Neurocomputing 163 (2015) 106–114

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Improving the kernel regularized least squares method for small-sample regression

Igor Braga, Maria Carolina Monard

Institute of Mathematics and Computer Science, University of São Paulo, Av. Trabalhador São-carlense, 400, São Carlos–SP 13566-590, Brazil

ARTICLE INFO

Article history:
Received 11 February 2014
Received in revised form
30 November 2014
Accepted 10 December 2014
Available online 4 April 2015

Keywords:
Non-linear regression
kernel regularized least squares
Cross-validation
RBF kernel
Spline kernel
Parameter selection

ABSTRACT

The kernel regularized least squares (KRLS) method uses the kernel trick to perform non-linear regression estimation. Its performance depends on proper selection of both a kernel function and a regularization parameter. In practice, cross-validation along with the Gaussian RBF kernel have been widely used for carrying out model selection for KRLS. However, when training data is scarce, this combination often leads to poor regression estimation. In order to mitigate this issue, we follow two lines of investigation in this paper. First, we explore a new type of kernel function that is less susceptible to overfitting than the RBF kernel. Then, we consider alternative parameter selection methods that have been shown to perform well for other regression methods. Experiments conducted on real-world datasets show that an additive spline kernel greatly outperforms both the RBF and a previously proposed multiplicative spline kernel. We also find that the parameter selection procedure Finite Prediction Error (FPE) is a competitive alternative to cross-validation when using the additive splines kernel.

1. Introduction

Non-linear regression estimation is an important scientific modeling tool. Several methods have been proposed to tackle this estimation problem, with the most flexible and powerful ones falling in the category of the so-called kernel methods [1]. Among those is the kernel regularized least squares (KRLS) method [2–4], which enjoys good statistical and computational properties.

In a nutshell, the kernel regularized least squares method works as follows. Using a sequence of training data

$$(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n), \quad \boldsymbol{x} \in R^d, y \in R, \tag{1}$$

drawn *i.i.d.* from a fixed but unknown probability distribution $P(\boldsymbol{x}, y)$, a function $f_{K,\gamma}(\boldsymbol{x})$ is obtained as the solution of the minimization problem

$$f_{K,\gamma} = \arg\min_{f \in \mathcal{H}_K} \left[\frac{1}{n} \sum_{i=1}^{n} \left(y_i - f(\boldsymbol{x}_i)\right)^2 + \gamma \|f\|_K^2 \right], \tag{2}$$

where $\gamma > 0$ is a real-valued regularization parameter and $\mathcal{H}_K$ is a Reproducing Kernel Hilbert Space (RKHS) induced by a kernel K. A function $f \in \mathcal{H}_K$ with bounded $\|f\|_K$ satisfies some regularity properties (*e.g.*, smoothness), hence the use of the term "regularized" to name the method.

In order to apply KRLS successfully, that is, to use the obtained $f_{K,\gamma}(\boldsymbol{x})$ to predict the output y of unseen $\boldsymbol{x}$, we must find such $f_{K,\gamma}$ that (1) fits the training sequence well (*i.e.*, minimizes the squared loss) and (2) is a reasonably smooth function (*i.e.*, minimizes the norm $\|\cdot\|_K$). As Statistical Learning Theory dictates [5], one can always minimize the former at the expense of the latter, and vice versa. Therefore, proper selection of both the kernel K and the regularization parameter γ is indispensable for the generalization performance of KRLS.

Formally, the best choice of K and γ is the one that yields in Expression (2) a function $f_{K,\gamma}$ that minimizes the risk of prediction error as measured by the expected squared loss

$$\mathcal{R}(f) = \int (y - f(\boldsymbol{x}))^2 \, dP(\boldsymbol{x}, y). \tag{3}$$

The minimum of the functional $\mathcal{R}(f)$ is attained at the regression function [5, Chapter 1]. Thus, the closer $\mathcal{R}(f_{K,\gamma})$ is to the minimum of $\mathcal{R}(f)$, the closer the outputs of $f_{K,\gamma}$ are to those of the real regression function.

The choice of suitable K and γ belongs to the category of problems known as *model selection*. In contrast to the related category of *model assessment*, model selection does not require the estimation of the value of the prediction error $\mathcal{R}(f)$. It suffices to indicate the function with the smallest $\mathcal{R}(f)$ among a set of candidate functions $f_1, f_2, \ldots, f_N$.

In practice, the value of $\mathcal{R}(f)$ cannot be calculated because $P(\boldsymbol{x}, y)$ is unknown. A widely employed workaround in this case is

E-mail addresses: igorab@icmc.usp.br, igor.braga@bigdata.inf.br (I. Braga), mcmonard@icmc.usp.br (M.C. Monard).

http://dx.doi.org/10.1016/j.neucom.2014.12.097
0925-2312/

to use available data in a cross-validation setting, that is, to use some portion of the data to perform the minimization of Expression (2) for several candidates of K and γ, and to reserve the other portion for approximating $\mathcal{R}(f)$ and selecting the best K and γ. Conducting cross-validation in KRLS is relatively inexpensive compared to other learning methods, and this corresponds to the most interesting property of the method.

Given the universal approximation properties of the Gaussian RBF kernel [6]

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2}{\sigma^2}\right), \tag{4}$$

it has become the kernel of choice in much of machine learning research. However, these nice theoretical properties of the RBF kernel do not extend very well to practice. When combined with cross-validation and small training sets, RBF kernels have a great potential for overfitting. Recently, there has been a renewed interest in developing kernels with less potential for overfitting while retaining a good approximation property [7].

The kernel regularized least squares method is computationally efficient in small sample situations, although it may be rendered ineffective by the issues plaguing the popular combination of cross-validation and RBF kernels. Having that in mind, in this paper we follow [7] and investigate the use of splines as a safer choice to compose a multidimensional kernel function. We go one step further in this work and propose the use of additive spline kernels instead of multiplicative ones. We have found experimental evidence that the additive version is more appropriate to regression estimation in small sample situations.

We then proceed by investigating alternative statistical and heuristic procedures for the selection of the regularization parameter γ. The procedures we consider were shown to perform well for other regression methods, and, to the best of our knowledge, have not been applied to KRLS before. Surprisingly, though, most of these procedures fail to outperform cross-validation in small sample situations. A notable exception is the Finite Prediction Error (FPE) method, which has performed as well as cross-validation when both were used in combination with the additive spline kernel.

The remainder of this paper is organized as follows. In Section 2 we show how the minimization problem in Expression (2) is solved for fixed K and γ. In Section 3 we describe the issues surrounding the choice of a kernel function and present arguments in defense of the additive spline kernel. In Sections 4 and 5 we describe statistical and heuristic procedures used in this work to perform parameter selection, starting with an explanation on how to efficiently conduct leave-one-out cross-validation in KRLS. In Section 6 we report experimental evidence in favor of the additive spline kernel and also the results of the experimental evaluation of the considered parameter selection procedures. We conclude and give indications of future work in Section 7.

2. Solving the minimization problem of KRLS

The content in this section is informational and also introduces some notation used afterwards. To start with, note that KRLS requires the choice of a symmetric, positive definite kernel function $k : (R^d \times R^d) \mapsto R$ that spans the set of functions $\mathcal{H}_K$ under consideration. An example of such function is the well-known Gaussian RBF kernel—Expression (4). In this section, we assume that a kernel function $k(\boldsymbol{x}', \boldsymbol{x})$ has already been chosen, including eventual parameters.

By the representer theorem [8], the minimizer in Expression (2) has an expansion of the form

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i k(\boldsymbol{x}_i, \boldsymbol{x}), \quad \alpha_i \in R. \tag{5}$$

Hereafter, we denote by $\boldsymbol{y}$ the $n \times 1$ vector $[y_1, \ldots, y_n]^\top$ and by K the $n \times n$ matrix with entries $k_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. We also denote by $\boldsymbol{\alpha}$ the $n \times 1$ vector $[\alpha_1, \ldots, \alpha_n]^\top$.

Plugging Expression (5) into Expression (2) yields the following expression for calculating the squared loss:

$$\frac{1}{n}\sum_{i=1}^{n} \left(y_i - f(\boldsymbol{x}_i)\right)^2 = \frac{1}{n}\boldsymbol{\alpha}^\top KK\boldsymbol{\alpha} - \frac{2}{n}\boldsymbol{\alpha}^\top K\boldsymbol{y} + const. \tag{6}$$

Moreover, by considering the special properties of the norm in an RKHS, we have that $\|f\|_K^2 = \boldsymbol{\alpha}^\top K\boldsymbol{\alpha}$. Ignoring the constant term in Expression (6), we arrive at the following quadratic minimization problem for Expression (2):

$$\boldsymbol{\alpha}_\gamma = \arg\min_{\boldsymbol{\alpha} \in R^n} \left[\frac{1}{n}\boldsymbol{\alpha}^\top KK\boldsymbol{\alpha} - \frac{2}{n}\boldsymbol{\alpha}^\top K\boldsymbol{y} + \gamma\boldsymbol{\alpha}^\top K\boldsymbol{\alpha}\right]. \tag{7}$$

A necessary and sufficient condition for the solution of this minimization problem is obtained by taking the derivative of Expression (7) with respect to each α_i and equating it to zero. By doing that, we arrive at the following system of linear equations:

$$\frac{2}{n}KK\boldsymbol{\alpha}_\gamma - \frac{2}{n}K\boldsymbol{y} + 2\gamma K\boldsymbol{\alpha}_\gamma = 0. \tag{8}$$

Denoting by I the $n \times n$ identity matrix, extracting $1/n$ from γ, and solving for $\boldsymbol{\alpha}_\gamma$ in Expression (8), we arrive at the solution of the minimization problem in Expression (7):

$$\boldsymbol{\alpha}_\gamma = (K + \gamma I)^{-1}\boldsymbol{y}. \tag{9}$$

Plugging (9) into Expression (5) yields the closed form expression for the function minimizing Expression (2).

Most model selection procedures require the calculation of $\boldsymbol{\alpha}_\gamma$ for a fair number of γ candidates. In order to avoid solving one system of linear equations for each new γ, one can take advantage of the eigendecomposition of the kernel matrix: $K = U\Sigma U^\top$, where U is the $n \times n$ matrix formed by the eigenvectors of K and Σ is the $n \times n$ diagonal matrix containing the eigenvalues σ_i of K. Denoting by Λ_γ the $n \times n$ diagonal matrix with entries $\lambda_{ii} = 1/(\sigma_i + \gamma)$, $\boldsymbol{\alpha}_\gamma$ can be calculated by performing only matrix multiplications

$$\boldsymbol{\alpha}_\gamma = U\Lambda_\gamma U^\top \boldsymbol{y}. \tag{10}$$

Both the eigendecomposition of a matrix and a typical algorithm for solving a dense system of linear equations can be carried out in $O(n^3)$ time, with smaller constants for the latter. However, the eigendecomposition may still be preferable depending on n and the number of γ candidates considered.

3. Choosing a kernel function for KRLS

The choice of a kernel function for kernel regularized least squares defines the set of functions where the minimization of Expression (2) occurs. For example, if a linear kernel $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^\top \boldsymbol{x}_j$ is chosen, then the function obtained by KRLS will be a hyperplane in the input space, which is enough for learning linear regressions. However, the regression function is not linear in the input space for a variety of practical problems. This is why we have to choose between kernel functions that generate non-linear functions in the input space.

A typical non-linear kernel is the widely used Gaussian Radial Basis Function (RBF)—Expression (4). In fact, this expression defines a family of kernel functions parameterized by $\sigma > 0$, the so-called width parameter. By controlling σ, it is possible to achieve universal