A COMPARATIVE STUDY OF PRE-IMAGE TECHNIQUES: THE KERNEL AUTOREGRESSIVE CASE

Maya Kallas [1,2], Paul Honeine [1], Clovis Francis [2] and Hassan Amoud [3]

[1] Institut Charles Delaunay (UMR CNRS 6279), LM2S, Université de Technologie de Troyes, France
[2] Laboratoire d'analyse des systèmes (LASYS), Université Libanaise, Lebanon
[3] Azm Center for Research in Biotechnology and its Applications, Lebanese University, Lebanon

ABSTRACT

The autoregressive (AR) model is one of the most used techniques for time series analysis, applied to study stationary as well as non-stationary processes. However, being a linear technique, it is not adapted for nonlinear systems. Recently, we introduced the kernel AR model, a straightforward extension of the AR model to the nonlinear case. It is based on the concept of kernel machines, where data are nonlinearly mapped from the input space to a feature space. The AR model can thus be applied on the mapped data. Nevertheless, in order to predict future samples, one needs to go back to the input space, by solving the pre-image problem. The prediction performance highly depends on the considered pre-image technique. In this paper, a comparative study of several state-of-the-art pre-image techniques is conducted for the kernel AR model, investigating the prediction error with the optimal model parameters, as well as the computational complexity. The conformal map approach presents results as good as the well known fixed-point iterative method, with less computational time. This is shown on unidimensional and multidimensional chaotic time series.

Index Terms— kernel machines, autoregressive model, nonlinear models, pre-image problem, prediction

1. INTRODUCTION

Time series analysis and prediction is of an important role in many domains. One of the most useful technique for time series analysis is the autoregressive (AR) model. This model is based on the prediction of future samples using a linear combination of a (usually small) number of previous samples. The model parameters include the weights in the linear expansion and the model order, i.e., the number of previous samples. Once these parameters estimated, the AR model allows to predict future samples. Owing to linear algebra, the AR technique for modeling and prediction is easy to compute and implement. Still, it is based on a linear model, unadapted to study nonlinear systems.

This work is partly supported by the Lebanese University and the French-Lebanese research program CEDRE No. 10 SCI F15/L5.

In order to provide nonlinear models, one may consider the concept of kernel machines. It is based on mapping the data from the input space to a feature space where the data is assumed linear. Estimating the model parameters in the feature space can be done using inner products between mapped data, evaluated by using a kernel function, without the need to explicit the feature space. This is known as the *kernel trick* [1], mostly known by Vapnik's Support Vector Machines (SVM) [2]. It allows to derive nonlinear techniques based on linear ones, essentially without additional computational cost, as illustrated with kernel principal component analysis (PCA), kernel Fisher discriminant analysis, and SVM novelty detection, only to name a few. See [3] for a survey. The concept of the kernel machines have been applied for time series analysis, for instance with the support vector regression [2], and the kernel Kalman filter [4]. However, these machines do not tackle the simplicity of the AR model, resulting into high computational cost.

To adapt the simplicity of the AR concept for nonlinear systems, we recently introduced in [5] the so-called kernel AR model, by combining the flexibility and easiness of both the AR model and the kernel machines to incorporate nonlinearities. To this end, samples are nonlinearly mapped from the input space into a high-dimensional feature space. The AR model parameters are determined by evaluating a kernel function, without the need to exhibit the nonlinear map. While the model parameters are evaluated in the feature space, the predicted future samples need to be estimated back into the input space, i.e. the space of samples. Thus, a pre-image technique, mapping from feature space to the input space, is required. Without a pre-image technique, the kernel AR model fails to perform a prediction scheme, as illustrated in [6].

The prediction performance of the kernel AR model depends highly on the pre-image technique. Several techniques have been proposed to solve the ill-posed pre-image problem. This problem is initially introduced in [7], where a fixed-point iterative method is proposed. This method, as well as a gradient descent approach, suffer from numerical instabilities and local minima. Another pre-image technique has been proposed in [8], where a multidimensional-scaling approach is

presented, considering the relationship between feature-space distances and input-space distances. Lately, a more direct method based on the relationship between inner-products in both spaces [9]. See [10] for a recent review, with several applications in signal processing. The latter technique is as good as the fixed-point iterative one with low computation time, and does not suffer from any instabilities. In this paper, we study the influence of these pre-image techniques on the performance of the kernel AR model for prediction, as well as an estimation of the computational time.

The rest of the paper is organized as follows: In the next section, we present the kernel AR model. In Section 3, the prediction scheme for evaluating the future samples is derived, with several pre-image techniques considered. Section 4 provides a comparative study of these pre-image techniques on two univariate (*Laser* and MG_{30}) and two multidimensional time series (*Ikeda map* and *Lorenz attractor*).

2. KERNEL AR MODEL

Let us first define the classical AR model, widely used in analysis of stationary and non-stationary time series. The main idea behind the AR model is to predict future samples based on a linear combination of some previous ones. This linear combination of previous samples is defined by some parameters α_i for $i = 1, 2, \ldots, p$, where p denotes the order of the model, i.e., the number of previous samples in the expansion. Taking for example a time series $x_1, x_2, \ldots, x_n$, an AR model of order *p* is defined by

$$x_i = \sum_{j=1}^{p} \alpha_{p-j+1}\, x_{i-j} + \varepsilon_i, \tag{1}$$

for $i = p+1, \ldots, n$, where $\alpha_1, \alpha_2, \ldots, \alpha_p$ are constants representing the model parameters and ε_i is assumed to be a white noise with a zero mean. The optimal model parameters, as well as the model order, can be estimated by minimizing the least square prediction error, namely

$$\min_{\boldsymbol{\alpha}} \sum_{i=p+1}^{n} \left(x_i - \Big(\sum_{j=1}^{p} \alpha_{p-j+1}\, x_{i-j} + \varepsilon_i \Big) \right)^2 .$$

The optimal values are obtained by setting to zero the derivative of the above cost function with respect to each parameter. Once the model parameters are defined, one can use the model (1) to predict future samples. The resulting model, connecting the previous samples to future ones, is linear.

In order to extend the linear model into a nonlinear one, we consider the concept of kernel machines. A kernel is a symmetric and continuous function defined by $\kappa\colon \mathcal{X} \times \mathcal{X} \mapsto \mathcal{X}$, where $\mathcal{X}$ is an input space with the Euclidean dot product $\boldsymbol{x}_i \cdot \boldsymbol{x}_j$ for any $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}$. If $\sum_{i,j} \alpha_i \alpha_j \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0$ for all $\alpha_i, \alpha_j \in \mathbb{R}$ and all $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}$, then $\kappa(\cdot, \cdot)$ is a positive semi-definite kernel. Moore-Aronszajn theorem [11] states that each positive semi-definite kernel corresponds to a unique (up to an isometry) feature space and vice-versa. This feature space $\mathcal{H}$ is obtained using a mapping function $\Phi\colon \mathcal{X} \mapsto \mathcal{H}$, such that

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \Phi(\boldsymbol{x}_i), \Phi(\boldsymbol{x}_j) \rangle_{\mathcal{H}}, \tag{2}$$

for any $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}$, where $\langle \cdot\,, \cdot \rangle_{\mathcal{H}}$ denotes the corresponding inner product in $\mathcal{H}$, and let $\| \cdot \|_{\mathcal{H}}$ be its corresponding norm. Many machine learning techniques can be written in terms of inner product of data. By substituting the canonical inner product by a positive semi-definite kernel, one implicitly applies a mapping function Φ. This is the so-called *kernel trick*. The kernel is often referred to as the *reproducing kernel* and the feature space is the *reproducing kernel Hilbert space* (RKHS).

To derive the kernel AR model, we consider a nonlinear mapping function $\Phi(\cdot)$ from the input space $\mathcal{X}$ to some feature space $\mathcal{H}$. Thus, for a time series, the sequence $x_1, x_2, \ldots, x_n$ in the input space is mapped into the set of images $\Phi(x_1), \Phi(x_2), \ldots, \Phi(x_n)$ in the feature space. Using the same AR principle, each sample x_i is now replaced by its corresponding $\Phi(x_i)$, thus (1) is now written as

$$\Phi(x_i) = \sum_{j=1}^{p} \alpha_{p-j+1}\, \Phi(x_{i-j}) + \varepsilon_i^{\Phi}, \tag{3}$$

where the result $\Phi(x_i)$ is defined in the feature space [5]. In a more compact way, the above equation can be represented matrix-wise by

$$\Phi(x_i) = \boldsymbol{\varphi}_i\, \boldsymbol{\alpha},$$

where $\boldsymbol{\alpha}$ is the vector of the model parameters, $\boldsymbol{\alpha} = [\alpha_p, \alpha_{p-1}, \cdots, \alpha_1]^\top$, and $\boldsymbol{\varphi}$ contains the last p previous mapped samples, that is $\boldsymbol{\varphi}_i = [\Phi(x_{i-1}), \Phi(x_{i-2}), \cdots, \Phi(x_{i-p})]$.

By analogy with the linear form of kernel AR model, the model parameters α_{p-j+1} are estimated by minimizing the least square prediction error, between the estimated value $\sum_{j=1}^{p} \alpha_{p-j+1} \Phi(x_{i-j})$ and the real one mapped to $\Phi(x_i)$. For a sequence of n available samples, we minimize with respect to $\boldsymbol{\alpha}$ the cost function

$$\min_{\boldsymbol{\alpha}} \sum_{i=p+1}^{n} \Big\| \Phi(x_i) - \sum_{j=1}^{p} \alpha_{p-j+1} \Phi(x_{i-j}) \Big\|_{\mathcal{H}}^2 . \tag{4}$$

By expanding this expression using the definition of $\| \cdot \|_{\mathcal{H}}^2 = \langle \cdot, \cdot \rangle_{\mathcal{H}}$, and then taking its derivative with respect to $\boldsymbol{\alpha}$, and setting it to zero, we get the optimal parameters $\boldsymbol{\alpha}$ of the nonlinear AR model with

$$\boldsymbol{\alpha} = \left(\sum_{i=p+1}^{n} \boldsymbol{K}_i \right)^{-1} \left(\sum_{i=p+1}^{n} \boldsymbol{\kappa}_i \right),$$

where $\boldsymbol{K}_i = \boldsymbol{\varphi}_i^\top \boldsymbol{\varphi}_i$ and $\boldsymbol{\kappa}_i = \boldsymbol{\varphi}_i^\top \Phi(x_i)$ is a column vector of p entries. It is obvious that the optimal parameters require the inversion of a p-by-p matrix, p is often small.

3. PRE-IMAGE TECHNIQUES

Once the model parameters $\boldsymbol{\alpha}$ estimated, we can now predict future samples based on (5), namely for $i > n$

$$\psi_i = \sum_{j=1}^{p} \alpha_{p-j+1}\, \Phi(x_{i-j}). \tag{5}$$

In this expression, the result is denoted by ψ_i, since such result in the feature space may not correspond to a valid image of some arbitrary sample in the input space. Therefore, we need to go back to the input space and determine the counterpart in the original space having its image ψ_i in the feature space. Nevertheless, the exact pre-image may seldom exist, and when it exists, it may not be unique. This is known as the *pre-image problem*, where one seeks an approximate solution x^* whose counterpart $\Phi(x^*)$ is as close as possible to ψ_i.

The resulting optimization problem is defined by

$$x_i^* = \arg\min_x \frac{1}{2}\|\Phi(x) - \psi_i\|_{\mathcal{H}}^2,$$

or equivalently by replacing ψ_i with its definition in the above equation, we obtain

$$x_i^* = \arg\min_x \frac{1}{2}\Big\|\Phi(x) - \sum_{j=1}^{p} \alpha_{p-j+1}\, \Phi(x_{i-j})\Big\|_{\mathcal{H}}^2.$$

Next, we consider the equivalent optimization problem

$$x_i^* = \arg\min_x J_i(x),$$

where

$$J_i(x) = -\sum_{j=1}^{p} \alpha_{p-j+1}\, \kappa(x_{i-j}, x) + \frac{1}{2}\kappa(x, x),$$

with the elimination of the term independent of x.

Many techniques have been proposed in literature, such as the gradient descent scheme, the fixed-point iterative method, the multi-dimensional scaling technique and the conformal map approach. We will describe each one of them in the following subsections.

3.1. Gradient descent scheme

The gradient descent scheme is a first-order iterative optimization technique. To solve the above optimization problem, one takes steps η_t proportional to the opposite direction of the gradient ∇_x with respect to x of the function J_i. This is expressed by

$$x_{i,t+1}^* = x_{i,t}^* - \eta_t \nabla_x J_i(x_{i,t}),$$

where the index t denotes the iterative technique. As this technique starts from an initial point, and presents an iterative scheme, one might have to run the algorithm many times from different initial points in order to obtain the global minimum, without getting stuck in local minima. In practice, the stepsize parameter η_t is constant, independent of the iteration t.

3.2. Fixed point iterative scheme

When using kernels, one can implement easily the fixed-point iterative scheme [7]. Based on the gradient descent scheme, we take into consideration the type of the kernel used, as illustrated next for the Gaussian and the polynomial kernels.

The Gaussian kernel is defined by

$$\exp(-\|x_i - x_j\|^2/2\sigma^2),$$

for any $x_i, x_j \in \mathcal{X}$, where σ is a tunable bandwidth parameter. The resulting cost function $J_i(x)$ is defined by

$$-\sum_{j=1}^{p} \alpha_{p-j+1} \exp(-\|x_{i-j} - x\|^2/2\sigma^2).$$

Taking the derivative of the above expression with respect to x, and setting it to zero, we get the fixed-point iterative expression

$$x_{i,t+1}^* = \frac{\sum_{j=1}^{p} \alpha_{p-j+1} \exp(-\|x_{i-j} - x_{i,t}^*\|^2/2\sigma^2)\, x_{i-j}}{\sum_{j=1}^{p} \alpha_{p-j+1} \exp(-\|x_{i-j} - x_{i,t}^*\|^2/2\sigma^2)}.$$

The polynomial kernel is defined by

$$(\langle x_i,\, x_j\rangle + c)^q,$$

where q is a positive integer, and c is a positive parameter often set to 1. The cost function $J_i(x)$ associated with this kernel is defined by

$$-\sum_{j=1}^{p} \alpha_{p-j+1}(\langle x_{i-j},\, x\rangle + c)^q + \frac{1}{2}(\langle x,\, x\rangle + c)^q,$$

leading to the fixed-point iterative expression

$$x_{i,t+1}^* = \frac{-\sum_{j=1}^{p} \alpha_{q-j+1}(\langle x_{i-j},\, x_{i,t}^*\rangle + c)^{q-1}\, x_{i-j}}{(\langle x_{i,t}^*,\, x_{i,t}^*\rangle + c)^{q-1}}.$$

3.3. Multi-Dimensional Scaling

For many commonly used kernels, there exists a simple relationship between $\|x_k - x_\ell\|$ in the input space and $\|\Phi(x_k) - \Phi(x_\ell)\|_{\mathcal{H}}$ in the feature space [12]. Using this concept, a multi-dimensional scaling (MDS) approach is considered in [8]. Consider the distance in the feature space, $\delta_{i,j} = \|\psi_i - \Phi(x_{i-j})\|_{\mathcal{H}}$, and its counterpart in the input space, $\|x^* - x_{i-j}\|$, for $j = 1, 2, \ldots, p$. Ideally, these distances are equal

$$\|x_i^* - x_{i-j}\| = \|\psi_i - \Phi(x_{i-j})\|_{\mathcal{H}},$$

for every $j = 1, 2, \ldots, p$. In order to find the pre-image, we evaluate the above equation for the p available samples $x_{i-1}, x_{i-2}, \ldots, x_{i-p}$, namely

$$2\langle x_i^*,\, x_{i-j}\rangle = \langle x_i^*,\, x_i^*\rangle + \langle x_{i-j},\, x_{i-j}\rangle - \delta_{i,j},$$

for $j = 1, 2, \ldots, p$. After taking the average of centered data, the pre-image value is obtained with

$$x_i^* = \frac{1}{2}(\boldsymbol{X}_i\boldsymbol{X}_i^\top)^{-1}\boldsymbol{X}_i\Big(\text{diag}(\boldsymbol{X}_i^\top\boldsymbol{X}_i) - [\delta_1^2\ \delta_2^2\ \cdots\ \delta_n^2]^\top\Big)$$

where $\boldsymbol{X}_i = [x_{i-1}, x_{i-2}, \cdots, x_{i-p}]$ and diag$(\cdot)$ is the diagonal operator. This expression is only valid for the Gaussian kernel.

3.4. Conformal Map Approach

Using the same strategy as MDS, not only the distance is conserved, the angular mesure is also the same by preserving inner product measures, since $x_i^\top x_k/\|x_i\|\|x_k\|$ defines the cosine of the angle between x_i and x_k. To this end, a coordinate system in the feature space is constructed having an isometry with respect to the input space. Once this coordinate system is build, any element in the feature space can be written as a combination its projection onto these coordinate functions. Details of the conformal map algorithm are given in [9]. This results into the following expression

$$x_i^* = (\boldsymbol{X}_i\boldsymbol{X}_i^\top)^{-1}\boldsymbol{X}_i\big(\boldsymbol{X}_i^\top\boldsymbol{X}_i - \eta\,\boldsymbol{K}_i^{-1}\big)\,\boldsymbol{\alpha},$$

where η is a tunable regularization parameter. It is worth noting that this expression is independent of the nature of the kernel type. As it is investigated in the next section, this technique presents very good results, such as the fixed-point technique and it only needs a fraction of time for the calculation to be done.

4. EXPERIMENTS

In order to give a comparative study for each of the aforementioned pre-image methods, we considered a well-defined benchmark of four time series:

- The *Laser* is a sequence of laser measurements taken from the Santa Fe competition (dataset A).
- The *Mackey-Glass* time series provides a model of the blood cells production evolution. It is defined by a delay differential equation

  $$\frac{dx(t)}{dt} = -0.1\,x(t) + \frac{0.2\,x(t-\tau)}{1 + x(t-\tau)^{10}}$$

  which, for values of τ greater than 16.8, shows some highly nonlinear chaotic behavior. For $\tau = 30$, the time series is denoted MG_{30}.
- The *Ikeda map* refers to a two dimensional series of laser dynamics. Starting from an initial point, $\boldsymbol{x}(0) = [x_1(0), x_2(0)]^\top$, it is defined by

  $$\begin{cases} \omega(t) = c1 - c3/(1 + x_1^2(t) + x_2^2(t)) \\ x_1(t+1) = r + c2(x_1(t)\cos\omega(t) + x_2(t)\sin\omega(t)) \\ x_2(t+1) = c2(x_1(t)\sin\omega(t) + x_2(t)\cos\omega(t)) \end{cases}$$

  where $c1, c2, c3$ and r are constants; in our case, we set $c1 = 0.4, c2 = 0.84, c3 = 6.0, r = 1.0$ and $\boldsymbol{x}(0) = [1\ \ 0.001]^\top$.
- A *Lorenz attractor* is the solution of the system defined by the following differential equations

  $$\begin{cases} \frac{dx(t)}{dt} = -a\,x(t) + a\,y(t) \\ \frac{dy(t)}{dt} = -x(t)\,z(t) + r\,x(t) - y(t) \\ \frac{dy(t)}{dt} = +x(t)\,y(t) - b\,z(t) \end{cases}$$

  For the experimentations, we set $a = 10$, $r = 28$ and $b = 8/3$.

For a comparative study, the parameters were learnt from the first 300 samples of the time series, and the mean square error (MSE) of prediction is estimated over the next 300 samples[1]. In the learning stage, the model parameters and its order were estimated, as well as the tunable parameters such as the kernel parameter (σ or q) and η. The optimal values for σ and η were estimated by a grid search within values $\{2^{-12}, 2^{-11}, \ldots, 2^{11}, 2^{12}\}$. For the polynomial kernel, the optimal value of the parameter q was chosen from $q \in \{1, 2, 3, 4, 5, 6\}$. Such values have been chosen in order to elaborate each and every one of these techniques. For the two iterative methods, 100 iterations were needed to arrive to the optimal values.

Results are given in Table 1 for the Gaussian kernel and Table 2 for the polynomial kernel. It is obvious that, independent of the kernel type, the iterative pre-image techniques such as the gradient descent scheme and the fixed-point iterative method required more computational time than the MDS and conformal map approaches. For the MSE, except the case of the *Laser* data with the Gaussian kernel, both the fixed-point and the conformal map techniques provided essentially the least MSE. It is clear that, taking into account the computational time, the conformal map is the best choice.

5. CONCLUSION

We presented the kernel AR model to predict future samples for any time series. This is done by combining the concept of kernel machines with the AR model. The prediction of upcoming samples is defined by a pre-image schema. In this

[1]The computational time was estimated on Matlab 2009 running on a desktop dual-core PC Pentium 3.4 GHz 1GO RAM.

Table 1. Comparison between several pre-image techniques applied on different types of time series using a Gaussian kernel.

		Laser	MG_{30}	*Ikeda*	*Lorenz*
Gradient	σ	2^{-10}	2^{-6}	2^{-3}	2^{3}
	time	3.0736	3.0953	6.0193	9.4399
	MSE	876.1293	0.0832	0.7187	150.0145
fixed-pt.	σ	2^{2}	2	2^{10}	2^{6}
	time	5.6392	8.2743	12.8256	34.2385
	MSE	16.5673	**0.0162**	**0.5194**	**0.00035**
MDS	σ	2^{10}	2^{-10}	2^{10}	2^{2}
	time	0.1002	0.1608	0.1976	0.2735
	MSE	**11.5991**	0.083	0.5825	99.2851
conformal	σ	2^{5}	2	2^{3}	2^{2}
	η	2^{-10}	2^{-10}	2^{-10}	2^{-10}
	time	1.0172	0.9916	1.9249	2.3637
	MSE	17.1484	**0.0166**	**0.5201**	0.1079

Table 2. Comparison between several pre-image techniques, using a polynomial kernel.

		Laser	MG_{30}	*Ikeda*	*Lorenz*
Gradient	q	2	5	6	2
	η	2^{-10}	2^{-2}	2^{-12}	2^{-11}
	time	1.9851	2.0025	3.7099	5.5710
	MSE	876.1293	0.1000	0.7187	339.3405
fixed-pt.	q	5	2	2	5
	time	7.2244	8.1867	19.0268	23.6776
	MSE	**16.0169**	**0.0161**	**0.5246**	**0.007**
conformal	q	2	2	2	2
	η	2^{-9}	2^{-10}	2^{-9}	2^{-10}
	time	0.5113	0.4877	0.9250	1.2632
	MSE	**18.6591**	**0.0160**	**0.5171**	**0.0025**

paper, a comparative study between several pre-image techniques is given. To this end, a Gaussian and a Polynomial kernels are used on four different time series. Experiments show that the conformal map, recently introduced by two of the authors, is the best pre-image technique from both the mean square prediction error and the computational time.

6. REFERENCES

[1] M. A. Aizerman, E. A. Braverman, and L. Rozonoer, "Theoretical foundations of the potential function method in pattern recognition learning," in *Automation and Remote Control*, 1964, number 25, pp. 821–837.

[2] V. N. Vapnik, *Statistical Learning Theory*, Wiley-Interscience, September 1998.

[3] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, 2004.

[4] L. Ralaivola and F. D'alche-Buc, "Time series filtering, smoothing and learning using the kernel kalman filter," in *Proc. IEEE International Joint Conference on Neural Networks*, 2005, vol. 3, pp. 1449–1454.

[5] M. Kallas, P. Honeine, C. Richard, C. Francis, and H. Amoud, "Kernel-based autoregressive modeling with a pre-image technique," in *IEEE Workshop on Statistical Signal Processing*, Nice, France, 28-30 June 2011.

[6] R. Kumar and C. V. Jawahar, "Kernel approach to autoregressive modeling," in *The 13th National Conference on Communications (NCC)*, Kanpur, India, January 2007.

[7] S. Mika, B. Schölkopf, A. Smola, K.-R. Müller, M. Scholz, and G. Rätsch, "Kernel PCA and de-noising

in feature spaces," in *Proceedings of the 1998 conference on Advances in neural information processing systems II*, Cambridge, MA, USA, 1999, pp. 536–542, MIT Press.

[8] J. T. Kwok and I. W. Tsang, "The pre-image problem in kernel methods," in *ICML*, T. Fawcett and N. Mishra, Eds. 2003, pp. 408–415, AAAI Press.

[9] P. Honeine and C. Richard, "Solving the pre-image problem in kernel machines: a direct method," in *Proc. IEEE Workshop on Machine Learning for Signal Processing (MLSP)*, Grenoble, France, September 2009.

[10] P. Honeine and C. Richard, "Pre-image problem in kernel-based machine learning," *IEEE Signal Processing Magazine*, vol. 28 (2), March 2011, to appear.

[11] N. Aronszajn, "Theory of reproducing kernels," *Trans. Amer. Math. Soc.*, vol. 68, pp. 337–404, 1950.

[12] C. K.I. Williams, "On a connection between kernel pca and metric multidimensional scaling," in *Advances in Neural Information Processing Systems 13*. 2001, pp. 675–681, MIT Press.