SEMI-SUPERVISED KERNEL METHODS FOR REGRESSION ESTIMATION

Alexei Pozdnoukhov and Samy Bengio

IDIAP Research Institute, CP 592, CH-1920 Martigny
Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland

ABSTRACT

The paper presents a semi-supervised kernel method for regression estimation in the presence of unlabeled patterns. The method exploits a recently proposed data-dependent kernel which is constructed in order to represent the inner geometry of the data. This kernel is implemented into Kernel Regression methods (SVR, KRR). Experimental results aim to highlight the properties of the method and its advantages as compared to fully supervised approaches. The influence of the parameters on the model properties was evaluated experimentally. One artificial and two real-world datasets were used to demonstrate the performance of the proposed algorithm.

1. INTRODUCTION

The problem of using unlabeled data is of increasing attention in Machine Learning. By unlabeled data, we mean those data samples which consist of the input values only, while the desired output value is unknown. In signal processing this is the situation when the signal was registered but not classified (processed) due to malfunction, time restrictions or by any other reasons. Furthermore, methods making use jointly of labeled and unlabeled data are called *semi-supervised*. In fact, most real-life learning problems are actually semi-supervised.

The information one obtains from the unlabeled part of the dataset can be of different nature. A common approach is to consider the *manifold assumption.* This implies that data actually belong to some lower dimensional manifold in high dimensional input space. A large body of literature is devoted to the exploration of such an approach; see [1] and references therein.

Given the explosive growth of interest in the field of kernel methods, non-parametric data-dependent kernels which reflect the inner geometry of the data are of particular interest. A general approach was recently proposed in [4]. Here, we implement the proposed kernel for kernel regression estimators, discuss the obtained method and explore its properties in a number of experiments on artificial and real data. We consider multidimensional regression tasks and, in particular, time series prediction with missing values.

This work has been supported by the IST Programme of the EC, PASCAL, IST-2002-506778, funded in part by the Swiss OFES. It was also partially funded by the Swiss NCCR project (IM)2.

2. SEMI-SUPERVISED LEARNING

Machine Learning approaches can be divided into supervised and unsupervised learning algorithms. The supervised learner aims at estimating the input-output relationship (dependency or function) $f(\mathbf{x})$ by using a training data set $\{\mathbf{x}_i, y_i\}$, $i = 1, \ldots, N$ where the inputs $\mathbf{x}$ are n-dimensional vectors and the labels (or system responses) y are continuous values for regression tasks and discrete (e.g., boolean) for classification problems; in unsupervised learning, however, only raw data $\mathbf{x}_i$ are available, without the corresponding labels y_i. The algorithms belonging to this group are various clustering and (principal or independent) component analysis routines.

Furthermore, the situation where some labeled patterns are provided together with unlabeled ones, arises frequently. This is called *semi-supervised learning.* When predictions have to be made to given unlabeled locations only, this particular situation is called transductive learning. Recently several approaches to semi-supervised learning were proposed. The LDS algorithm [2], Transductive SVM, Graph and Gradient Transductive SVM [3], and a group of Manifold Learning methods [1] are the core of those recently developed techniques.

Most of the work done in this field is related to fully unsupervised tasks or semi-supervised classification problems. Semi-supervised regression methods are, however, much less studied. In this paper, we combine recent developments in the field of manifold learning with kernel regression learners such as Support Vector Regression and Kernel Ridge Regression.

3. DATA DEPENDENT KERNELS

A semi-positive definite function which satisfies Mercer conditions is called a kernel. This implies that it corresponds to a dot product in some space (Reproducing Kernel Hilbert Space, RKHS), sometimes referred to as a feature space. Generally, given a (linear) algorithm, which includes data samples in the form of dot products only, one can obtain a (non-linear) kernel version of it by substituting the dot products with kernel functions [5]. The choice of the kernel function is an open issue. Using some typical kernels like Gaussian RBF, one takes into account some knowledge (distance-based similarity of the samples). However, neither the inner geometry of the data nor local structures are reflected with this choice. Hereafter, we briefly present a method of [4] for constructing non-parametric semi-supervised kernels which eliminate these drawbacks. We will follow the notation of [4]. Given data points $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, and some RKHS H, consider the evaluation map $S(\mathbf{f}) = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)); S : H \to \mathbb{R}^n$. The semi-norm on $\mathbb{R}^n$ is given by a symmetric semi-definite matrix M,

$$\|S(\mathbf{f})\|^2 = \mathbf{f}^T\mathbf{M}\mathbf{f}, \tag{1}$$

where we denoted $\mathbf{f} = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n))$ and T means transpose. The exact explicit form of the corresponding reproducing kernel $\tilde{k}(\mathbf{x}, \mathbf{x}')$ was derived in [4] and is given by:

$$\tilde{k}(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}') - k_{\mathbf{x}}^T(\mathbf{I} + \mathbf{MK})^{-1}\mathbf{M}k_{\mathbf{x}'} \tag{2}$$

where $\mathbf{K}$ is the complete kernel matrix of $k(\cdot, \cdot)$, $k_{\mathbf{x}}$ represents one row of $\mathbf{K}$ and $\mathbf{I}$ is the identity matrix. In the presence of unlabeled data, the choice of $\mathbf{M}$ implements the smoothness assumption with respect to its geometric structure. As shown in [1], this is achieved by taking $\mathbf{M} = \gamma\mathcal{L}$, $\mathcal{L}$ being the Laplacian matrix of the graph built on unlabeled samples, and γ a regularization parameter which defines the extent of kernel deformation. By setting γ=0 one obtains the original kernel, as it is clearly seen with (2).

4. KERNEL REGRESSION METHODS

We state the general problem of regression estimation as it is presented in the scope of Statistical Leaning Theory [6]. Suppose we are given a set of observations $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$ generated from an unknown probability distribution $P(\mathbf{X}, \mathbf{Y})$ with $\mathbf{x}_i \in \mathbb{R}^{\mathrm{n}}, y_{\mathrm{i}} \in \mathbb{R}$ and a class of functions $F = \{f|\mathbb{R}^{\mathrm{n}} \to \mathbb{R}\}$. Our task is to find a function f from the given class of functions that minimizes a risk functional:

$$R[f] = \int Q(y - f(\mathbf{x}), \mathbf{x})dP(\mathbf{x}, y) \ , \tag{3}$$

where Q is a loss function indicating how the difference between measurement value and model's prediction is penalized. As $P(\mathbf{x}, y)$ is unknown, the empirical risk is used instead:

$$R_{emp} = \frac{1}{N}\sum_{i=1}^{N} Q(y - f(\mathbf{x}), \mathbf{x}). \tag{4}$$

4.1. Support Vector Regression

The Support Vector Regression model is based on the linear ε-insensitive loss functions. Following the Structural Risk Minimization principle of Vapnik, the model complexity has to be penalized simultaneously with keeping empirical risk (training error) small. The complexity of linear functions $F = \{f(\mathbf{x})|f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b\}$ can be controlled by the term $\|\mathbf{w}\|^2$ [6]. Introducing the trade-off constant C, this results in the following optimization problem:

$$\text{minimize} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \tag{5}$$

$$\text{subject to} \begin{cases} f(\mathbf{x}_i) - y_i - \varepsilon \le \xi_i \\ -f(\mathbf{x}_i) + y_i - \varepsilon \le \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \text{ for } i = 1, \ldots, N \ . \end{cases}$$

Introducing Lagrange multipliers leads to the following dual formulation of the problem, where dot products were substituted with a kernel function:

$$\begin{aligned} \text{maximize} \quad & -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)k(\mathbf{x}_i, \mathbf{x}_j) \\ & -\varepsilon\sum_{i=1}^{N}(\alpha_i^* + \alpha_i) + \sum_{i=1}^{N} y_i(\alpha_i^* - \alpha_i) \end{aligned} \tag{6}$$

$$\text{subject to} \begin{cases} \displaystyle\sum_{i=1}^{N}(\alpha_i^* - \alpha_i) = 0 \\ 0 \le \alpha_i^*, \alpha_i \le C \text{ for } i = 1, \ldots, N \ . \end{cases}$$

This problem is a Quadratic Programming problem hence can be numerically solved by a number of methods. The prediction is a non-linear regression function:

$$f(\mathbf{x}) = \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)k(\mathbf{x}_i, \mathbf{x}) + b \tag{7}$$

where b can be found easily given the constraints in 6. Finally, let us summarize the expected properties of the constructed algorithm and highlight the issues of its practical use. Generally, the method is non-linear and robust. The parameters of SVR are:

C - the parameter that defines the trade-off between training error and model complexity. In dual formulation C defines the upper bound of the multipliers α_i and α_i^* (6), hence defines the maximal influence the sample can have on the solution. This means that the more noisy the data the less should be the value of C.

ε - the width of the insensitive region of the loss function. This is the parameter that defines the sparseness of the SVR solution - the points that lie inside the ε-tube have zero weights.

4.2. Kernel Ridge Regression

Kernel Ridge Regression is a regularized least square approach, which leads to the same form of regression function (7). However, it exploits the square loss function, and α coefficients can be obtained from the following closed form expression:

$$\alpha = (\mathbf{K}^T\mathbf{K} + \delta\mathbf{I})^{-1}\mathbf{K}^T\mathbf{Y} \tag{8}$$

where δ is a regularization parameter and $\mathbf{Y}$ is the vector of training outputs. Note that an iterative method can also be used to train the KRR model.

5. EXPERIMENTS

The experiments described below were carried out on the following datasets: spiral, Boston housing and sunspots. The first one is an artificial dataset, which we use to explore and illustrate the basic properties of the method. The other two are real-world datasets, commonly used in machine learning for benchmarking different algorithms.

5.1. Kernel Choice

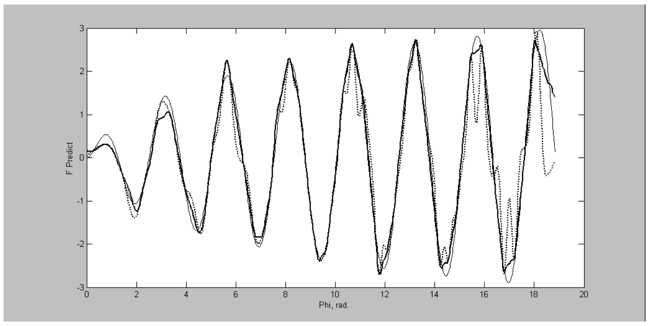
In this paper, we used the kernel described in Section 3, using the standard Gaussian RBF kernel with bandwidth σ as a base kernel. Gaussian RBF is used in all the baseline supervised algorithms as well. Another parameter to select is the regularization parameter γ of the modified kernel (2). This will be explored empirically in this section.

5.2. Spiral: 2D Synthetic Example

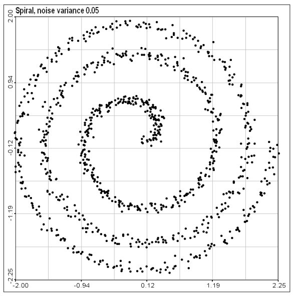
This dataset was artificially generated with:

$$\begin{cases} x^1(\phi) = \frac{1}{2}\sqrt{\phi}\cos(\phi) + N(0,\sigma_x) \\ x^2(\phi) = \frac{1}{2}\sqrt{\phi}\sin(\phi) + N(0,\sigma_x) \\ f(\phi) = \ln(1+\phi)\sin(\frac{5}{2}\phi) + N(0,\sigma_f) \end{cases} \tag{9}$$

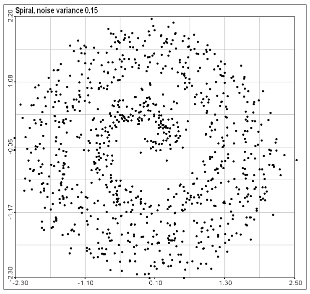
in the range of $\phi \in [0; 6\pi]$. The function $f(\phi)$ to predict is defined on the 2D spiral. This function is presented in Figure 1a with a thin solid line. Both coordinates and function values are corrupted with normal noise of variance σ_x^2 and σ_f^2 correspondingly. Two random data realizations are presented in Figures 1b and 1c. We compare the performance of the proposed method and the standard Support Vector Regression with Gaussian RBF kernel. Labeled part of the training set consist of 100 randomly selected samples, while an other set of 900 samples were provided unlabeled to the semi-supervised method. The results are averaged over 10 runs of the algorithm (each run selecting different training and test examples), and its performance was measured in terms of RMSE using the known underlying function $f(\phi)$ (9).



(a) Function $f(\phi)$ and its estimates by semi-supervised (bold line) and standard (dotted line) SVR.



(b) Inputs, σ_x=0.05



(c) Inputs, σ_x=0.15

Fig. 1. 2D spiral data.

Figure 2a presents the dependence of the testing error of both methods with respect to the variance of noise in the inputs σ_x. The top curve (with higher RMSE) corresponds to standard SVR. As can be seen, semi-supervised regression (bottom curve) is preferable for a large region of noise variance, provided that some geometrical structure in the data remains.

Figure 2b presents the dependence of the testing error of both methods with respect to the kernel regularization parameter γ. The dashed line corresponds to the testing error of the basic SVR. The semi-supervised method outperforms SVR for a large range of values.

5.3. Boston housing: High Dimensional Regression Estimation

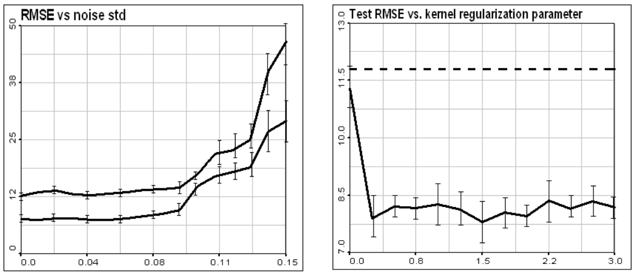
The task here is to predict the median price of the houses in certain area of Boston based on 12 continuous and 1 binary variables defining the characteristics of the area. The training dataset consists of 466 samples, while 40 samples were reserved for testing. The parameters of the methods were tuned with cross-validation error. Unlabeled data were randomly chosen by removing the labels from 50% of data samples. The results were averaged over 10 runs of the algorithm (each run with different training and test examples).

Table 1 presents training, testing error and training time of the following algorithms: the considered SVR with semi-supervised kernel (SemiSVR), SVR with Gaussian RBF kernel, the standard method of Kernel Ridge

Table 1. Experimental results for Boston Housing database.

	Boston housing results		
Algorithm	Train err.	Test err.	Training time, s
SVR	4.0	5.3	0.3
SemiSVR	3.5	5.0	0.5
KRR	2.7	4.0	0.3
SemiKRR	3.5	4.0	0.5

Regression (KRR), and KRR with a semi-supervised kernel (SemiKRR). The results suggest that no significant improvement was achieved on this dataset. There was probably not enough data samples to model the manifold in the 13-dimensional input space.



(a) RMSE, X-axis: σ_x. (b) RMSE, X-axis: γ.

Fig. 2. Spiral data experimental results.

5.4. Sunspots: Time Series Prediction with Missing Values

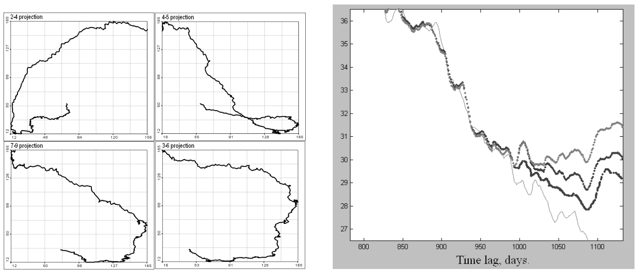
This dataset is a time series representing the number of visible sunspots per day. The following embedding was used to apply a regression estimator for predictions: to predict the yearly average of the year starting next day using the previous 12 yearly averages. The series is thus smoothed by averaging. Figure 3a presents some 2D projections (trajectories) in the embedded input space. One can observe a distinct structure of the inputs, which justifies the use of manifold-based semi-supervised methods for making predictions.

We used only one part of the series containing 2000 values. 50% of the labels were deleted from the series to simulate missing data. Hence, the unlabeled part of the dataset consisted of 1000 samples, and the training set also contained 1000 labeled samples. Missing values in inputs were averaged using two nearest neighbors in time. The obtained results are summarized in Table 2. The results are averaged over 10 runs of the algorithm where different sections of 2000 points were selected randomly.

Predictions are presented in Figure 3b, for γ=0 (standard SVR), γ=0.1, γ=1. The semi-supervised SVR gives better forecasting for longer time periods, for higher values of γ.

Table 2. Experimental results for the Sunspots database.

	Sunspots results		
Algorithm	Train err.	Test err.	Training time, s
SVR	10.3	15.8	10.1
SemiSVR	9.4	12.3	30.4
KRR	11.3	17.5	12.6
SemiKRR	12.1	14.0	35.3



(a) Some 2D trajectories (b) SemiSVR Predictions

Fig. 3. Sunspots database results

6. CONCLUSIONS

In this paper we proposed to implement the recently developed data-dependent semi-supervised kernel for regression estimation methods, namely Support Vector Regression and Kernel Ridge Regression. Thus, the methods are adapted for semi-supervised learning problems. Some issues of the practical use of the methods were considered. We have shown that the semi-supervised methods do benefit in the case where there exists some geometrical structure in data. A significant improvements in performance compared to baseline supervised kernel regression estimators was shown in a number of experiments on synthetic and real-life datasets.

7. REFERENCES

[1] Belkin, M. Problems of Learning on Manifolds. Ph.D. dissertation, University of Chicago, 2003.

[2] Chapelle, O., Zien, A. Semi-supervised Classification by Low Density Separation. In Proc. of AI&Statistics, 2005.

[3] Joachims, T. Transductive Learning via Specral Graph Partitioning In Proc. of ICML, 2003.

[4] Sindhwani, V., Niyogi, P., Belkin, M. Beyond the Point Cloud: from Transductive to Semi-supervised Learning In Proc. of ICML'05, Bonn, Germany.

[5] Scholkopf, B., Smola, A.J. Learning with Kernels. MIT press, Cambridge, MA, 2002.

[6] V. Vapnik. Statistical Learning Theory. J.Wiley, NY, 1998.