TIME SERIES PREDICTION BASED ON THE RELEVANCE VECTOR MACHINE WITH ADAPTIVE KERNELS

Joaquin Quiñonero-Candela and Lars Kai Hansen

Informatics and Mathematical Modelling, Technical University of Denmark
Richard Petersens Plads, Building 321, DK-2800 Kongens Lyngby, Denmark
{jqc,lkhansen}@imm.dtu.dk

ABSTRACT

The *Relevance Vector Machine* (RVM) introduced by Tipping is a probabilistic model similar to the widespread *Support Vector Machines* (SVM), but where the training takes place in a Bayesian framework, and where predictive distributions of the outputs instead of point estimates are obtained. In this paper we focus on the use of RVM's for regression. We modify this method for training generalized linear models by adapting automatically the width of the basis functions to the optimal for the data at hand. Our Adaptive RVM is tried for prediction on the chaotic Mackey-Glass time series. Much superior performance than with the standard RVM and than with other methods like neural networks and local linear models is obtained.

1. INTRODUCTION

Generalized linear models perform a nonlinear projection of the input space into a transformed space by means of a set of nonlinear basis functions. A pure linear model is then applied to the transformed space, whose dimension is equal to the number of nonlinear basis functions. Given an input $\mathbf{x}$, the output of the generalized linear model is given by

$$y(\mathbf{x}) = \sum_{j=1}^{M} \omega_j \, \phi_j(\mathbf{x}) + \omega_0 \tag{1}$$

where $\{\phi_j\}$ are the nonlinear basis functions and $\{\omega_j\}$ are the model 'weights'. Unlike in the *Support Vector Machines* (SVM) framework where the basis functions must satisfy Mercer's kernel theorem, in the RVM case there is no restriction on the basis functions [1, 2]. In our case, the basis functions are chosen as Gaussians centered on each of the training points. The model we use can be seen as a particular case of a single hidden layer RBF network with Gaussian radial basis functions centered on the training points.

This work is funded by the EU *Multi-Agent Control* Research Training Network - EC TMR grant HPRNCT-1999-00107.

Like SVM's, RVM's yield a sparse solution, i.e., the model is built on a few 'key' training vectors only (like a pruned version of the particular RBF network). But as in the SVM case, no optimization of the basis functions is performed along with the training of the model weights. We propose a modification of the RVM algorithm that includes the optimization of the basis functions, in particular of the variance of the Gaussian functions that we use. We will show that our Adaptive RVM allows the model to be virtually non-parametric, while the performance of basic RVM's depends dramatically on a good choice of the parameters of the basis functions.

In the next section, we summarize the Bayesian framework used to train RVM's, and in Section 3 we highlight the importance of adapting the basis functions and present our improvement to the RVM. Finally, we compare the Adaptive RVM algorithm with other methods for predicting the Mackey-Glass chaotic time series.

2. THE RELEVANCE VECTOR MACHINE

Once the basis functions of the model described in equation (1) are defined, a maximum likelihood approach like the normal equations could be used for training the model weights $\{\omega_j\}$. Training such a flexible linear model, with as many parameters (weights) as training examples using maximum likelihood leads to over-fitting. Generalization capability can be pursued by doing the training in a Bayesian framework.

Rather than attempting to make point predictions of the optimal value of the model weight parameters, a *prior* distribution is defined over each of the weights. In the RVM framework, Gaussian prior distributions are chosen:

$$p(\omega_j|\alpha_j) = \sqrt{\frac{\alpha_j}{2\pi}} \exp\left(-\frac{1}{2}\alpha_j \, \omega_j^2\right) \tag{2}$$

where α_j is the *hyperparameter* that governs the prior defined over the weight ω_j.

Given a set of input-target training pairs $\{x_i, t_i\}_{i=1}^N$, assuming that the targets are independent and that the noise of the data is Gaussian with variance σ^2, the likelihood of the training set can be written as

$$p(\mathbf{t}|\boldsymbol{\omega},\sigma^2) = \left(2\pi\sigma^2\right)^{-N/2} \exp\left(-\frac{1}{2\sigma^2}\left\|\mathbf{t}-\mathbf{\Phi}\,\boldsymbol{\omega}\right\|^2\right) \quad (3)$$

where $\mathbf{t} = (t_1,\ldots,t_N)^T$, $\boldsymbol{\omega} = (\omega_1,\ldots,\omega_N)^T$ and $\mathbf{\Phi}$ is a matrix whose rows contain the response of all basis functions to the inputs $(\mathbf{\Phi})_{i,:} = [1,\phi_1(\mathbf{x}_i),\ldots,\phi_N(\mathbf{x}_i)]$.

With the prior and the likelihood distributions, the *posterior* distribution over the weights can be computed using Bayes rule

$$p(\boldsymbol{\omega}|\mathbf{t},\boldsymbol{\alpha},\sigma^2) = \frac{p(\mathbf{t}|\boldsymbol{\omega},\sigma^2)p(\boldsymbol{\omega}|\boldsymbol{\alpha})}{p(\mathbf{t}|\boldsymbol{\alpha},\sigma^2)} \quad (4)$$

where $\boldsymbol{\alpha} = (\alpha_0,\ldots,\alpha_N)^T$. The resulting posterior distribution over the weights is the multi-variate Gaussian distribution

$$p(\boldsymbol{\omega}|\mathbf{t},\boldsymbol{\alpha},\sigma^2) = \mathcal{N}(\boldsymbol{\mu},\mathbf{\Sigma}) \quad (5)$$

where the covariance and the mean are respectively given by:

$$\begin{aligned}\mathbf{\Sigma} &= (\sigma^{-2}\mathbf{\Phi}^T\mathbf{\Phi}+\mathbf{A})^{-1} && (6)\\ \boldsymbol{\mu} &= \sigma^{-2}\mathbf{\Sigma}\mathbf{\Phi}^T\mathbf{t} && (7)\end{aligned}$$

with $\mathbf{A} = \mathrm{diag}(\alpha_0,\ldots,\alpha_N)$.

The likelihood distribution over the training targets, given by equation (3), can be "marginalized" by integrating out the weights:

$$p(\mathbf{t}|\boldsymbol{\alpha},\sigma^2) = \int p(\mathbf{t}|\boldsymbol{\omega},\sigma^2)\,p(\boldsymbol{\omega}|\boldsymbol{\alpha})\,d\boldsymbol{\omega} \quad (8)$$

to obtain the *marginal likelihood* for the hyperparameters:

$$p(\mathbf{t}|\boldsymbol{\alpha},\sigma^2) = \mathcal{N}(0,\mathbf{C}) \quad (9)$$

where the covariance is given by $\mathbf{C} = \sigma^2\mathbf{I}+\mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^T$.

In the RVM scheme, the estimated value of the model weights is given by the mean of the posterior distribution (5), which is also the *maximum a posteriori* (MP) estimate of the weights. The MP estimate of the weights depends on the value of the hyperparameters $\boldsymbol{\alpha}$ and of the noise σ^2. The estimate of these two variables $\hat{\boldsymbol{\alpha}}$ and $\hat{\sigma}^2$ is obtained by maximizing the marginal likelihood (9).

The uncertainty about the optimal value of the weights reflected by the posterior distribution (5) is used to express uncertainty about the predictions made by the model. Given a new input $\mathbf{x}_*$, the probability distribution of the corresponding output is given by the *predictive distribution*

$$p(t_*|\mathbf{x}_*,\hat{\boldsymbol{\alpha}},\hat{\sigma}^2) = \int p(t_*|\mathbf{x}_*,\boldsymbol{\omega},\hat{\sigma}^2)\,p(\boldsymbol{\omega}|\mathbf{t},\hat{\boldsymbol{\alpha}},\hat{\sigma}^2)\,d\boldsymbol{\omega} \quad (10)$$

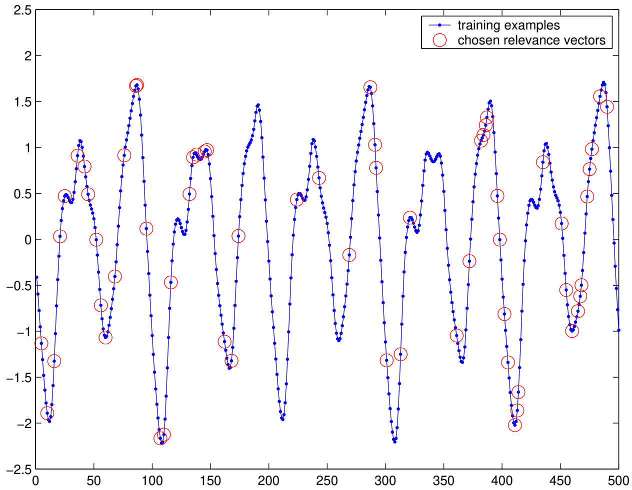

Fig. 1. Relevance Vectors chosen from the training set to build a generalized linear model for prediction.

which has the Gaussian form

$$p(t_*|\mathbf{x}_*,\hat{\boldsymbol{\alpha}},\hat{\sigma}^2) = \mathcal{N}(y_*,\sigma_*^2) \quad (11)$$

where the mean and the variance (*uncertainty*) of the prediction are respectively

$$\begin{aligned} y_* &= (\mathbf{\Phi})_{i,:}\,\boldsymbol{\mu} && (12)\\ \sigma_*^2 &= \hat{\sigma}^2 + (\mathbf{\Phi})_{i,:}\,\mathbf{\Sigma}\,(\mathbf{\Phi})_{i,:}^T && (13)\end{aligned}$$

The maximization of the marginal likelihood (9) with respect to $\boldsymbol{\alpha}$ and σ^2 is performed iteratively, as there is no closed solution [1]. In practice, during the iterative re-estimation many of the hyperparameters α_j approach infinity, yielding a posterior distribution (5) of the corresponding weight ω_j that tends to be a delta function centered around zero. The corresponding weight is thus deleted from the model, as well as its associated basis function $\phi_j(\mathbf{x})$. In the RVM framework, each basis function $\phi_j(\mathbf{x})$ is associated to (or centered around) a training example $\mathbf{x}_j$ so that $\phi_j(\mathbf{x}) = g(\mathbf{x}_j,\mathbf{x})$. The model is built on the few training examples whose associated hyperparameters do not go to infinity during the training process, leading to a sparse solution. These remaining examples are called the *Relevance Vectors* (RV).

We here want to examine the RVM approach for time series prediction. We choose a hard prediction problem, the MacKey-Glass chaotic time series, which is well-known for its strong non-linearity. Optimized non-linear models can have a prediction error which is three orders of magnitude lower than an optimized linear model [3]. Figure 1 shows a piece of the chaotic time series and we have furthermore marked the training targets associated to the RV's extracted from a training set composed by 500 samples of the Mackey-Glass chaotic time series.

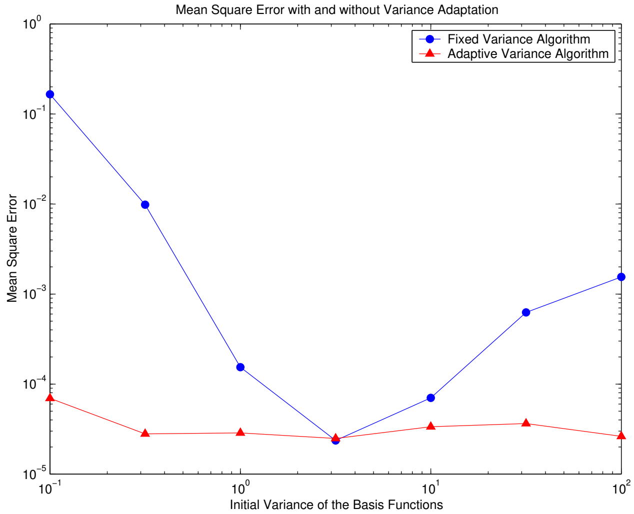

Fig. 2. Prediction mean square error with and without adapting the variance of the basis functions.

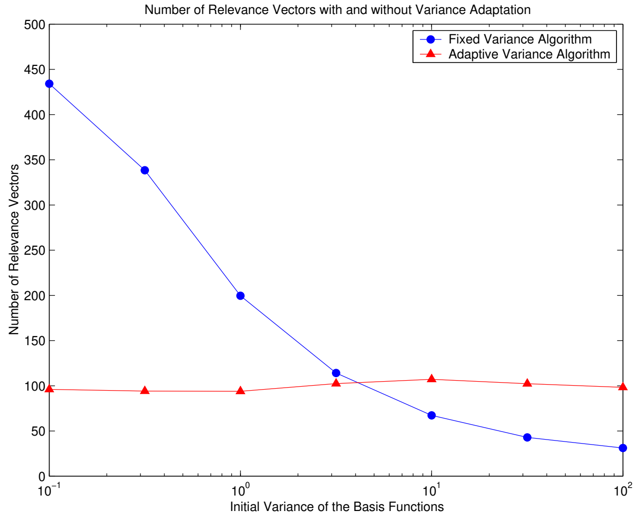

Fig. 3. Number of RV's selected with and without adapting the basis functions with respect to their initial width.

The Mackey-Glass attractor is a non-linear chaotic system described by the following equation:

$$\frac{dz(t)}{dt} = -bz(t) + a\frac{z(t-\tau)}{1+z(t-\tau)^{10}} \qquad (14)$$

where the constants are set to $a = 0.2$, $b = 0.1$ and $\tau = 17$. The series is resampled with period 1 according to standard practice. The inputs are formed by $L = 16$ samples spaced 6 periods from each other $\mathbf{x}_k = [z(k-6), z(k-12), \ldots, z(k-6L)]$ and the targets are chosen to be $t_k = z(k)$ to perform six steps ahead prediction [3].

The standard RVM approach is used, with Gaussian basis functions of fixed variance $\nu^2 = 5$.

3. ADAPTING THE BASIS FUNCTIONS

In the training process of a generalized linear model (1) under the RVM scheme described in the previous section, only the weights and hyperparameters are optimized. It is assumed that the basis functions are given. Yet the performance of the model depends dramatically on the choice of the basis functions and the value of their parameters. In the work presented in this paper the basis functions are isotropic Gaussian functions of the same variance, one centered on each training point. The variance is held constant in the conventional RVM approach, while we optimize it in the Adaptive RVM.

The importance of the kernel width parameter is illustrated in Figure 2. We build a generalized linear model (1), that we train using both the conventional RVM scheme, and our adaptive version of it for a time series prediction problem. We here use 700 training examples, and a large set of 8500 test examples to monitor performance. The upper curve in Figure 2 shows the mean square error obtained by training the RVM for a set of increasing widths of the basis functions. Each experiment is repeated 10 times: average values are represented. We note that the performance heavily depends on the width of the basis functions. The similar experiment using the adaptive scheme, described below, where the variance is optimized from variable initial values, systematically improves performance relative to the fixed variance case.

For a given number of training examples, the number of RV chosen depends on the variance of the basis functions. Figure 3 shows the number of RV's chosen as a function of the initial variance both for the conventional and the adaptive approaches. Our adaptive approach selects the number of RV's that allows the best performance, independently of the initial value of the basis functions' variance.

The RVM method iteratively maximizes the marginal likelihood (9) with respect to the hyper-parameters $\boldsymbol{\alpha}$ and to the noise σ^2. We can re-write the marginal likelihood to explicitly condition it on the variance ν^2 of the Gaussian basis functions

$$p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2, \nu^2) = \mathcal{N}(0, \sigma^2\mathbf{I} + \mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^{\mathbf{T}}) \qquad (15)$$

which depends on ν^2 through the basis functions matrix $\mathbf{\Phi}$.

In our approach, we maximize (15) with respect to ν^2 at each iteration. This is done by maximizing the logarithm of the marginal likelihood. As the width of the basis functions is equal for all, we have to solve a 1D search problem. Evaluating the derivative of the logarithm of (15) with respect to ν^2 is computationally much more expensive than just evaluating the marginal likelihood, hence we decided to use a direct search method due to Hooke and Jeeves [4].

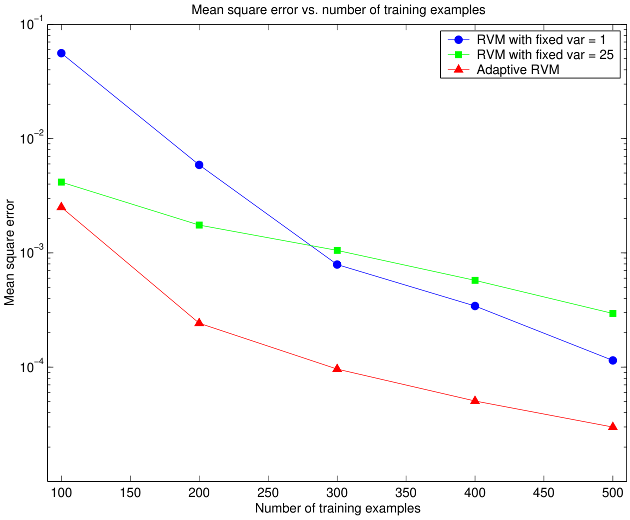

Fig. 4. Prediction mean square error as a function of the number of training examples, for a big and a small value of the variance for the conventional RVM and for the Adaptive RVM.

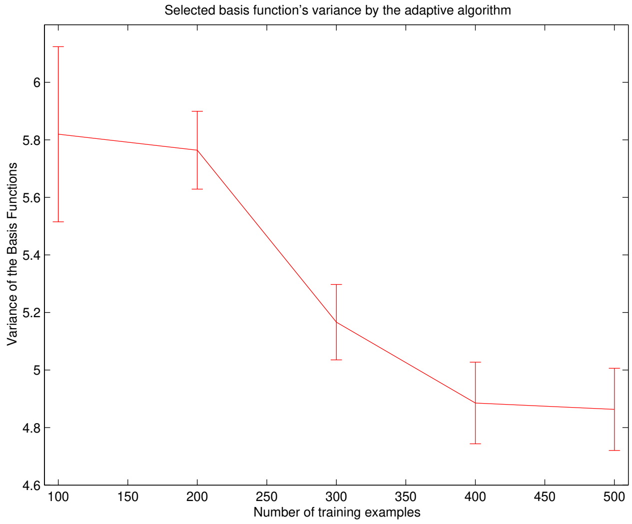

Fig. 5. Value of the variance ν^2 chosen by the Adaptive RVM for different numbers of training examples.

From Figure 2 it appears clearly that for a given number of training examples, there exists an optimal value the basis function width ν^2. But this optimal value depends on the number of training examples, as can be seen from Figure 4. While the conventional RVM performs well for the number of training examples that suits it's fixed ν^2, our approach adapts ν^2 to an optimal value. Figure 5 illustrates how the optimal value of ν^2 decreases for larger training sets, the number of RV's was also found to increase (data not shown).

	Train	**Test**
Simple linear model	9.7×10^{-2}	9.6×10^{-2}
5 nearest-neighbors	4.8×10^{-7}	8.4×10^{-5}
Pruned network	3.1×10^{-5}	3.4×10^{-5}
Adaptive RVM	2.3×10^{-6}	5.5×10^{-6}

Table 1. Training and test mean square prediction error for the Mackey-Glass chaotic time series.

We compare our Adaptive RVM with a simple linear model, with a 5 nearest-neighbors local linear model and with the pruned neural network used in [3] for 6 steps ahead prediction. The training set contains 1000 examples, and the test set 8500 examples. Average values of 10 repetitions are presented. The Adaptive RVM uses an average of 108 RV's in this example. It is remarkable that the Adaptive RVM so clearly outperforms a carefully optimized MLP, we currently investigate other time series prediction problems in order to test the hypothesis that highly non-linear problems are better modeled by non-parametric models with Bayesian complexity control.

4. CONCLUSIONS

Sparse generalized linear models like the RVM (and SVM's) present excellent performance on time series prediction, but are severely limited by the manual choice of the parameters of the basis functions. To overcome this limitation, we propose the Adaptive RVM that automatically optimizes the parameters of the basis functions. The resulting time series predictor outperforms a carefully optimized artificial neural network. The approach can be generalized to locally adapt the kernel widths yielding an even more flexible predictor, however, optimization then becomes non-trivial.

Acknowledgements We would like to thank Carl E. Rasmussen, Michael Saunders and Hans Bruun Nielsen for useful discussion.

5. REFERENCES

[1] Michael E. Tipping, "Sparse bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[2] V. N. Vapnik, *Statistical Learning Theory*, Wiley, New York, 1998.

[3] Claus Svarer, Lars K. Hansen, Jan Larsen, and Carl E. Rasmussen, "Designer networks for time series processing," *Proceedings of the III IEEE Workshop on Neural Networks for Signal Processing*, pp. 78–87, 1993.

[4] R. Hooke and T. A. Jeeves, " "direct search" solution of numerical and statistical problems," *J. Assoc. Comput.*, pp. 212–229, March 1961.