Spatial Regression in the Transductive Setting

Annalisa Appice, Michelangelo Ceci, Vincenzo Rizzi, Marco Romano, and Donato Malerba

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

Abstract. Spatial data are typically affected by positive autocorrelation. Formally, autocorrelation is the property of random variables taking values at pairs of locations a certain distance apart which are more similar than expected for randomly associated pairs of observations. This property is related to the smoothness assumption made in transduction, according to if two points in a high-density region are close, then the corresponding outputs should also be close. Transduction in spatial data mining also finds motivation in the prior availability of large sets of unlabelled data to be predicted which is typical of spatial data applications. We propose a spatial regression method, named SpReCo, that works in a transductive setting and employs a co-training technique to learn separate learners: one based on attribute values measured at some specific spatial positions, and the other based on attribute values derived by aggregating measurements in the neighborhood of those spatial positions. Each learner labels the unlabeled data for the other during the learning process. Predictive accuracy is evaluated on several spatial databases.

1 Introduction

Spatial Data Mining investigates how interesting and useful but implicit knowledge can be extracted from spatial data [8]. The main issue faced by spatial data mining methods is *positive spatial autocorrelation*. Formally, spatial autocorrelation is defined as the property of random variables taking values, at pairs of locations a certain distance apart, that are more similar than expected for randomly associated pairs of observations. Spatial autocorrelation is a clear violation of a basic assumption made by traditional data mining methods, namely the independence of observations in the training sample, and it is responsible of their poor performance [4]. Another issue, which is usually met in predictive spatial data mining tasks, is the *scarcity of labeled data*, since manual annotation of large data sets can be very costly. 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 predictors.

Two learning settings have been proposed in the literature: the semi-supervised setting [11] and the transductive setting [7]. The former is a type of inductive learning, since the learned function is used to make predictions on any possible

observation. The latter asks for less, since it is only interested in making predictions for a set of unlabeled data known at the learning time. Actually, in many spatial domains observations to be predicted are already known in advance.

Transduction is based on the *smoothness assumption* according to which if two points in a high-density region are close, then the corresponding outputs should also be close [3]. Interestingly, in spatial domains, where closeness of points corresponds to some spatial distance measure, this assumption is implied by positive spatial autocorrelation. These considerations motivate the investigation of learning a spatial model in a transductive setting in order to deal with both spatial autocorrelation and scarcity of labeled data.

This paper addresses the problem of regression in spatial data mining. To formulate a solution for this task we face two main issues. First, we design a learner which accounts for the spatial continuity over attribute values (autocorrelation) in predicting the response value and, at the same time, it disaggregates the regression surface according to the spatial structure of different areas of homogeneous dependence. Second, we have to devise a suitable technique for the prediction of spatial unlabelled data as accurate as possible. We provide answers to both issues. In particular, we propose a spatial data mining algorithm, named SpReCo (Spatial Regression with Co-Training), which resorts to the co-training paradigm [2] in order to learn regression models from different views of the same data. Regression models are model trees, each of which labels the unlabeled data for the other during the learning process. Co-training is made possible by separately learning two model trees: one based only on attribute values measured at some specific spatial positions, and the other based on attribute values derived by aggregating measurements of the explanatory attributes in the neighborhood of those spatial positions. The latter accounts for the possible spatial dependence and can be used to improve the prediction of the response value made by the former. In order to choose appropriate unlabeled observations to label, SpReCo estimates labeling confidence through consulting the influence of the labeling of each unlabeled observation in a k-NN based re-prediction of the labeled observations which are close the unlabeled one. The final prediction of unlabeled observations is computed as a weighted average of the regression estimates generated by both learners. In this paper, the model tree inducer presented in [1] is chosen to learn regressors at each iteration of SpReCo.

This paper is organized as follows. The next section introduces the transductive learning problem. Section 3 describes the algorithm SpReCo. Experimental results are reported in Section 4. Finally, Section 5 concludes.

2 Problem Statement

In this work, spatial information is modeled according to the *field* model [12], i.e., the space is seen as a continuous surface over which features vary, and the spatial variation is defined by a number of functions $f : \mathbb{R}^2 \mapsto \textit{Attribute domain}$. The set D of observations is a set of tuples $(id, u, v, \mathbf{x}, y)$, where id is a primary key which identifies the position $\langle u, v \rangle \in \mathbb{R}^2$, $\mathbf{x}$ is the vector of values measured for

the explanatory attributes $(X_1, \ldots X_d)$ at the position $\langle u, v \rangle$, and y is a (possibly unknown) response value with range in $\mathbb{R}$.

In the transductive setting, the spatial regression problem is formulated as follows. Given (i) the training set $T \subset D$, (ii) the projection of the working set $W = D - T$ on $ID \times U \times V \times \mathbf{X}$ and (iii) a spatial distance in $\mathbb{R}^2 \mapsto \mathbb{R}$ which defines the spatial neighborhood of a position $\langle u, v \rangle$; the goal is to find a prediction of the unknown response value of each observation in the working set W which is as accurate as possible.

The learner receives full information (including responses) on the observations in T and partial information (without responses) on the observations in W and is required to predict the response values of the observations in W. This formulation of the spatial regression problem is coherent with the original formulation of the problem of function estimation in a transductive setting is distribution-free and requires that both T and W are sampled from D without replacement. This means that, unlike the standard inductive setting, the observations in the training (and working) set are supposed to be mutually dependent.

3 Algorithm

The top-level description of SpReCo is reported in Algorithm 1. Let D, T, and W be the original set, the training set and the working set, respectively ($D = T \cup W$). SpReCo first derives an alternative description $\overline{D} = \overline{T} \cup \overline{W}$ of data in D and then uses training sets in D and $\overline{D}$ to iteratively induce distinct regression models. Each regression model is then refined with the help of the unlabeled observations which are labeled by the latest version of the other regression model. Details of the main function in SpReCo are provides below.

Let $e = (id_e, u_e, v_e, \mathbf{x_e}, y_e)$ be an observation of D, $\overline{e} = (id_e, u_e, v_e, \overline{\mathbf{x_e}}, y_e$ is constructed in $\overline{D}$ by opportunely aggregating the observations falling in the neighborhood of $\Omega_D(e)$ (see function *neighborhoodBasedDescription*()). $\Omega_D(e)$ denotes the set of the h observations $p \in D$ which are taken in D and are located at the sites in the neighborhood of e. h is the size of $\Omega_D(e)$ and it is a user-defined parameter of SpReCo. The Euclidean distance is computed to determine nearest neighbors. $\overline{x_{i_e}}(i = 1, \ldots, d)$ is computed as follows:

$$\overline{x_{i_e}} = \frac{\sum\limits_{p \in \Omega_D(e)} (x_{i_p} \times w(e,p))}{\sum\limits_{p \in \Omega_D(e)} w(e,p)} \text{ with } i = 1, \ldots, d. \tag{1}$$

where $w(e, p)$ is a weight defined on the basis of the Euclidean proximity of p to e. $w(e, p)$ is computed as follow:

$$w(e,p) = e^{-\frac{distance(e,p)^2}{b_e^2}}, \tag{2}$$

where b_e is the bandwidth that depends on the site $\langle u_e, v_e \rangle$, that is, $b_e = \max_{p \in \Omega_D(e)} distance(e, p)$.

Algorithm 1 Spatial regression with co-training.

SpReCo(T, W)
$\langle \overline{T}, \overline{W} \rangle \leftarrow$ neighborhoodBasedDescription$(T \cup W)$;
$T_0 \leftarrow T$; $W_0 \leftarrow W$; $T_1 \leftarrow \overline{T}$; $W_1 \leftarrow \overline{W}$; $Y_W = \oslash$; $i \leftarrow 1$;
repeat
  change $\leftarrow$ false; $t_0 \leftarrow$learner(T_0); $t_1 \leftarrow$learner(T_1);
  for $j \in \{0, 1\}$ **do**
    $L_0 \leftarrow \oslash$; $L_1 \leftarrow \oslash$; $\epsilon_{Pos} \leftarrow 0$; $\epsilon_{Neg} \leftarrow 0$;
    for $e \in W_j$ **do**
      $e_{\widehat{y}} \leftarrow$ response(t_j, e);
      $\Omega_{T_j}(e) \leftarrow$neighborhood(e,$T_j$);
      for $p \in \Omega_{T_j}(e)$ **do**
        $\epsilon_p \leftarrow (p_y - knn(p, T_j))^2 - (p_y - knn(p, T_j \cup \langle e, \widehat{y_e} \rangle))^2$;
        if $(\epsilon_p \geq 0)$ $\epsilon_{Pos} \leftarrow \epsilon_{Pos} + 1$;
        else $\epsilon_{Neg} \leftarrow \epsilon_{Neg} + 1$;
      end for
      if $(\epsilon_{Pos} \geq \epsilon_{Neg})$
        $L_{1-j} \leftarrow L_{1-j} \cup \langle$ instance$(e, W_{1-j}), e_{\widehat{y}} \rangle$;
        change $\leftarrow$ true;
      end if
    end for
  end for
  $T_0 \leftarrow T_0 \cup L_0$; $W_0 \leftarrow W_0 - L_0$; $T_1 \leftarrow T_1 \cup L_1$; $W_1 \leftarrow W_1 - L_1$;
until $(++i \geq$ MAX_ITERS AND *change*$)$;
$m_0 \leftarrow$ mse(t_0, T); $m_1 \leftarrow$ mse$(t_1, \overline{T})$;
if $(m_0 > m_1)$ $\omega_0 \leftarrow 1$; $\omega_1 \leftarrow m0/m1$;
else $\omega_0 \leftarrow m1/m0$; $\omega_1 \leftarrow 1$;
for $e \in W$ **do**
  $\overline{e} \leftarrow$ instance$(e, \overline{W})$; $Y_W \leftarrow Y_W \cup \langle e, \frac{\text{response}(t_0,e)\omega_0 + \text{response}(t_1,\overline{e})\omega_1}{\omega_0+\omega_1} \rangle$;
end for
return Y_W

At each iteration, SpReCo induces two regression models, i.e., t_0 and t_1, from the set of labeled data T_0 and T_1, respectively. At the first iteration, T_0 and T_1 correspond the training data T and $\overline{T}$, respectively. In this work, the model tree learner in [1] is used as the base learner to instantiate t_0 and t_1. The use of model trees is due to their capability of do not imposing any a-priory defined global form of regression surface, but assuming a functional form at local level. Model trees are here induced in a stepwise fashion. At each step of tree construction, the choice is to either partition the current training set (split node) or to introduce a regression attribute in the linear models. Each regression model t_j $(j = 0, 1)$ is then used to predict the response values $e_{\widehat{y}}$ of the still unlabeled observations e which currently belong to the working set W_j(see function *predict*()). Confidence of predicted labels is estimated and the most confident labels are identified.

The confidence of labeling is estimated by a k-NN learner. Heuristically, the most confidently labeled observation e should be the one which makes a k-NN learner most consistent with the labeled set. According to this property, let $e \in T_j$

be a working observation and $e_{\widehat{y}}$ the response currently predicted for e whose confidence has to be evaluated, the k-NN is used to re-predict each observation p falling in $\Omega_{T_j}(e)$, where $\Omega_{T_j}(e)$ denotes the neighborhood of e determined in the training set T_j. The squared error of the k-NN without the information provided by $(e, e_{\widehat{y}})$ is compared with the squared error of the k-NN with the information provided by $(e, e_{\widehat{y}})$. Let ϵ_p be the result of subtracting the former squared error from the latter squared, that is:

$$\epsilon_p = (p_y - \text{knn}(p, T_j))^2 - (p_y - \text{knn}(p, T_j \cup \langle e, e_{\widehat{y}} \rangle))^2 \tag{3}$$

where p_y is the response that originally labels p in T_j at the current iteration of SpReCo. The function $\text{knn}(p, T_j)$ returns the k-NN learner with training set T_j, while $\text{knn}(p, T_j \cup \langle e, e_{\widehat{y}} \rangle)$ returns the k-NN learner with training set $T_j \cup \langle e, e_{\widehat{y}} \rangle$. As k-NN learner, we adopt a version of weighted k-Nearest Neighbor algorithm [9], where responses in a k sized neighborhood of p are weighted according to same weighting function defined in Equation 2. By denoting as:

$$\epsilon_{Pos} = |\{p \in \Omega_{T_j}(e) | \epsilon_p \geq 0\}| \text{ and } \epsilon_{Neg} = |\{p \in \Omega_{T_j}(e) | \epsilon_p < 0\}| \tag{4}$$

with $|\bullet|$ the cardinality of a set $\bullet$, the labeling $e_{\widehat{y}}$ is estimated as confident if $\epsilon_{Pos} \geq \epsilon_{Neg}$, un-confident otherwise.

Predicted labels which are identified as confident by t_j are used to label the observations in the working set W_{1-j} of the other learner t_{1-j}. Let $e \in W_j$ be a working observation and $e_{\widehat{y}}$ the response value predicted by t_j for e. If prediction $e_{\widehat{y}}$ is evaluated as confident, then SpReCo retrieves $instance(e, W_{1-j})$ that is the observation of W_{1-j} identified by e_{id}. $instance(e, W_{1-j})$ is removed from W_{1-j}, labeled with $e_{\widehat{y}}$ and added to T_{1-j} for the next iteration of SpReCo.

The learning process stops when the maximum number of learning iterations, MAX_ITERS, is reached, or there is no unlabeled observation which can be confidently moved from the working set to the training set. Finally, predictions of learners t_j constructed in the last iterations of SpReCo are averaged as a final prediction of the working observations. The mean square error (see function $mse()$) of both learners is computed on the original training set (T and $\overline{T}$, respectively) and it is used to weight predictions of two learners.

4 Experiments

SpReCo has been validated on four spatial databases. *USA Geographical Analysis Spatial Data (GASD)* [10] contains 3,107 observations on USA county votes cast in 1980 presidential election. For each county it contains the total number of votes cast in the 1980 presidential election (response attribute), the population of 18 years of age or older, the population with a 12th grade or higher education, the number of owner-occupied housing units, the aggregate income, the X and Y spatial coordinates. *Forest Fires* [5] is public available for research at UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/). It collects 512 forest fire observations from January 2000 to December 2003 in Montesinho natural park in

the northeast region of Portugal. The data include the burned area of the forest in ha (1ha/100 = 100 m^2) (response attribute), the Fine Fuel Moisture Code, the Duff Moisture Code, the Drought Code, the Initial Spread Index, the temperature in Celsius degrees, the relative humidity, the wind speed in km/h, the outside rain in mm/m^2, the X and Y spatial coordinates within the Montesinho park map. *North-West England (NWE)* contain census data provided in European project SPIN!(http://www.ais. fraunhofer.de/KD/SPIN/project.html). Census data are provided by 1998 Census for the 1011 wards of North West England area. For each ward, data include percentage of mortality (response attribute) and measures of deprivation such as, Jarman Underprivileged Area Score, Townsend score, Carstairs score and the Department of the Environments Index, the X and Y spatial coordinates of the ward centroid. By removing observations with null values, only 979 observations are used in this experiments. *Sigmea-Real* [6] collects 817 measurements of the rate of herbicide resistance of two lines of plants (response attributes), that is, the transgenic male-fertile (MF) and the non-transgenic male-sterile (MS) line of oilseed rape. Explanatory attributes are the cardinal direction and distance from the center of the donor field, the visual angle between the sampling plot and the donor field, and the shortest distance between the plot and the nearest edge of the donor field, the X and Y spatial coordinates of the plant.

The experiments aim at validating the actual advantage of the transductive algorithm over the basic inductive algorithm when few labeled observations are available. The basic inductive algorithm is the model tree learner induced from the original training data collection (T). We also evaluate the advantages of a co-training implementation in transductive learning. The empirical comparison is based on the mean square error (MSE) of the two algorithms. To estimate the MSE, a K-fold cross validation is performed. K is set to 10 in experiments performed with GASD dataset, and $K = 5$ in experiments performed with Forest Fires, NWE and Sigmea Real. 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 MSE error over the K-folds (Avg.MSE). Unlike the standard cross-validation approach, only one fold is used for the training set. In this way we simulate datasets with a small percentage of labeled cases (the training set) and a large percentage of unlabeled data (the working set).

Since the performance of the algorithm may depend on (i) the size h of the neighborhood used to determine the alternate data view employed in the co-training and (ii) the size k of the neighborhood used in k-NN evaluation of confidence of labeling performed by each learner, experiments for different h and k are performed in order to set the optimal value. h ranges among 1 (i.e. the case transductive learning is performed without co-training), 5, 10 ,15 and 20. k ranges among 5, 10, 15 and 20. The Avg.MSE performed by SpReCo ($MAX\_ITERS = 5$) is compared with the Avg.MSE performed with the baseline model tree learner. The comparison is performed in terms of percentage of error loss that is reported in Table 1. The error loss refers to decrease (or

Table 1. Percentage of error loss: inductive learning vs transductive learning

h	k	GASD	Forest Fires	NWE	Sigmea Areal - MS	Sigmea Areal - MF
1	5	8.92%	-1.32%	1.14%	0.09%	0.01%
1	10	10.91%	-2.43%	2.27%	0.09%	-0.57%
1	15	10.97%	-3.06%	-0.23%	0.05%	-0.56%
1	20	10.82%	-1.89%	1.99%	0.09%	-0.39%
5	5	14.68%	19.80%	3.89%	4.75%	-1.02%
5	10	14.70%	20.85%	3.46%	4.73%	-3.96%
5	15	14.55%	20.13%	3.73%	3.57%	-2.40%
5	20	14.39%	20.05%	3.89%	-1.44%	-1.93%
10	5	13.67%	21.52%	2.59%	-0.32%	-2.97%
10	10	13.95%	20.94%	2.62%	0.24%	-4.13%
10	15	13.91%	20.28%	2.60%	-0.88%	-5.62%
10	20	12.90%	20.85%	2.81%	1.27%	-3.88%
15	5	13.53%	20.64%	2.81%	0.38%	-2.44%
15	10	11.60%	20.58%	2.98%	-0.83%	-3.91%
15	15	13.52%	20.78%	2.80%	2.01%	-1.07%
15	20	12.00%	21.03%	2.88%	1.23%	-2.74%
20	5	10.83%	21.29%	2.59%	3.70%	-3.71%
20	10	11.63%	20.54%	2.81%	1.69%	-4.98%
20	15	12.45%	20.27%	2.62%	1.69%	-4.79%
20	20	11.37%	20.91%	2.52%	5.62%	-3.39%

increase) of the Avg.MSE of the transductive algorithm with respect to the inductive algorithm. A positive (negative) value is in favor of the transductive (inductive) algorithm.

Results confirm that SpReCo performs generally better than the basic model tree learner in almost all the tested transductive settings. The exception is represented by Sigmea Real (MF) that is the only dataset where the baseline inductive learner outperforms the trasductive one. Our justification is that the worse performance of the learner may depend on the fact that this dataset exhibits about 65% of observations which are labeled as zero which leads to a degradation of both predictive capability of the learner that operates with the aggregate data view in the co-training and capability of identifying confident labels.

Although, the improvements in accuracy provided by SpReCo depends on h and k values, obtained results do not provide any suggestion on optimum choice of h and k. In general, best accuracy is obtained with $h > 1$, that is, in the case co-training does not degenerate in self training. This confirms that co-training improves accuracy of transductive learner in the case of spatial regression task.

5 Conclusions

In this work we propose a spatial regression method, named SpReCo, which works in transductive learning and is designed in a a co-training style. Two regression models are induced from two views of the same set of labeled data.

Each regression model is used to predict the response of the unlabeled data (working set) for the other learner during the learning process. Confidence of the labeling is estimated through consulting the influence of the labeling of unlabeled observations on a K-NN based re-prediction of the labeled ones. The final prediction of unlabeled observations is made by a weighted average of the regression estimates predicted by the last learners obtained by co-training. The transductive learner has been compared to its inductive counterpart on several spatial datasets. Experimental results are generally in favor of the transductive algorithm with co-training.

Acknowledgments

This work is supported by PRIN COFIN Project 2007 "Multi-relational approach to spatial data mining" and the Strategic Project PS121: "Telecommunication Facilities and Wireless Sensor Networks in Emergency Management". The authors thank Saso Dzeroski for kindly providing Sigmea Areal dataset.

References

1. A. Appice and S. Dzeroski. Stepwise induction of multi-target model trees. In J. N. Kok, J. Koronacki, R. L. de Mántaras, S. Matwin, D. Mladenic, and A. Skowron, editors, *18th European Conference on Machine Learning, ECML 2007*, volume 4701 of *Lecture Notes in Computer Science*, pages 502–509. Springer-Verlag, 2007.
2. A. Blum and T. M. Mitchell. Combining labeled and unlabeled data with co-training. In *COLT*, pages 92–100, 1998.
3. O. Chapelle, B. Scholkopf, and A. Zien. *Semi-supervised learning*. MIT Press, Cambridge:MA, 2006.
4. S. Chawla, S. Shekhar, and W. L. Wu. Modeling spatial dependencies for mining geospatial data: An introduction. In H. J. Miller and c. . J. Han, editors, *Geographic data mining and Knowledge Discovery GKD*. Taylor and Francis, 2001.
5. P. Cortez and A. Morais. A data mining approach to predict forest fires using meteorological data. pages 512–523. APPIA, 2007.
6. D. Demšar, M. Debeljak, C. Lavigne, and S. Džeroski. Modelling pollen dispersal of genetically modified oilseed rape within the field. In *Abstracts of the 90th ESA Annual Meeting, The Ecological Society of America*, page 152, 2005.
7. A. Gammerman, K. S. Azoury, and V. Vapnik. Learning by transduction. In G. F. Cooper and S. Moral, editors, *14th Conference on Uncertainty in Artificial Intelligence, UAI 1998*, pages 148–155. Morgan Kaufmann, 1998.
8. K. Koperski. *Progressive Refinement Approach to Spatial Data Mining*. PhD thesis, Computing Science, Simon Fraser University, British Columbia, Canada, 1999.
9. T. Mitchell. *Machine Learning*. McGraw Hill, 1997.
10. P. Pace and R. Barry. Quick computation of regression with a spatially autoregressive dependent variable. *Geographical Analysis*, 29(3):232–247, 1997.
11. B. Shahshahani and D. Landgrebe. The effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon. *IEEE Transactions on Geoscience and Remote Sensing*, 32:1087–1095, 1994.
12. S. Shekhar and S. Chawla. *Spatial databases: A tour*. Prentice Hall, Upper Saddle River:NJ, 2003.