ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 251-256

Segmentation of Switching Dynamics with a Hidden Markov Model of Neural Prediction Experts

Alex Aussem and Corinne Boutevin
LIMOS (FRE CNRS 2239) - University Blaise Pascal, Clermont II,
B.P. 125, 63173 AUBIERE Cedex FRANCE - E-mail: alex@isima.fr

Abstract.
We discuss a framework for modeling the switching dynamics of a time series based on hidden Markov models (HMM) of prediction experts, here neural networks. Learning is treated as a maximum likelihood problem. In particular, we present an Expectation-Maximization (EM) algorithm for adjusting the expert parameters as well as the HMM transition probabilities. Based on this algorithm, we develop a heuristic that achieves a hard segmentation of the time series into distinct dynamical modes and the simultaneous specialization of the prediction experts on the segments. We present examples of the application of this algorithm to the segmentation of artificial and financial time series.

1 Introduction

In this paper, we consider a parametric dynamical system with discrete states to which are associated prediction experts. The state transitions are assumed to be governed by an homogeneous Markov chain, as discussed in Kohlmorgen et al. [6]. The architecture can be viewed as a particular case of input/output HMM (IOHMM) [1] in which the transitions are independent on the inputs and accomplish a hard partitioning of the input space.

The architecture is interpreted as a statistical model and learning is treated as a maximum likelihood problem; in particular we present a generalized EM (GEM) algorithm [4] for adjusting the expert parameters as well as the state transition probabilities, considering the internal states as missing data. In order to gradually achieve a soft-to-hard segmentation of the data and meanwhile to increase the robustness of the algorithm, we introduce a heuristic that adjusts on the fly the dispersion of the error distribution of the prediction experts. The model performance is first assessed on artificially generated chaotic data and then on a financial time series.

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 251-256

2 Prediction experts

In the following we assume that the reader is familiar with the basic principles of HMM. For a thorough introduction, we refer to the tutorial by Rabiner [7]. Our presentation closely derives from Kohlmorgen et al. [6]. Consider an HMM where each state $i = 1, \ldots, T$ is associated to a prediction expert. The prediction expert predicts the future value $y_t = x_{t+\tau}$ of a time series $\{x_t\}$ or some exogene variable, given a vector of past values $x_{t-d}^t = (x_{t-d}, \ldots, x_t)$ of arbitrary length. d is the embedding dimension and τ is the delay parameter. We assume that the target variables y_t, at each time t, is given by some deterministic function, $f_i(x_{t-d}^t)$, where i is current dynamic mode, with added noise ϵ_t, so that

$$y_t = f_i(x_{t-d}^t) + \epsilon_t \tag{1}$$

The errors ϵ_t are assumed to have a normal distribution with zero mean and a (unknown) standard deviation, σ, which does not depend on x_{t-d}^t or on t. Let $a = 1/2\sigma^2$, the conditional probability distribution of target variables for each expert is given by

$$p(y_t|x_{t-d}^t, i) = \sqrt{\frac{a}{\pi}} \mathrm{e}^{-a\,(y_t - f_i(x_{t-d}^t))^2} \tag{2}$$

Note that the parameter $a > 0$ will be adjusted in the course of training. The HMM state transition matrix $A = \{a_{ij}\}$ determines the probability to switch from state i to state j. This matrix A is kept constant in [6] to incorporate some prior knowledge about the rate the switching events occur. In this paper, we extend further the procedure by allowing the training procedure to adjust the transition rates when no prior knowledge is available.

3 Expert training

We seek to model the functions $f_i()$ by prediction experts. The training is performed by Generalized Expectation-Maximization (GEM) algorithm to help find the maximum likelihood estimator of the system. We have data, the time series $\{x_t\}$, which have been observed and data, the hidden states $\{s_t\}$ at each time t, which have not, and a vector of parameters θ. By making use the auxiliary EM function [3], it is easily shown that maximizing the likelihood for the *complete* data problem is equivalent to minimizing the cost

$$E = -\sum_{i=1}^{N} \sum_{t=d}^{T} \ln\left[p(y_t|x_{t-d}^t, s_t = i; \theta)\right] \gamma_t(i) - \sum_{j=1}^{N} \xi_t(i,j) \ln a_{ij} \tag{3}$$

where $\xi_t(i,j) = P(s_t = i, s_{t+1} = j | x_1^T, y_1^T; \theta^{old})$ and $\gamma_t(i) = \sum_j \xi_t(i,j)$. The M-step requires minimizing E with respect to the expert network parameters and the HMM transition probabilities. Generalized EM algorithm (GEM),

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 251-256

that simply produces a decrease in E. Examining Eq. 3, we see that the expert network parameters influence E only through the first term, and the HMM transition probabilities influence E only through the second term. Thus the M-step reduces to separate maximization problems representing each a particular maximum likelihood problem.

Transition probabilities - The optimal transition probabilities are readily obtained: the derivative of E with respect to the transition probabilities, a_{ij}, under the constraint $\sum_j a_{ij} = 1$ yields, by application of the Lagrange theorem, the following values $a^\star_{ij} = (\sum_{t=1}^{T-1} \xi_t(i,j))/(\sum_{t=1}^{T-1} \gamma_t(i))$.

Expert parameters - The M-step for the expert parameters reduces to a weighted least squares problem, that can be solved by any least-squares algorithm suited to the regression model employed. The derivative of the error with respect to the output of an expert f_k can drastically be simplified, $\partial E/\partial f_k = -2a \sum_{t=1}^{T} \gamma_t(k)(y_t - f_k(x_{t-d}^t))$.

4 Segmentation

Alternating the E and M steps yields a local maximum of the system likelihood. However the optimal likelihood principle yields a solution in which the parameters of an expert are influenced by the overall training set, and not only on the subset of the data assigned to that expert. This is undesirable. In order to achieve a hard segmentation and exclusively assign the data points to the experts, Kohlmorgen et al. introduce a deterministic annealing by means of a soft-max function over $\gamma_t(i)$ (i.e., $e^{\gamma_t(i)/\theta}/(\sum_{j=1}^{N} e^{\gamma_t(j)/\theta})$) where the "temperature" θ is gradually decreased during training. Unfortunately, the heuristic ends up with a solution that exclusively assigns the data points at time t to expert $argmax_j P(s_t = j|x_1^T, y_1^T)$, which is clearly sub-optimal as discussed in [7].

Instead we would like the $\gamma_t(i)$'s to converge towards the optimal state sequence obtained by the well-known Viterbi algorithm [5], (i.e., $argmax_{s_1,\ldots,s_T} P(s_1,\ldots,s_T|x_1^T, y_1^T)$). Therefore, we modify on the fly the parameter a in order to play on the error variance. Starting with small values of a allows the diversification of the experts: the matrix $\mathbf{A}$ being symmetric, no HMM state is favored and thus all $\gamma_t(i)$ are equals to $1/N$ regardless of the expert errors.

Deriving the error with respect to a, yields a unique optimal value $a^\star = 1/2H$ where $H = \frac{1}{T}\sum_{i=1}^{N}\sum_{t=d}^{T} \gamma_t(i)(y_t - f_i(x_{t-d}^t))^2$ is the model expected squared error and is therefore time dependent. As training progresses, H and thus $a^\star = 1$ converge to a limiting value. Also, the idea is to start with $a = 0$ to promote the diversification of the experts and then slowly increase a up to $a^\star$, and then freeze the system once $a^\star$ has converged and a stable segmentation is obtained by application of the Viterbi algorithm.

To summarize, the method is decomposed in two stages. The first stage is a standard application of the GEM training, except that a and the a_{ij}'s are gradually increased, at the end of each epoch, up to $a^\star$ and $a^\star_{ij}$. the optimal

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 251-256

values calculated at the current iteration. The first stage normally ends up with a stable segmentation. In the second stage, the Viterbi algorithm exclusively assigns the data to the experts and therefore promote the specialization of the experts.

5 Simulation results

The method is illustrated on artificial and real-world data. The same MLP is used in all experiments. The architecture consists of a tapped delay line of size 6, 8 hidden units and a linear output unit. For each setting we ran 10 trials with different seeds for the initial weights. To gauge performance of the HMM-based EM algorithm, the resulting MSE is compared to a reference MSE and a minimum MSE. The reference MSE is that obtained from a single neural net on the overall data set. The minimum MSE on artificial data is that obtained from neural nets pre-trained independently on their own dynamic until complete convergence. The latter can be viewed as the lowest possible value for the MSE we can expect from the algorithm, provided we know the true segmentation.

Mackey-Glass: For this first example, consider the Mackey-Glass equations, $dx(t)/dt = -0.1x(t) + 0.2x(t-\Delta)/(1+x(t-\Delta)^{10})$. A 4-*th* order Runge-Kutta technique was used to simulate the chaotic series. The stationary operating modes where established by using different delays $\Delta = 17$, $\Delta = 23$, and $\Delta = 30$ and a sampling rate of $f = 6$. 500 data were generated in total. Each mode covers exactly 100 data points. The lowest MSE obtained with a single NN is $MSE = 0.117$. The minimum MSE is $MSE = 0.048$. The HMM-based EM algorithm yields on average $MSE = 0.074$. Fig.1 shows a typical segmentation, along with the output of each NN. As may be observed, each NN achieves almost perfect forecasts when it is selected as expert, and give erroneous predictions outside its own dynamic. A typical error curve is shown in Fig. 2.

Hénon-Logistic maps: In this second example, the mapping underlying the time series randomly alternates between the Hénon bi-dimensional system $x(k+1) = 1.0 - 1.4 \cdot x^2(k) + 0.3 \cdot x(k-1)$ and the logistic map $x(k+1) = 4.0 \cdot x(k) \cdot (1 - x(k))$. 5000 data were generated. At each time step, a transition occurred at random with probability 1/100. Therefore, the mean segment length is 100. In order to have both series in the same y-range, the Hénon map was linearly rescaled into $(0, 1)$. As the task is more difficult, we were forced to fix the initial value for $a_{21} = a_{12}$ to 1/100, otherwise, the system did not converge properly. The MSE obtained with a single NN is $MSE = 0.018$. The minimum MSE obtained with 2 pre-trained neural nets is $MSE = 0.009$. The HMM-based EM algorithm yields on average $MSE = 0.011$. Results are not plotted for conciseness. Although the convergence is slower and sensitive to the initial value for the transition probabilities. Here again, each NN achieves almost perfect forecasts when it is selected as expert, and give erroneous predictions outside its own dynamic.

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 251-256

Real data: a stock option index made up from 4085 patterns was used. Of course, the minimum MSE could not be calculated as we have no idea what the optimal segmentation is. First the data was differentiated $y_t = (x_{t+1} - x_t)/x_t$ and afterwards rescaled in the range $(-1, +1)$. The number of neural nets was varied. We started with a single NN and obtained $MSE = 0.6$. The best performance is obtained using 3 experts, $MSE = 0.022$. A fourth expert did not prove useful. As may be seen in Fig. 2 on a subset of the data, the HMM-based EM algorithm did not find distinct dynamics. Instead, it has split the series according to the magnitude of the data so as to perfectly over fit the actual time series. While this not the result we expected, it demonstrates the model ability to perfectly overfit a large amount of data (4085) given a moderate number of adjustable parameters (151 in total).

6 Conclusion

We presented a new method for the unsupervised segmentation and identification of switching dynamics, based on a combination of a hidden Markov model and neural experts as discussed in [6]. The method was illustrated of artificial and real data. On-line prediction is left for future work.

References

[1] Y. Bengio, P. Frasconi, "An input-output HMM architecture", in *NIPS'94: Advances in Neural Information Processing Systems 7*, Morgan Kaufmann, 1995.

[2] L.E. Baum, T. Petrie, G. Soules, N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains", *Annals of Mathematical Statistics*, Vol. 41, pp. 164-171, 1970.

[3] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995.

[4] A. Dempster, N. Laird, D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", *J. Royal Stat. Soc. Series B*, **39**, pp. 1-38, 1977.

[5] G.D. Forney, "The Viterbi algorithm", *Proceedings of the IEEE*, Vol. 61, 1973.

[6] J. Kohlmorgen, S. Lemm, K.-R Müller, S. Liehr, K. Pawelzik, "Fast change point detection in switching dynamics using a hidden Markov model of prediction experts", *ICANN'99*, pp. 204-208, 1999.

[7] L.R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition", In *Proceeding of the IEEE*, Vol. 77, No. 2, pp. 257-285, 1989.

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 251-256

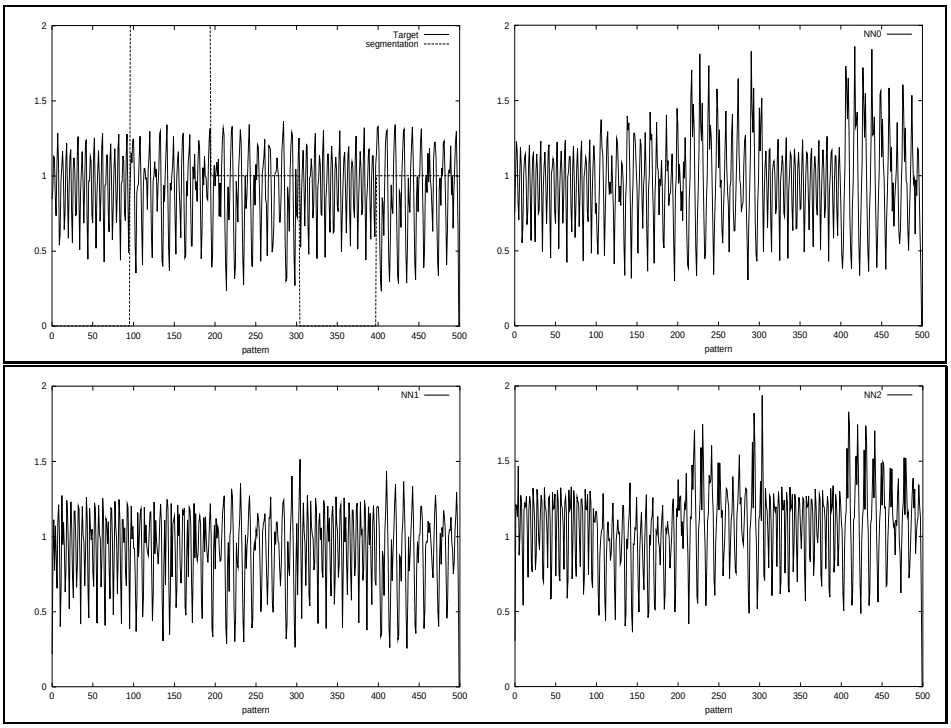

Figure 1: Mackey-Glass series. Upper left plot: true series and the segmentation obtained by the HMM-based EM algorithm. Other plots: the neural expert outputs.

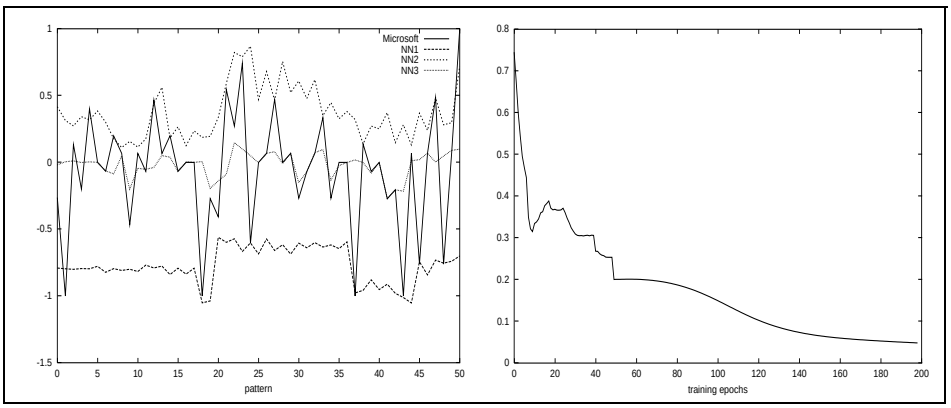

Figure 2: Left: stock option index in plain line, 3 neural experts in dotted line. Right: Typical error curve.