Kernel Approach to Autoregressive Modeling

Ranjeeth Kumar and C. V. Jawahar

Centre for Visual Information Technology
International Institute of Information Technology, Hyderabad, 500 032, India
jawahar@iiit.ac.in

Abstract

A kernel-based approach for nonlinear modeling of time series data is proposed in this paper. Autoregressive modeling is achieved in a feature space defined by a kernel function using a linear algorithm. The method extends the advantages of the conventional autoregressive models to characterization of nonlinear signals through the intelligent use of kernel functions.Experiments with synthetic signals demonstrate that this method seems to be a promising alternative to nonlinear modeling schemes.

1. Introduction

Nonlinear modeling of time series data has been studied extensively in the past [1]. A well-known statistical approach to the study of nonlinear relationships is first to transform the variables such that the relationship is linearized, and thereafter to use linear modeling techniques on the transformed data for developing a compact representation of the signal. This paradigm is similar to the one adopted in machine learning to detect complex patterns in data efficiently through the use of kernel functions [2] that avoid the explicit transformation of the data. This paper proposes a novel method for nonlinear modeling of signals by the application of an autoregressive model on the data mapped to a feature space defined by a kernel function. This extends the advantages of the autoregressive modeling techniques to the characterization of nonlinear signals. A shortcoming of the approach is that, in all but a few cases, the signal can not be reconstructed in the input space. However the model parameters thus estimated can be used as features that characterize the signal which in turn can be used for tasks like recognition and classification.

Many nonlinear modeling techniques require iterative optimization of objective functions. These iterative optimization procedures are computationally intensive and often numerically unstable. They could also fail to converge to the desired solution [3]. In contrast, linear models and algorithms are computationally efficient. They could also have closed-form solutions. Moreover linear algorithms [4] are numerically more stable and statistical inferences made using them are more reliable [5]. Due to the efficiency and reliability of linear methods they have been used widely for the tasks of modeling and recognition, even in cases where there is no adequate justification for the linearity assumption.

The motivation for the current work is to retain the advantages of the linear modeling schemes to solve the modeling problems, which need nonlinearity. The use of kernel functions is a natural way to achieve this. They have been widely used for detection of complex nonlinear patterns in the data that are linear in a feature space defined by the kernel function. We use autoregressive model along with kernel functions to perform linear modeling in a feature space defined by the kernel function. The proposed algorithms requires only a kernel matrix which is independent of the dimensionality of the associated feature space. The work of Chakrabartty et.al [6] that uses the kernel trick to capture higher order correlation between speech samples, in our knowledge, is the closest to the current one. They use growth transformation with kernel regression for extraction of robust speech features with empirical evidence on robustness of the kernel-based features.

We show that the parameters of an autoregressive model applied to the data in a feature space defined by the kernel function can be computed from the kernel matrix alone by minimizing the prediction error.

2. AR Model and Kernel Methods

This section reviews the required fundamentals and introduces the notations followed. Autoregressive (AR) model is a popular linear model employed for the modeling time series data in applications like speech processing, image compression, redundancy removal, on-line handwriting recognition etc. An AR model explains the univariate time series data by expressing the signal at instant i as

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

where the process mean is assumed to be zero and e_i is white noise. The value of p is known as the order of the AR model. This is a linear regression of the sample at instant i against previous p samples. The parameters of the model can be estimated by minimizing the squared prediction error

$$\xi = \sum_{i=p+1}^{l} (x_i - \hat{x}_i)^2 = \sum_{i=p+1}^{l} (x_i - \sum_{j=1}^{p} \alpha_{p-j+1} x_{i-j})^2$$

where l is the total number of samples.The solution can be obtained by setting $\frac{\partial \xi}{\partial \alpha} = 0$ and solving the resulting set of linear equations. Autocorrelation, Levinson-Durbin recursion are some of the popular methods for numerical computation. The modeling scheme can be extended to vector valued sequence $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_l$ in a straightforward manner with the definitions

$$\hat{\mathbf{x}}_i = \sum_{j=1}^{p} \alpha_{p-j+1} \mathbf{x}_{i-j}$$

$$\xi = \sum_{j=p+1}^{l} \|\mathbf{x}_j - \hat{\mathbf{x}}_j\|^2$$

Kernel methods [2] are a new class of algorithms used for detection of complex nonlinear patterns in the data. A Kernel Function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbf{R}$ is a symmetric bilinear real-valued function such that the Kernel Matrix $\mathbf{K}$ defined by κ i.e $\mathbf{K}_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$ restricted to finite subset of $\mathcal{X}$ is positive semi-definite where $\mathcal{X}$ is the input space. It can be shown that there exists a map $\phi : \mathcal{X} \to \mathbf{F}$ where $\mathbf{F}$ is a Hilbert Space with the inner product satisfying the condition

$$\kappa(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle. \tag{1}$$

For instance the kernel function $\kappa(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^t \mathbf{y})^2$ defines the inner product in a feature space corresponding to the map $\phi : \begin{bmatrix} x_1 & x_2 \end{bmatrix}^t \mapsto \begin{bmatrix} x_1^2 & x_2^2 & \sqrt{2} x_1 x_2 \end{bmatrix}^t$ which maps data in a 2-d space to a 3-d space. Thus algorithms requiring only the pairwise inner product information can be easily applied to the data in the feature space using a kernel function. In this paper, we argue that autoregressive modeling can be done in the feature space without the explicitly mapping the samples into a new space. Section 3 shows that the α_i can be computed by minimizing the error ξ in the feature space $\mathbf{F}$ defined by the kernel function κ from the knowledge of the kernel matrix alone.

3. AR Model in the Feature Space

The map $\phi : \mathcal{X} \to \mathbf{F}$ maps the elements of the sequence $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_l$ to the sequence $\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \cdots \phi(\mathbf{x}_l)$ in the feature space. We use the notation

$$\mathbf{X} = \begin{bmatrix} \phi(\mathbf{x}_1) & \phi(\mathbf{x}_2) & \cdots & \phi(\mathbf{x}_l) \end{bmatrix}$$

for the data matrix and $\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_p \end{bmatrix}^t$ for the vector of prediction coefficients. Thus the prediction equation becomes

$$\widehat{\phi(\mathbf{x}_i)} = \sum_{j=1}^{p} \alpha_{p-j+1} \phi(\mathbf{x}_{i-j})$$

which can be rewritten as

$$\widehat{\phi(\mathbf{x}_i)} = \begin{bmatrix} \phi(\mathbf{x}_1) & \phi(\mathbf{x}_2) & \cdots & \phi(\mathbf{x}_l) \end{bmatrix} \begin{bmatrix} \mathbf{0}_{(i-p-1) \times p} \\ \mathbf{I}_p \\ \mathbf{0}_{(l-i+1) \times p} \end{bmatrix} \boldsymbol{\alpha}$$

where $\mathbf{0}_{m \times n}$ is an $m \times n$ matrix with all entries equal to 0 and $\mathbf{I}_m$ is an $m \times m$ identity matrix. Using $\mathbf{J}_i$ for the second matrix in the above equation we have

$$\widehat{\phi(\mathbf{x}_i)} = \mathbf{X} \mathbf{J}_i \boldsymbol{\alpha}$$

Note that the matrix $\mathbf{X}$ is of dimension $N \times l$ where N is the dimension of the feature space which is typically very high. It is infeasible to compute the map $\phi(.)$ and hence this matrix is inaccessible. Fortunately, $\boldsymbol{\alpha}$ can be estimated without the knowledge of $\mathbf{X}$ as shown below. The prediction error between the real and predicted samples in the feature space can be written as

$$\xi(\boldsymbol{\alpha}) = \sum_{i=p+1}^{l} \|\phi(\mathbf{x}_i) - \widehat{\phi(\mathbf{x}_i)}\|^2$$

Noting that $\|\mathbf{x}\|^2 = \langle \mathbf{x}, \mathbf{x} \rangle = \mathbf{x}^t \mathbf{x}$, substituting the expression for the predicted sample and using the bilinearity of inner product we have

$$\xi(\boldsymbol{\alpha}) = \sum_{i=p+1}^{l} \Big(\phi^t(\mathbf{x}_i) \phi(\mathbf{x}_i) - 2 \phi^t(\mathbf{x}_i) \mathbf{X} \mathbf{J}_i \boldsymbol{\alpha} + \boldsymbol{\alpha}^t \mathbf{J}_i{}^t \mathbf{X}^t \mathbf{X} \mathbf{J}_i \boldsymbol{\alpha} \Big) \tag{2}$$

Setting $\frac{\partial \xi}{\partial \boldsymbol{\alpha}} = 0$ and solving for $\boldsymbol{\alpha}$ we have

$$\boldsymbol{\alpha} = \left(\sum_{i=p+1}^{l} \mathbf{J}_i{}^t \mathbf{X}^t \mathbf{X} \mathbf{J}_i \right)^{-1} \left(\sum_{i=p+1}^{l} \mathbf{J}_i{}^t \mathbf{X}^t \phi(\mathbf{x}_i) \right)$$

Note that $\mathbf{X}^t \mathbf{X}$ is precisely the Kernel matrix $\mathbf{K}$ defined by $\mathbf{K}_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$ which is of dimension $l \times l$ independent of N. The vector $\mathbf{X}^t \phi(\mathbf{x}_i)$ is the ith column $\mathbf{K}^i$of the Kernel matrix. Substituting these in to the equation

$$\boldsymbol{\alpha} = \left(\sum_{i=p+1}^{l} \mathbf{J}_i{}^t \mathbf{K} \mathbf{J}_i \right)^{-1} \left(\sum_{i=p+1}^{l} \mathbf{J}_i{}^t \mathbf{K}^i \right)$$

It can be easily seen that the final computation requires only the Kernel Matrix $\mathbf{K}$ which can be computed using the kernel function κ. Thus the model-parameter estimation can be done efficiently irrespective of the dimensionality of the feature space. The zero mean assumption can be handled by centering the kernel matrix in the feature space : $\mathbf{K_{ij}} = \langle \phi(\mathbf{x_i}) -$

$\frac{1}{l}\sum_{k=1}^{l}\phi(\mathbf{x_k}), \phi(\mathbf{x_j}) - \frac{1}{l}\sum_{k=1}^{l}\phi(\mathbf{x_k})\rangle$. This can be done by modifying the original kernel matrix as

$$\mathbf{K} = \mathbf{K} - \frac{1}{l}\mathbf{1K} - \frac{1}{l}\mathbf{K1} + \frac{1}{l^2}\mathbf{1K1}$$

where $\mathbf{1}$ is an $l \times l$ matrix with all entries equal to one. Note that the residual can also be measured by using the kernel matrix alone.

Note that the matrix $\sum_{i=p+1}^{l}\mathbf{J}_i{}^t\mathbf{K}\mathbf{J}_i$ is a sum of kernel matrices and is positive semi-definite. It is not guaranteed to be invertible always. In such cases numerical techniques like adding an additional term $\lambda\mathbf{I}_p$ can be used. This is equivalent to adding a penalty term controlling the norm of $\boldsymbol{\alpha}$. Algorithm 1 summarizes the entire procedure. Since the order of the model is typically much smaller compared to the number of samples, the algorithm runs in time $O(l^2c)$ where c is the cost of evaluation the kernel function on a pair of data points. An interesting observation is that the complete $l \times l$ kernel matrix is accessed only during the centering operation. For the remaining computation only a $p \times p$ kernel (sub-) matrix moving along the diagonal of the full kernel matrix is accessed. This can be exploited in improving the computational complexity of the algorithm.

Algorithm 1 Kernel AR model

```
Input : $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_l\}, p, \kappa(.,.)$
Output : $\boldsymbol{\alpha}$ , $\xi$
Compute the Kernel Matrix , $\mathbf{K}_{ij} \leftarrow \kappa(\mathbf{x}_i, \mathbf{x}_j)$
$\mathbf{K} \leftarrow \mathbf{K} - \frac{1}{l}\mathbf{1K} - \frac{1}{l}\mathbf{K1} + \frac{1}{l^2}\mathbf{1K1}$
$\mathbf{B} \leftarrow \mathbf{0}_{p\times p}$
$\mathbf{y} \leftarrow \mathbf{0}_{p\times 1}$
for $i = p+1 \cdots l$ do
  $\mathbf{B} \leftarrow \mathbf{B} + \mathbf{K}(i-p : i-1, i-p : i-1)$
  $\mathbf{y} \leftarrow \mathbf{y} + \mathbf{K}(i-p : i-1, i)$
end for
$\boldsymbol{\alpha} \leftarrow (\mathbf{B} + \lambda\mathbf{I}_p)^{-1}\mathbf{y}$
$\xi \leftarrow 0$
for $i = p+1 \cdots l$ do
  $\xi \leftarrow \xi + \mathbf{K}_{ii} - 2 * \mathbf{K}(i-p : i-1, i)^t\boldsymbol{\alpha}$
  $+\boldsymbol{\alpha}^t\mathbf{K}(i-p : i-1, i-p : i-1)\boldsymbol{\alpha}$
end for
return $\boldsymbol{\alpha}$ , $\xi$
```

We have shown that autoregressive modeling of data in a transformed space can be done efficiently using the kernel trick. A drawback of using the kernel trick is the lack of explicit control of the transformed space and inaccessibility to the samples in it. If the type of the nonlinearity is known apriori, an appropriate kernel can be employed for obtaining the accurate model. When there is no information about the type of nonlinearity, selection of kernel becomes more of an empirical procedure. While selection of kernel functions appropriate for a given task is still an open problem, empirical methods are shown to be effective for the selection kernels. The model parameters estimated using the proposed method can be used as a representation of the sequence for use with other algorithms for classification and recognition.

The algorithm presented uses the efficiency of the linear model with the nonlinearity introduced by the kernel trick to give an efficient nonlinear modeling scheme. Interpretation of AR modeling in the feature space is rather difficult as the nonlinear map involved $\phi(.)$ can distort the spatial ordering and continuity in which case application of an AR model may not have justification. This problem is more pronounced in the multidimensional case where the geometry of signal in the feature space can be quite complex. Despite this apparent lack of physical justification the efficiency and simplicity of the method make it very promising for nonlinear modeling.

4. Experiments, Results and Discussions

The kernel version of the AR modeling scheme proposed in the previous section, models a signal in the feature space. However, the predicted signal can not always be mapped to the input space. This is because the points $\phi(\mathbf{x}_i)$ are difficult to compute. Even if $\phi(.)$ is computationally feasible, an inverse mapping may not be defined for all the points. Thus it is difficult to test the goodness of the fit in the input space. However, when the input space is one dimensional, and kernel function is polynomial, i.e., $\kappa(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^t\mathbf{y})^d$, it is possible to compute and invert the nonlinear map $\phi(x) = x^d$. For the experimentation, we use 1-d signals synthesized using known and unknown generative models. We then model the signal with the proposed algorithm using polynomial kernels of various degrees. Some of the results are described and discussed in the rest of this section.

A signal is synthesized using the following generative model $x_n^7 = x_{n-1}^7 - 3x_{n-2}^7 + 3x_{n-3}^7$. It may be noted that this signal will have a linear structure in the feature space defined by the polynomial kernel of degree seven. We model the signal with polynomial kernels of varying degree. Performance of the modeling is analyzed with the help of prediction error, which is defined as the sum of squared errors between the predicted(modeled) and actual samples. This computation is done in the input space, so that the performance of all the kernels can be compared and analyzed in the same framework. Figure 1 shows the relative values of the average prediction error in the input space plotted against the degree of the kernel. Note that the computations are done only for the integer values of the kernel degree. Prediction error is computed as the mean value of the prediction error over 1000 trials and for different of the order of prediction.

The minimum error was obtained for kernel with degree seven. It may be noticed that the higher degree kernels perform much better the linear kernels. However, the error associated with various higher order kernels need

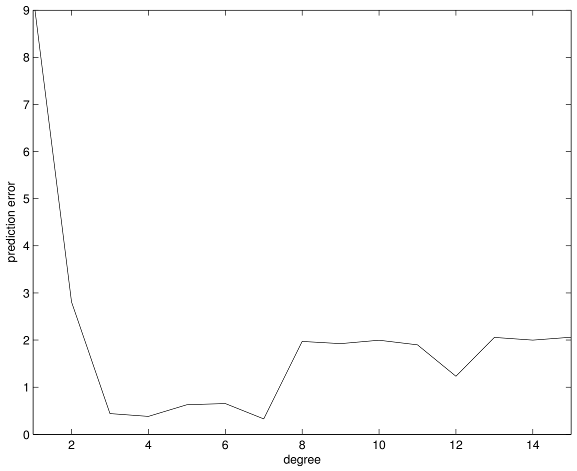

Figure 1: Prediction error in the input space on a signal generated by $x_n^7 = x_{n-1}^7 - 3x_{n-2}^7 + 3x_{n-3}^7$. Note the minima at $d = 7$.

not be monotonically decreasing. This observation highlights the fact that selection of appropriate kernel is possible for the accurate modeling. Also the experiment proves that the method does not overfit the data as in the case of many nonlinear modeling schemes. Similar results were obtained for many other generating models. The model parameters obtained in each case were the same as the parameters of the generating model . This experiment demonstrates that the algorithm performs as expected, and can model the signal with very high accuracy, provided the kernel can approximate the nonlinearity in the signal.

$\frac{SNR\rightarrow}{d\downarrow}$	**30dB**	**24dB**	**20dB**	**18dB**	**16dB**
1	0.68	1.53	2.79	4.22	6.06
2	0.78	1.64	2.90	4.30	6.11
3	0.98	1.86	3.06	4.45	6.20
4	1.16	2.04	3.15	4.50	6.12

Table 1: Table showing the percentage change in error as the noise percentage is varied for kernels of various degrees

The second experiment is performed to study the sensitivity of the algorithm to the presence of the noise in the input data. When noise is present in the data, the modeling scheme can perform inferior in the feature space. The sensitivity of the algorithm may depend on the map $\phi(.)$ and hence the kernel function. We conducted experiments with various signal to noise ratios of a synthetic signal and the percentage deterioration in the input space is studied. Table 4 depicts the percentage increase in the prediction error as the noise increases in the input space. It can be seen that the sensitivity of the (higher degree) kernel AR modeling scheme is very similar to the conventional AR modeling (i.e., linear kernels) procedure. As the noise increases, all kernels monotonically deteriorate in performance. As the noise increases, very often the linear methods break faster compared to the proposed kernel scheme, with higher order polynomial kernels.

The third experiment was done to test the effectiveness of the algorithm on signals whose generative model is unknown. Once again it was observed that the kernels of higher degree model the signal better. The proposed algorithm was also compared with nonlinear autoregressive modeling methods like the Nadaraya-Watson estimate and Local linear estimate. Preliminary results show that the method proposed here performs comparatively and is more stable since it does not require any iterative optimization. We are exploring the applicability of this method to handwritten character recognition and the preliminary results are superior to those obtained using the conventional AR modeling schemes.

5. Conclusions and Future Work

We presented a novel method for modeling nonlinear relations in time series by application of a linear model in a kernel-defined feature space. Preliminary experiments on synthetic one dimensional signals show promising results and we believe the method can be used as a feature extraction method for multidimensional sequence recognition tasks like video-event recognition and handwriting recognition to achieve better performance and robustness. We are conducting further experiments in this direction.

6. References

[1] Johana K Suykens and Joos Vandewalle, *Nonlinear Modeling*, Springer, 1998.

[2] John Shawe Taylor and Nello Christianni, *Kernel Methods for Pattern Analysis*, Cambridge University Press, 2004.

[3] George A. F. Seber and C. J. Wild, *Nonlinear Regression*, Wiley-IEEE, 2003.

[4] Ronald Christensen, *Advanced Linear Modeling*, Springer, 2001.

[5] Vladimir Naumovich Vapnik, *The Nature of Statistical Learning Theory*, Springer, 1999.

[6] Shantanu Chakrabartty, Yunbin Deng, and Gert Cauwenberghs, "Robust Speech Feature Extraction by Growth Transformation in Reproducing Kernel Hilbert Space," in *International Conference on Acoustics, Speech and Signal Processing*, 2004, vol. 1, pp. 133–136.