MODELLING CONDITIONAL PROBABILITY DISTRIBUTIONS FOR PERIODIC VARIABLES

I T Nabney, C M Bishop and C Legleye

Neural Computing Research Group, Aston University, UK.

ABSTRACT

Most of the common techniques for estimating conditional probability densities are inappropriate for applications involving periodic variables. In this paper we introduce two novel techniques for tackling such problems, and investigate their performance using synthetic data. We then apply these techniques to the problem of extracting the distribution of wind vector directions from radar scatterometer data gathered by a remote-sensing satellite.

1 INTRODUCTION

Many applications of neural networks can be formulated in terms of a multi-variate non-linear mapping from an input vector $\mathbf{x}$ to a target vector $\mathbf{t}$. A conventional neural network approach, based on least squares for example, leads to a network mapping which approximates the regression (i.e. average) of $\mathbf{t}$ given $\mathbf{x}$. For mappings which are multi-valued, however, this approach breaks down, since the average of two solutions is not necessarily a valid solution. This problem can be resolved by modelling a more complete description of the relationship between input and target, obtained by estimating the conditional probability density of $\mathbf{t}$ conditioned on $\mathbf{x}$, written as $p(\mathbf{t}|\mathbf{x})$. Although techniques exist for modelling such densities when the target vectors lie in Euclidean space, they are not appropriate when the targets are periodic. Direction and (calendar) time are two quantities that are periodic and which occur frequently in applications.

In this paper, we introduce two novel techniques for modelling conditional distributions for periodic variables, and investigate their performance using synthetic data. We then apply these techniques to the problem of determining the wind direction from radar scatterometer data gathered by the ERS-1 remote sensing satellite. The results are compared with a simple extension of a standard technique for periodic density estimation.

2 DENSITY ESTIMATION FOR PERIODIC VARIABLES

A commonly used technique for *unconditional* density estimation is based on mixture models of the form

$$p(\mathbf{t}) = \sum_{i=1}^{m} \alpha_i \phi_i(\mathbf{t}) \tag{1}$$

where α_i are called mixing coefficients, and the kernel functions $\phi_i(\mathbf{t})$ are typically chosen to be Gaussians. In order to turn this into a model for conditional density estimation, we simply make the mixing coefficients and any parameters in the kernels into functions of the input vector $\mathbf{x}$. This can be achieved by setting these parameters from the outputs of a neural network which takes $\mathbf{x}$ as input. This technique underlies the 'mixture of experts' model (Jacobs *et al.* (1)) and has also been considered by a number of other authors (White (2); Bishop (3); Lui (4)).

In this section we extend this technique to provide two distinct methods for modelling a conditional density $p(\theta|\mathbf{x})$ of a periodic variable θ. We also discuss a third, conventional technique which will be used to provide a benchmark for comparative results.

2.1 Transformation to an Extended Variable Domain

The first technique which we consider involves a transformation from a Euclidean variable $\chi \in (-\infty, \infty)$ to the periodic variable $\theta \in [0, 2\pi)$ of the form $\theta = \chi \bmod 2\pi$. This induces a transformation of density functions p with domain $\mathbb{R}$ to functions $\tilde{p}$ with domain $[0, 2\pi)$ as follows:

$$\tilde{p}(\theta|\mathbf{x}) = \sum_{N=-\infty}^{\infty} p(\theta + N2\pi|\mathbf{x}) \tag{2}$$

It is clear by construction that the function $\tilde{p}$ is a density function (since its integral is 1) and that it also satisfies the periodicity requirement $\tilde{p}(\theta + 2\pi|\mathbf{x}) = \tilde{p}(\theta|\mathbf{x})$.

Various choices for the kernel functions making up the mixture $p(\mathbf{t}|\mathbf{x})$ are possible, but in this paper we shall restrict attention to functions which are Gaussian of the form

$$\phi(\mathbf{t}|\mathbf{x}) = \frac{1}{(2\pi)^{c/2}\sigma_i(\mathbf{x})^c} \exp\left\{-\frac{\|\mathbf{t} - \mu_i(\mathbf{x})\|^2}{2\sigma_i(\mathbf{x})^2}\right\} \tag{3}$$

where $\mathbf{t} \in \mathbb{R}^c$. This formula assumes that the components of the output vector are statistically independent within each component of the destribution, and can be described by a common variance $\sigma_i(\mathbf{x})^2$.

'Artificial Neural Networks', 26-28 June 1995. Conference Publication No. 409. © IEE. 1995

This is not a serious restriction, since a mixture model with kernels as in equation (3) can approximate any density function to arbitrary accuracy with suitable choice of parameters (see McLachlan and Basford (5)). In any case, the target variables we consider in this paper are one-dimensional.

The density function $p(\chi|\mathbf{x})$ is modelled using a combination of a neural network and a mixture model as described above. In this paper we use a standard multi-layer perceptron with a single hidden layer of sigmoidal units and an output layer of linear units. In order to ensure that the mixture model in equation (1) is a density function, it is necessary that the mixing coefficients $\alpha_i(\mathbf{x})$ satisfy the constraints

$$\sum_{i=1}^{m} \alpha_i(\mathbf{x}) = 1, \tag{4}$$

$$0 \leq \alpha_i(\mathbf{x}) \leq 1. \tag{5}$$

These constraints are satisfied by choosing the $\alpha_i(\mathbf{x})$ to be related to a group of the network outputs by a 'softmax' function (1). The centres μ_i of the kernel functions are represented directly by the network outputs; this was motivated by the corresponding choice of an uninformative Bayesian prior, assuming that the relevant network outputs had uniform probability distributions (1). The standard deviations $\sigma_i(\mathbf{x})$ of the kernel functions represent *scale* parameters and so it is convenient to represent them in terms of the exponentials of the corresponding network outputs. This ensures that $\sigma_i(\mathbf{x}) > 0$ and discourages $\sigma_i(\mathbf{x})$ from tending to 0. It also corresponds to an uninformative prior in the Bayesian framework.

The error function E which is used is given by the negative logarithm of the likelihood function for the data with respect to the density function given by the network/mixture model combination. Derivatives of E with respect to the network weights can be computed using the rules of calculus (see (3)). These derivatives can then be used with standard optimisation procedures to find a minimum of the error function (which is a maximum likelihood solution). The results presented in this paper were obtained using a conjugate gradient algorithm. The maximum likelihood solution will be biased, but in our work the number of data points is large, so the bias will be small. The use of an exponential relationship between the kernel standard deviation and the network output also helps in this regard.

In a practical implementation, it is necessary to restrict the value of N in the summation. We have taken the summation over 7 complete periods of 2π spanning the range $(-7\pi, 7\pi)$. Since the component Gaussians have exponentially decaying tails, this represents a good approximation, provided that care is taken in initializing the network weights so that the initial kernels have their centres close to 0.

2.2 Mixtures of Circular Normal Densities

The second novel approach is also based on a mixture of kernel functions, but in this case the kernels themselves are periodic, thereby ensuring that the overall conditional density function is periodic. The particular form of kernel function which we use can be motivated by considering a velocity vector $\mathbf{v}$ in two-dimensional Euclidean space for which the probability distribution $p(\mathbf{v})$ is a symmetric Gaussian. By using the transformation $v_x = \|\mathbf{v}\| \cos\theta$, $v_y = \|\mathbf{v} \sin\theta\|$, we can show that the conditional distribution of the direction θ given the vector magnitude $\|\mathbf{v}\|$ is given by

$$p(\theta) = \frac{1}{2\pi I_0(m)} \exp\{m \cos(\theta - \theta_1)\} \tag{6}$$

which is known as a *circular normal* or *von Mises* distribution (Mardia (6)). The normalization coefficient is expressed in terms of the zeroth order modified Bessel function of the first kind, $I_0(m)$, and the parameter m (which depends on $\|\mathbf{v}\|$ in this derivation) is analogous to the inverse variance parameter in a conventional normal distribution. The parameter θ_1 corresponds to the peak of the density function.

With this choice of kernel function, we again use a neural network to determine the parameters $\alpha_i(\mathbf{x})$, $\theta_i(\mathbf{x})$ and $m_i(\mathbf{x})$ in a mixture model to generate a periodic conditional density function. The network weights are determined by minimizing the negative log likelihood of the training data. Because $I_0(m)$ is asymptotically an exponential function of m, some care must be taken in the implementation of the density function and its derivatives to avoid overflow in the results of intermediate calculations.

2.3 Soft Histograms

For the purposes of comparison, we also consider a simple extension of a conventional technique for conditional density estimation which can be applied directly to the problem of periodic variables. This involves a density model consisting of a fixed set of periodic kernel functions (given in this case by circular normal functions as in equation (6)), where the mixing proportions alone are determined by the outputs of a neural network and the centres θ_0 and variances m are fixed. We selected a uniform distribution of centres, and m equal for each kernel and sufficiently large to allow some degree of overlap, so that the network outputs determine the mixing coefficients only. The value for m was chosen to be 0.6 times the inter-centre distance. Of course, it could be made a parameter for optimization, but there was insufficient time to do this for the present paper. This approach is a 'soft' version of modelling the histogram of the target variable (i.e. banding the range and putting the actual values into frequency bins).

Clearly a major drawback of fixed kernel methods is that the number of kernels must grow exponentially with the dimensionality of the input space. However, for a single output variable, which occurs most commonly in applications, they can be regarded as practical techniques.

3 Application to Synthetic Data

In order to test and compare the methods introduced in section 2, we first consider a simple problem involving synthetic data, for which the true underlying distribution function is known. This data set is intended to mimic the central properties of the real data to be discussed in the next section. It is generated from a mixture of two triangular distributions where the centres and widths are taken to be linear functions of a single input variable x with fixed mixing coefficients 0.6 and 0.4. Any values which fall outside $(-\pi, \pi)$ are mapped back into this range by shifting in multiples of 2π to give a distribution which is periodic in the output variable θ.

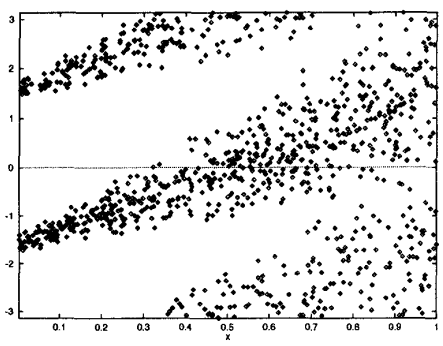

Figure 1: Scatter plot of the synthetic training data.

Three independent datasets (for training, validation and testing) were generated from this distribution, each containing 1000 data points. For each technique, training runs were carried out with the number of hidden units and kernels in the mixture model varied systematically to determine good values for them. 'Early stopping'[1] was used to reduce the effects of overfitting during training. Table 1 gives a summary of the best results obtained with each of the three methods.

A number of conclusions can be drawn from this series of experiments. As we might expect, the two adaptive mixture models achieved better results than those obtained with the fixed kernel functions. While the performance of the mixture models was poor

[1]The technique used was to to adjust the network parameters based on the error on the training set, and to monitor the error on the validation set. The final network was the one with the lowest *validation* error.

Method	Centres	Hidden Units	Validation Error	Test Error
1	2	13	1109.4	1128.7
1	6	7	1106.0	1130.2
2	1	14	1777.0	1808.3
2	2	1	1130.7	1147.0
2	2	3	1113.9	1132.9
2	6	8	1109.5	1133.9
2	10	6	1120.4	1136.5
3	36	3	1185.0	1205.4
3	36	7	1184.6	1223.5

Table 1: Results on synthetic data
Method 1: Transformation to an extended variable domain
Method 2: Mixture of adaptive circular normal functions
Method 3: Mixture of fixed kernel functions

when the number of kernel functions was too small (*i.e.* less than two in this case), the performance did not seriously degrade when the number of centres was greater than necessary. This is probably because, if there are more kernel functions than are needed to model the data, the network can either switch off redundant kernels by setting the corresponding mixing coefficients to small values or it can combine kernels by giving them similar centres and variances.

One problem that was noted in training the mixture models was that they were more likely to get stuck in local optima when the number of kernel functions was small. For example, with two circular normal kernels, six of the twelve training runs (with different numbers of hidden units) finished with a value for E in the region of 1450. By contrast, with 10 centres, just one of the twelve runs finished with E in that range. Thus it is important with this technique to use several starting positions for the weights; there was insufficient time to do this for the present paper.

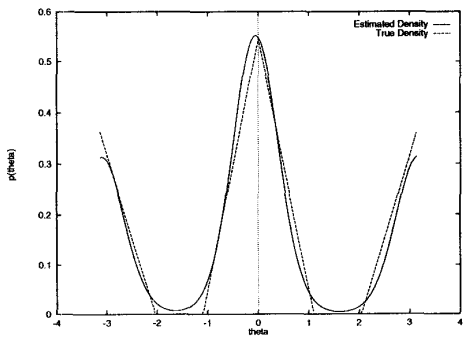

Figure 2: Distribution obtained from a mixture of adaptive circular normals with 3 hidden units and 2 centres. The input value x is 0.5.

The graph in figure 2 shows the match between the best adaptive circular normal network and the true generating distribution of the synthetic data at the input value $x = 0.5$. The graph in figure 3 shows the same information in a polar plot.

Other things being equal, we prefer the method using adaptive circular normal kernel functions, since there is a natural interpretation of the kernels in terms of the periodic target variable. A comparison of this technique with the soft histogram approach on real data is contained in the next section.

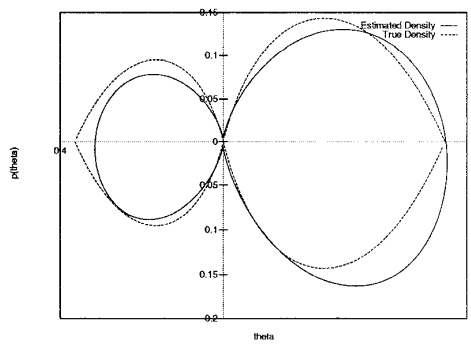

Figure 3: Polar plot of the distribution obtained from a mixture of adaptive circular normals with 3 hidden units and 2 centres. The input value x is 0.5.

4 Application to Radar Scatterometer Data

One of the original motivations for developing the techniques described in this paper was to provide an effective, principled approach to the analysis of radar scatterometer data from satellites such as the European Remote Sensing satellite ERS-1. The ERS-1 satellite is equipped with three C-band radar antennae which measure the total backscattered power (written as σ_0) along three directions relative to the satellite track, as shown in Figure 4. When the satellite passes over the ocean, the strengths of the backscattered signals are related to the surface ripples of the water (on a scale of a few centimetres) which in turn are determined by the low level winds.

Extraction of the wind vector from the radar signals represents an inverse problem which is typically multi-valued. Although determining the wind speed is relatively straightforward, the data gives rise to 'aliases' for the wind direction. For example, a wind direction of θ will give rise to similar radar signals to a wind direction of $\theta + \pi$, and there may be further aliases at other angles. A conventional neural network approach to this problem, based on a least squares estimate of the direction, would predict directions which were given by conditional averages. Since the average of several valid wind directions is not itself a valid solution, such an approach would clearly fail. In this paper we show how to extract the complete distribution of wind directions (conditioned on the three σ_0 values and incidence angle) and hence avoid such problems.

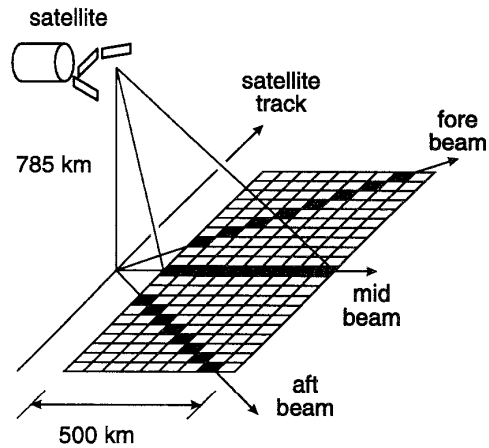

Figure 4: Schematic illustration of the ERS-1 satellite showing the footprints of the three radar scatterometers.

A large dataset of ERS-1 measurements, spanning a wide range of meteorological conditions, has been assembled by the European Space Agency in collaboration with the UK Meteorological Office. Labelling of the dataset was performed using wind vectors from the Met Office Numerical Weather Prediction model. These values were interpolated from the inherently coarse-grained model to regions coincident with the scatterometer cells. This was the best estimate of the wind vectors that was available in sufficient quantities for developing a neural network solution. It is suitable for predicting a background wind direction, but may be less appropriate in regions where there are smaller scale features, such as frontal zones and other areas of high gradients in wind speed or direction.

The data that was selected for the experiments reported in this paper was collected from low pressure (cyclonic) and high pressure (anti-cyclonic) circulations. These conditions, rather than cases that were homogeneous or with a simple gradient in speed or direction, were chosen to provide a more challenging task to test the modelling techniques. Ten windfields from each of the two categories were used: each windfield contains $19 \times 19 = 361$ cells each of which represents an area of approximately $50 \times 50\text{km}^2$. This gives a total of 7220 patterns, although the data for some of the cells was missing. When the data was split into three subsets, each contained 1963 patterns.

[2] Given that the total footprint width is 500 km, this implies that the cells overlap to some extent

We then trained both fixed and adaptive circular normal networks to model this data. The inputs used were the three values of σ_0 for the aft-beam, mid-beam and fore-beam, and the sine of the incidence angle of the mid-beam, since this angle strongly influences the reflected signal received by the scatterometer. The σ_0 inputs were scaled to have zero mean and unit variance, while the fourth input value was passed to the network unchanged. The target value was expressed as an angle clockwise from the satellite's forward path and converted to radians. Again, a conjugate gradient algorithm and 'early stopping' were used to train the networks.

Table 2 gives a summary of the preliminary results obtained with each of the three methods. As expected, the fact that this is a more complex domain than the synthetic problem meant that there were more difficulties with local optima. In fact, over 75% of the training runs ended with the network trapped in a local minimum of the error surface.

Method	Centres	Hidden Units	Validation Error	Test Error
1	2	20	2627.7	2689.5
1	4	15	2581.7	2641.4
1	8	12	2499.5	2718.0
2	36	24	2692.6	2784.5

Table 2: Results on satellite data
Method 1: Mixture of adaptive circular normal functions
Method 2: Mixture of fixed kernel functions

Table 2 shows that, although an adaptive-centre model with eight centres has the lowest error on the validation data, fewer centres are actually required to model the conditional density function well. This is also demonstrated by figure 5 which shows the conditional distribution of wind directions given by this network at a typical data point from the test set, and which is clearly bi-modal. This figure also shows how the peak of the density at the 'alias' direction is broader than that in the true direction at this data point.

This figure should be compared with figure 6 which shows the density at the same point given by the fixed kernel model; this distribution has four peaks. This is probably due to a sub-optimal choice of the variance parameter m; if this had been larger, it may have been possible to model the bi-modal distribution more accurately. Figure 7 shows the two distributions on the same polar plot, which also clearly shows the different alignments of the two distributions.

We conclude that the adaptive circular normal approach gives better results than the fixed kernel approach to this problem. At the test data point displayed, aliasing was found only at an angle of π, and this is true more generally.

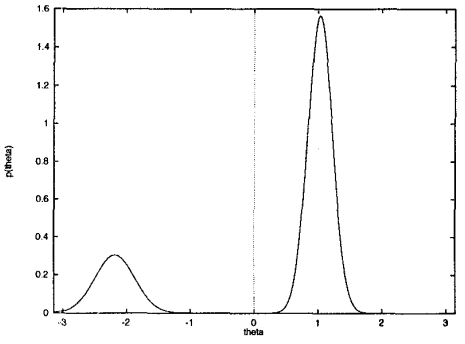

Figure 5: Plot of the distribution obtained from a mixture of adaptive circular normals with 12 hidden units and 8 centres at a test data point.

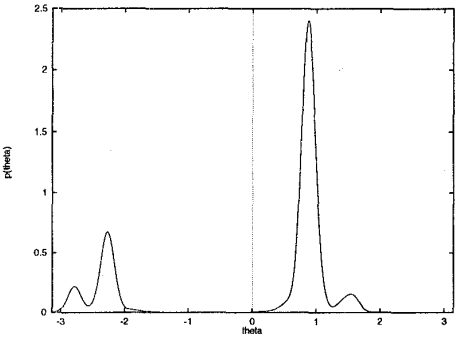

Figure 6: Plot of the distribution obtained from a mixture of fixed circular normals with 24 hidden units and 36 centres at a test data point.

An advantage of the density modelling approach is that it enables a better understanding of the direction ambiguities to be formed. In addition, it provides the most complete information for the next stage of processing, which is to 'de-alias' the wind directions by combining local information to determine the most probable overall wind field.

5 Discussion

In this paper we have introduced a new class of networks which can model conditional probability densities for periodic variables. We have illustrated the use of these networks in a simple problem involving synthetic data, and on radar scatterometer data. In both cases the networks outperformed the simpler fixed kernel approach to density modelling. A conventional network approach, involving the minimization of a sum-of-squares error function would have performed poorly on these problems since the required mapping is multi-valued.

One aspect of our approach that is more complex than conventional techniques is the problem of

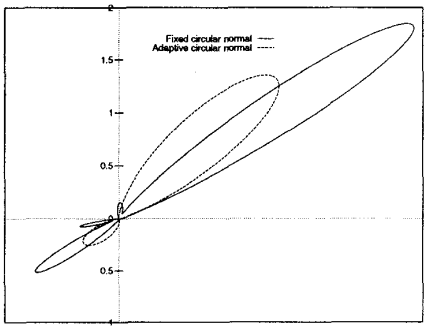

Figure 7: Polar plot of the distribution obtained from a mixture of 36 fixed kernels and a mixture of adaptive circular normals with 12 hidden units and 8 centres.

model order selection. The incorporation of a mixture model means that there are two structural parameters to select: the number of hidden units in the network and the number of kernels in the mixture model. Changing the number of kernels also changes the number of parameters in the mixture model, and hence the number of output units required in the network. Thus both structural parameters affect the number of adjustable parameters in the network. In this paper we used the simple approach of early stopping during training to limit the effective number of degrees of freedom in the network together with a systematic varying of the structural parameters. We found that the use of a greater number of kernel functions than was necessary did not degrade performance significantly. However, the number of experiments to determine good values for these parameters was large, and in our future research we intend to investigate techniques for the automation of model order selection.

6 Ackowledgements

We are grateful to the European Space Agency and the UK Meteorological Office for making available the ERS-1 data. We would also like to thank Iain Strachan and Ian Kirk of AEA Technology for a number of useful discussions relating to the interpretation of this data.

References

1. Jacobs R. A., Jordan M. I., Nowlan S. J., and Hinton G. E., 1991, "Adaptive mixtures of local experts", *Neural Computation*, **3**, 79–87.

2. White H., 1992, "Parametric statistical estimation with artificial neural networks", Technical report, University of San Diego, California.

3. Bishop C. M., 1994, "Mixture density networks", Technical Report NCRG/4288, Department of Computer Science, Aston University, U. K.

4. Lui Y., 1994, "Robust neural network parameter estimation and model selection for regression". In "Advances in Neural Information Processing Systems", number 6. Morgan Kaufmann.

5. McLachlan G. J. and Basford K. E., 1988, "Mixture models: Inference and Applications to Clustering", Marcel Dekker, New York.

6. Mardia K. V., 1972, "Statistics of Directional Data", Academic Press, London.