and k-NN method with automated selection of optimal neighbourhood. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) *ECML 2002*. LNCS (LNAI), vol. 2430, pp. 111–123. Springer, Heidelberg (2002)
19. Han, J., Kamber, M., Tung, A.K.H.: Spatial clustering methods in data mining. In: Han, J., Kamber, M. (eds.) *Geographic Data Mining and Knowledge Discovery*, pp. 188–217. Taylor and Francis, Abington (2001)
20. Joachims, T.: Transductive inference for text classification using support vector machines. In: *Proceedings of the 16th International Conference on Machine Learning (ICML 1999)*, pp. 200–209. Morgan Kaufmann, San Francisco (1999)
21. Joachims, T.: Transductive learning via spectral graph partitioning. In: *Proc. of the 20th International Conference on Machine Learning, ICML 2003*. Morgan Kaufmann, San Francisco (2003)
22. Klösgen, W., May, M.: Spatial subgroup mining integrated in an object-relational spatial database. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) *PKDD 2002*. LNCS (LNAI), vol. 2431, pp. 275–286. Springer, Heidelberg (2002)
23. Koperski, K.: Progressive Refinement Approach to Spatial Data Mining. PhD thesis, Computing Science, Simon Fraser University, British Columbia, Canada (1999)
24. Koperski, K., Han, J.: Discovery of spatial association rules in geographic information databases. In: Egenhofer, M.J., Herring, J.R. (eds.) *SSD 1995*. LNCS, vol. 951, pp. 47–66. Springer, Heidelberg (1995)

25. Krogel, M., Rawles, S., Zelezny, F., Flach, P., Lavrac, N., Wrobel, S.: Comparative evaluation of approaches to propositionalization. In: Horváth, T., Yamamoto, A. (eds.) *ILP 2003*. LNCS (LNAI), vol. 2835, pp. 197–214. Springer, Heidelberg (2003)
26. Krogel, M.-A., Scheffer, T.: Multi-relational learning, text mining, and semi-supervised learning for functional genomics. *Machine Learning* 57(1-2), 61–81 (2004)
27. Kukar, M., Kononenko, I.: Reliable classifications with machine learning. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) *ECML 2002*. LNCS (LNAI), vol. 2430, pp. 219–231. Springer, Heidelberg (2002)
28. Lavrač, N., Džeroski, S.: *Inductive logic programming: techniques and applications*. Ellis Horwood, Chichester (1994)
29. Legendre, P.: Spatial autocorrelation: Trouble or new paradigm. *Ecology* 74, 1659–1673 (1993)
30. Malerba, D.: Learning recursive theories in the normal ILP setting. *Fundamenta Informaticae* 57(1), 39–77 (2003)
31. Malerba, D., Appice, A., Ceci, M.: A data mining query language for knowledge discovery in a geographical information system. In: Meo, R., Lanzi, P.L., Klemettinen, M. (eds.) *Database Support for Data Mining Applications*. LNCS (LNAI), vol. 2682, pp. 95–116. Springer, Heidelberg (2004)
32. Malerba, D., Ceci, M., Appice, A.: Mining model trees from spatial data. In: Jorge, A.M., Torgo, L., Brazdil, P.B., Camacho, R., Gama, J. (eds.) *PKDD 2005*. LNCS (LNAI), vol. 3721, pp. 169–180. Springer, Heidelberg (2005)
33. Malerba, D., Esposito, F., Lanza, A., Lisi, F.A., Appice, A.: Empowering a GIS with inductive learning capabilities: The case of INGENS. *Journal of Computers, Environment and Urban Systems* 27, 265–281 (2003)
34. McIver, D., Friedl, M.: Estimating pixel-scale land cover classification confidence using non-parametric machine learning methods. *IEEE Transactions on Geoscience and Remote Sensing* 39(9), 1959–1968 (2001)
35. Michalski, R.S.: A theory and methodology of inductive learning. *Artificial Intelligence* 20(2), 111–161 (1983)
36. Mitchell, T.: *Machine Learning*. McGraw Hill, New York (1997)
37. Muggleton, S.: *Inductive logic programming*. Academic Press, London (1992)
38. Mukerjee, A., Joe, G.: A qualitative model for space. In: *Proceedings of AAAI 1990*, pp. 721–727. AAAI Press, Menlo Park (1990)
39. Robinson, J.A.: A machine oriented logic based on the resolution principle. *Journal of the ACM* 12, 23–41 (1965)
40. Sander, J., Ester, M., Kriegel, H., Xu, X.: Density-based clustering in spatial databases: The algorithm gbscan and its applications. *Data Mining and Knowledge Discovery* 2(2), 169–194 (1998)
41. Seeger, M.: *Learning with labeled and unlabeled data*. Technical report, University of Edinburgh (2001)
42. Shekhar, S., Schrater, P.R., Vatsavai, R.R., Wu, W., Chawla, S.: Spatial contextual classification and prediction models for mining geospatial data. *IEEE Transactions on Multimedia* 4(2), 174–188 (2002)
43. Shekhar, S., Vatsavai, R., Chawla, S.: Spatial classification and prediction models for geospatial data mining. In: Miller, H.J., Han, J. (eds.) *Geographic Data Mining and Knowledge Discovery*, 2nd edn., pp. 117–147. Taylor & Francis, Abington (2009)
44. Shekhar, S., Zhang, P., Huang, Y.: Spatial data mining. In: Maimon, O., Rokach, L. (eds.) *The Data Mining and Knowledge Discovery Handbook*, pp. 833–851. Springer, Heidelberg (2005)

45. Taskar, B., Segal, E., Koller, D.: Probabilistic classification and clustering in relational data. In: Nebel, B. (ed.) Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI 2001), pp. 870–878 (2001)
46. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1995)
47. Vapnik, V.: Statistical Learning Theory. Wiley, New York (1998)
48. Wessel, M.: Some practical issues in building a hybrid deductive geographic information system with a dl component. In: Proceedings of the 10th International Workshop on Knowledge Representation meets Databases (KRDB 2003), Hamburg, Germany, September 15-16. CEUR Workshop Proceedings, vol. 79. CEUR-WS.org (2003)
49. Wettschereck, D.: A study of Distance-Based Machine Learning Algorithms. PhD thesis, Oregon State University (1994)