HIDDEN MARKOV MIXTURES OF EXPERTS FOR PREDICTION OF NON–STATIONARY DYNAMICS

Stefan Liehr, Klaus Pawelzik
University of Bremen, Institute of Theoretical Neurophysics,
Kufsteiner Str., D–28334 Bremen, Germany
email: {sliehr,klaus}@physik.uni-bremen.de

Jens Kohlmorgen, Steven Lemm, Klaus–Robert Müller
GMD FIRST, Rudower Chaussee 5, D–12489 Berlin, Germany
email: {jek,lemm,klaus}@first.gmd.de

Abstract. The prediction of non–stationary dynamical systems may be performed by identifying appropriate sub–dynamics and an early detection of mode changes. In this paper, we present a framework which unifies the mixtures of experts approach and a generalized hidden Markov model with an input–dependent transition matrix: the Hidden Markov Mixtures of Experts (HMME). The gating procedure incorporates state memory, information about the current location in phase space, and the previous prediction performance. The experts and the hidden Markov gating model are simultaneously trained by an EM algorithm that maximizes the likelihood during an annealing procedure. The HMME architecture allows for a fast on–line detection of mode changes: change points are detected as soon as the incoming input data stream contains sufficient information to indicate a change in the dynamics.

INTRODUCTION

Non–stationarity is a severe problem in classification and prediction of dynamical systems. A basic framework for dealing with non–stationarity is the mixtures of experts (ME) architecture, introduced by Jacobs et al. [3]. The mixtures of experts framework aims at separating the seemingly complex global behavior into a couple of lower dimensional sub–dynamics which can be modeled more easily.
For example, the Lorenz system [5] exhibits switching between two different oscillatory modes which are globally non–linear, while each single oscillation can be assumed to be approximately linear near the corresponding fix–point. In this case, two linear models would be the optimal choice in order to resolve the dynamical structure of the system. Then, the non–linearity can be

incorporated into the gating procedure. A central problem of using a set of experts is therefore the calculation of the activities of each expert — called the gating problem.
Many solutions have been proposed for dealing with the gating problem [1, 2, 3, 4, 7, 9, 10]. In its original formulation [3], the mixtures of experts method can be applied to systems, where different regimes do not overlap in phase space (i.e. the input space). The expert activities are provided by a feed–forward gating network given the current location in phase space [3, 10]. The use of a recurrent gating network [2] allows to distinguish also between overlapping regimes.
An alternative, non–recurrent approach to distinguish between overlapping regimes is the annealed competition of experts (ACE) method [7]. It has its roots in statistical mechanics and is a purely performance–driven concept, which considers a moving average prediction error for estimating the activities instead of using a gating network.
In contrast to these approaches, we use the concept of hidden Markov models (HMM) and associate each prediction expert with a hidden state of the system. Moreover, we introduce a non–linear gating network that models the conditional probabilities of transitions between the predictors depending on the actual location in phase space and the previous prediction performance. Hence, this approach unifies previous approaches by integrating (1) input information, (2) performance information, (3) state information for modeling the gating probabilities, *and* (4) strict consistency between training and prediction task. It is therefore also substantially more general than related HMM based methods [1, 9], which either do not make use of performance information in the gating process [1] or do not use input information in the gating process [9].
Simulation results show that mode changes can be detected earlier compared to other methods, if all the three types of information are incorporated into the gating process. Likewise, the prediction performance can be improved significantly.

THE HMME ALGORITHM

In the following, we assume that the reader is already familiar with the basic principles of hidden Markov models (HMMs). For a thorough introduction, we refer to the tutorial of Rabiner [8].
The Hidden Markov Mixtures of Experts (HMME) architecture consists of three information processing components, whose interactions are illustrated in Figure 1.

1. Experts

Consider a set of K models $\{f^k\}_{k=1}^K$, which will also be called experts or predictors. At time step t, $1 \leq t \leq T$, each expert provides a prediction $y_t^k =$

$f^k(\vec{x}_t, \vec{\Theta}^k)$, which might be e.g. the estimate of a future value $y_t = x_{t+\tau}$ of a time series $\{x_t\}$ given a vector of past values $\vec{x}_t = (x_t, x_{t-\tau}, \ldots, x_{t-(d-1)\tau})$. The parameter d is called the embedding dimension and τ is called the delay parameter. Note that the extension to multivariate time series is straightforward.
The parameter vector of each model is denoted by $\vec{\Theta}^k$, the combined parameter vector of the experts is $\vec{\Theta} = (\vec{\Theta}^1, \ldots, \vec{\Theta}^K)$. Under a Gaussian assumption, the probability density that a particular predictor k might have produced an observation y_t is given by

$$r_d(y_t|k) \quad = \quad \sqrt{\frac{\beta}{\pi}} \exp\left(-\beta \epsilon_t^k\right) \quad \text{with} \quad \epsilon_t^k = \left(y_t - y_t^k\right)^2 \tag{1}$$

The parameter β can be interpreted as an inverse–temperature and is used for deterministic annealing during the training process.
By using Bayes' theorem, the probability that expert k has generated a given observation y_t, can be written as

$$r_t^k = r(k|y_t) \quad = \quad \frac{\exp\left(-\beta \epsilon_t^k\right)}{\sum_{l=1}^{K} \exp\left(-\beta \epsilon_t^l\right)}. \tag{2}$$

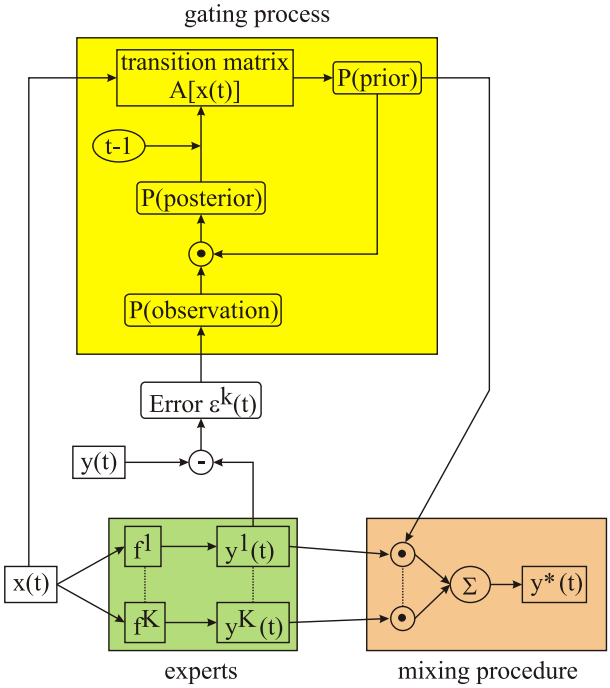

Figure 1: Architecture of the Hidden Markov Mixtures of Experts (HMME). The diagram shows the interactions of experts, hidden Markov gating, and mixing component. The different kinds of probabilities are denoted more intuitively by P(observation)= r_t^k, P(prior)= q_t^k and P(posterior)= p_t^k.

In eq. (2), we anneal the "temperature" $1/\beta$ to zero during training, which promotes the initial diversification of the experts and leads to an exclusive assignment of training data points to experts (hard segmentation). However, in order to obtain the best performance of the prediction system, we finally use the set of parameters $\vec{\Theta}$ at the temperature with the lowest prediction error.

2. Mixing

The joint prediction of the ensemble is given by a weighted sum of the individual outputs

$$y_t^* = \sum_{k=1}^{K} q_t^k y_t^k. \tag{3}$$

In figure 1 this is highlighted as the mixing procedure. The calculation of the mixing factors q_t^k is performed causally (eq. 5), i. e. no future information is used for calculating the estimate y_t^*. This is important for a strict consistency between the training and the application of the algorithm.
The probability distribution for observing y_t, using the entire prediction system, is given by

$$p_d(y_t|\vec{x}_t, \vec{\Theta}) \quad = \quad \sqrt{\frac{\beta}{\pi}} \exp\left(-\beta \left(y_t - y_t^*\right)^2\right) \tag{4}$$

3. Gating

The mixing factors q_t^k are also called the activations of each expert. They are calculated in the hidden Markov gating process. In order to understand how this calculation is performed, we have to consider an HMM, which consists of

a) a set $S = \{s^k\}$ of states, where each state is represented by a prediction expert f^k,

b) a matrix $A(\vec{x}_t) = \{a_t^{k|k'}\}$ of state transition probabilities, which, in our case, depend on the actual location $\vec{x}_t$ in the phase space,

c) an observation probability distribution $p(y_t|s^k) = r_d(y_t|k)$,

d) an initial state distribution $\pi = \{\pi^k\}$, which is assumed to be equally distributed.

The interesting point in this context is the input–dependent transition matrix $A(\vec{x}_t)$ and the way of processing the prior and posterior probabilities. The activation q_t^k is given by the *a priori* probability of being in state k at time t, which depends on the input–dependent transition probabilities $a_t^{k|k'}$ and

the *a posteriori* state probabilities p_{t-1}^k from the previous time step:

$$q_t^k = \sum_{k'=1}^{K} a_t^{k|k'} p_{t-1}^{k'} \quad \text{with} \quad p_t^k = \frac{r_t^k q_t^k}{\sum_{l=1}^{K} r_t^l q_t^l}. \tag{5}$$

In eq. (5), the posterior probabilities p_t^k are given, according to Bayes' theorem, by the prior probabilities q_t^k and the observation probabilities r_t^k, with an adequate normalization. Therefore, the posterior probabilities p_t^k contain information about the target value y_t, whereas the prior probabilities do not. The columns of the transition matrix and the probabilities sum to one, which is a requirement for the deduction of the learning rule:

$$\sum_k a_t^{k|k'} = \sum_k q_t^k = \sum_k p_t^k = \sum_k r_t^k \quad = \quad 1 \tag{6}$$

The transition matrix is represented by an input–dependent gating network[1] h with a $(K \times K)$–output matrix and a parameter vector $\vec{\Theta}^g$,

$$a_t^{k|k'} = h^{k|k'}(\vec{x}_t, \vec{\Theta}^g).$$

Training is performed by a gradient descent on the expected likelihood L. The log–likelihood can be decomposed into the free energy F, which provides a quality measure of the expert system, and the Kullback–Leibler "distance" KL, which yields to the same adaptation rule for the gating network as the second term of $\log L$:

$$\log L \quad = \quad \frac{1}{T} \sum_{t=1}^{T} \left(\log p_d(y_t|\vec{x}_t, \vec{\Theta}) + \sum_{k,k'} \xi_t^{kk'} \log a_t^{k|k'} \right) \tag{7}$$

$$F = -\frac{1}{\beta T} \sum_{t=1}^{T} \log p_d(y_t|\vec{x}_t, \vec{\Theta}) \quad \text{and} \quad KL = \frac{1}{T} \sum_{t=1}^{T} \sum_{k,k'} \xi_t^{kk'} \log \frac{\xi_t^{kk'}}{a_t^{k|k'}} \tag{8}$$

By minimizing KL the gating network learns to predict the joint probabilities $\xi_t^{kk'}$, which are calculated in accordance with the theory of hidden Markov models using the forward and backward probabilities α_t^k and β_t^k:

$$\xi_t^{kk'} \quad = \quad \frac{\beta_t^k r_t^k a_t^{kk'} \alpha_{t-1}^{k'}}{\sum_{ll'} \beta_t^l r_t^l a_t^{ll'} \alpha_{t-1}^{l'}} \tag{9}$$

In order to calculate the gradients of eq. (8), we use the method of Lagrange multipliers for incorporating the normalization conditions in eq. (6). The training can efficiently be performed by Expectation–Maximization (EM). The E–step consists of estimating the probabilities, the M–step adapts the models by minimizing the objective functions using gradient descent. Since in the M–step the probabilities are considered to be constant, the derivatives of the objective functions can be simplified drastically.

[1] We use a radial basis function (RBF) network of the Moody–Darken type [6] (look there for architecture and initialization) for the gating network.

EXPERIMENTAL RESULTS

We applied the HMME to different well–known systems, the deterministically switching logistic map and the Lorenz–System [5]. As shown in table 1, in both cases the HMME yields significantly better predictions than the ACE method.

Deterministically switching logistic map

The first example consists of a switching system of a noisy logistic map, $y_{t+1} = x_{t+1} = 4x_t(1 - x_t) + \eta_t$, and its "inverse", $y^I_{t+1} = x^I_{t+1} = 1 - 4x^I_t(1 - x^I_t) + \eta_t$, with uniform noise $\eta_t \in [-0.01, 0.01]$. The dynamics jumps from one mode to the other whenever $x_t \in [0.45, 0.55]$ holds. This is a system which exhibits non–linear behaviour, totally overlapping input spaces and a transition probability depending on the location in phase space. Additionally, the mean length of a mode is only about 16 time steps and therefore relatively short compared to previous applications of gated prediction systems. Modeling the jump processes is therefore very important for obtaining a high prediction quality. For the experts we use two radial basis function (RBF) networks of Moody–Darken type [6] with 6 RBFs each, and another network of the same type but with 10 RBFs for the gating network.
In table 1 the performance of our method is shown in comparison with the ACE–algorithm [7], which uses a lowpass filter instead of modeling transition probabilities. The advantage of the HMM–based method is first a fast detection of change points, which is possible because it does not depend on a fixed smoothing algorithm. Second, the method allows a recursive iteration of the prediction system with the possibility of predicting self–driven mode changes. Both properties are shown in Fig. 2.

Lorenz System

The second example is the Lorenz system [5], which is given by a set of three coupled differential equations. With the chosen parameters the Lorenz

system	algorithm	training	test 1	test 2
switching logistic map	HMME	0.00609	0.0125	0.0110
	ACE	0.00452	0.404	0.537
		$\tau_{ac} = 3$	$\tau_c = 2$	
Lorenz system	HMME	0.0270	0.0282	0.0279
	ACE	0.0287	0.0397	0.0381
		$\tau_{ac} = 9$	$\tau_c = 5$	

Table 1: Comparison of normalized mean squared errors (NMSE). Note that the filter used in the annealed competition of experts algorithm (ACE) is a–causal on the training data (τ_{ac}), while it is causal on the test data (τ_c). The Lorenz system is calculated with the parameters $\sigma = 16$, $b = 4$ and $r = 45.92$.

system exhibits a switching behaviour between oscillations around two fix–points. The system is globally non–linear, with the strongest non–linearity near the switching area from one oscillatory wing to the other, while each single oscillation can be assumed to be approximately linear near the corresponding fix–point. Therefore, we choose two linear experts and a non–linear gating network for modeling the Lorenz system. The non–linearity is thus incorporated into the gating procedure. The input and output of the experts are given by the state vector (X, Y, Z).
As shown in table 1, our algorithm yields significantly better predictions than ACE. In contrast to the logistic map example, however, the relative improvement is not that large. This can be explained first by deviations from a pure linear oscillation around the fix–points and second by the more difficult segmentation task due to the strong non–linear and non–instantaneous

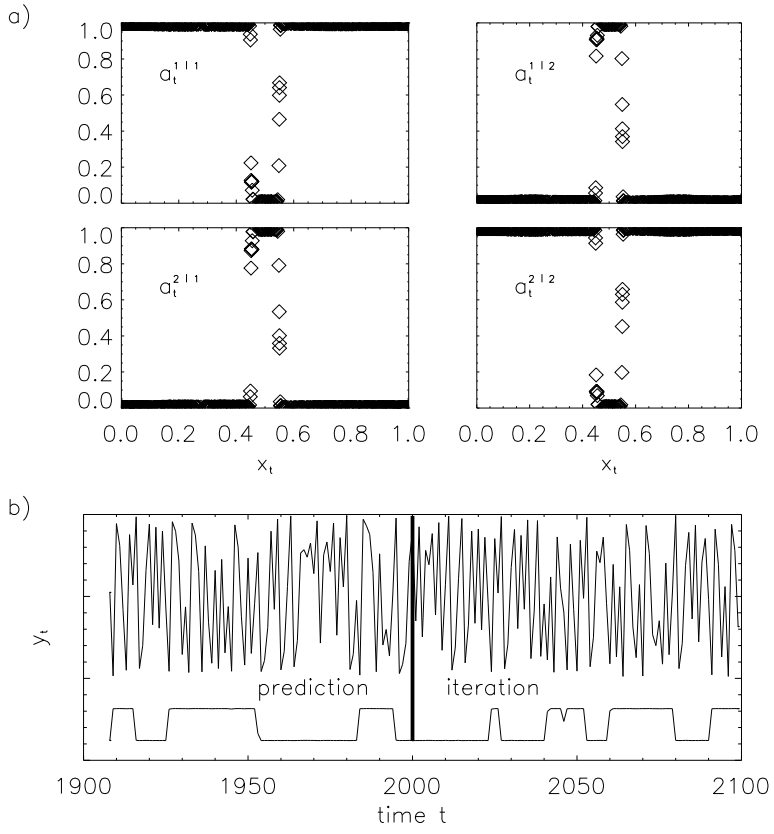

Figure 2: a) Input–dependent "elements" of the 2×2–transition matrix. The columns sum up to zero according to the normalization condition of eq. (6). b) Original time series of a validation dataset. The lower part shows the evolution of the activity of one expert during the one–step prediction task ($t < 2000$) and the iterated prediction task ($t > 2000$).

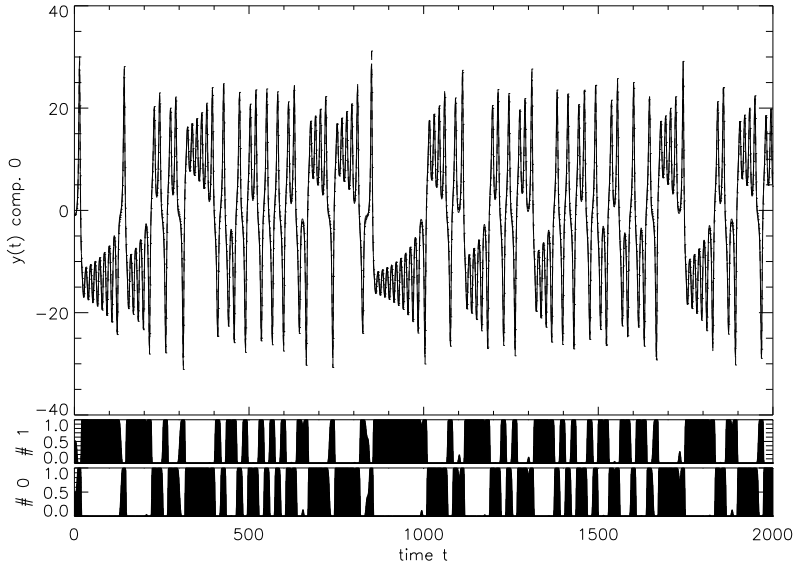

Figure 3: Segmentation of the dynamics of the Lorenz system during the one–step prediction task. The upper part shows the original time series of the X–coordinate. The prior probabilities of each expert is plotted in the lower two diagrams.

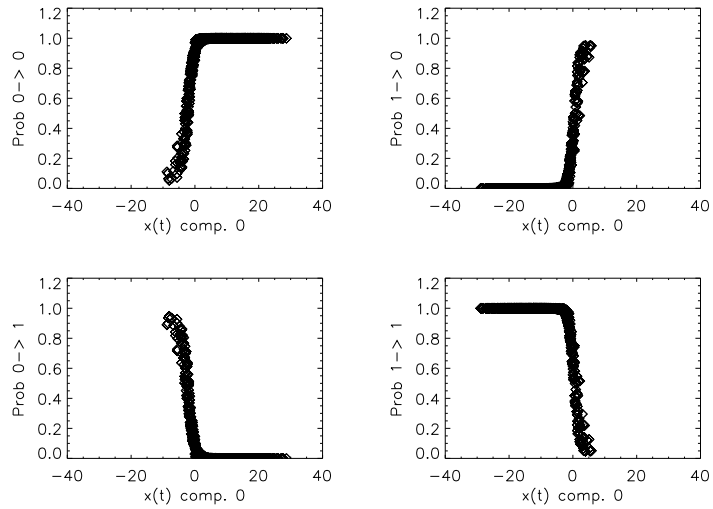

Figure 4: Representation of the input–dependent 2 × 2–output matrix of the gating network projected onto the plane (X/Prob). 'Prob' denotes the corrsponding transition probabilities.

switching process, as mentioned above.
Figure 3 demonstrates the segmentation of the dynamics. Obviously, each expert specializes on one oscillation. The algorithm follows even short–term mode changes. The time scale of the detection of change points can be seen in Figure 5. It clearly points out that the input–dependent gating procedure leads to an earlier detection of mode changes than the ACE algorithm. The final estimation of the input–dependent transition matrix projected onto the plane (X/Prob) between the X–coordinate of the state vector and the transistion probability is shown in Figure 4. It reflects the switching behavior of the Lorenz dynamics near $X = 0$.

CONCLUSION

We presented a generalized framework for unsupervised segmentation, identification, and prediction of switching dynamics. The architecture is composed of a mixture of experts and a hidden Markov model with an input–dependent state transition matrix. In contrast to existing methods, the system is trained in consistency with the prediction task and makes use of all available sources of information: input information from phase space, prediction error infor-

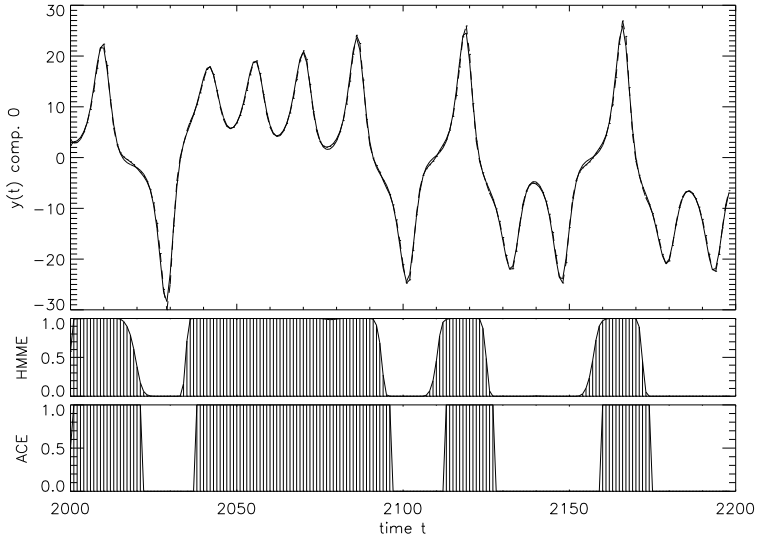

Figure 5: *Top:* Original dynamics and one–step prediction of the HMME–gated mixture of experts algorithm are almost identical. *Bottom:* The upper diagram shows the activation of one expert of the HMME–system, the lower diagram shows the activation of the corresponding expert of an expert system trained by the ACE–algorithm (without gating). The HMME detects the switching to the other wring much earlier and follows the drift process instantaneously. The non–gated expert system only performs a sudden switch, which is late compared to the real dynamical process.

mation, and HMM state information (memory). In particular, this allows for a fast detection of change points in on–line scenarios. Thereby, it can improve the prediction performance significantly. We expect that the method will be useful for the prediction of a wide range of natural signals, as e.g. climatologic or financial data.

Acknowledgement: We acknowledge support of the Deutsche Forschungsgemeinschaft (grants Pa569/2–1 and Ja379/51).

REFERENCES

[1] Y. Bengio and P. Frasconi, "An input output HMM architecture," in **Advances in Neural Information Processing Systems 7: NIPS 1994**, MIT Press, pp. 427–434, 1995.

[2] T. W. Cacciatore and S. J. Nowlan, "Mixtures of controllers for jump linear and non–linear plants," in **Advances in Neural Information Processing Systems 6: NIPS 1993**, MIT Press, pp. 719–726, 1994.

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

[4] A. Kehagias and V. Petridis, "Time series segmentation using predictive modular neural networks," **Neural Computation**, vol. 9, pp. 1691–1710, 1997.

[5] N. Lorenz, "Deterministic non–periodic flow," **J. Atmos. Sci.**, vol. 20, pp. 130, 1963.

[6] J. Moody and C. J. Darken, "Fast learning in networks of locally–tuned processing units," **Neural Computation**, vol. 1, pp. 281–294, 1989.

[7] K. Pawelzik, J. Kohlmorgen and K.-R. Müller, "Annealed competition of experts for a segmentation and classification of switching dynamics," **Neural Computation**, vol. 8, pp. 340–356, 1996.

[8] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," in A. Waibel and K. Lee, eds., **Readings in Speech Recognition**, Morgan Kaufmann, pp. 267–296, 1988.

[9] S. Shi and A. S. Weigend, "Taking time seriously: Hidden markov experts applied to financial engineering," **Proceedings of the IEEE/IAFE Conference on Computational Intelligence for Financial Engineering**, pp. 244–252, 1997.

[10] A. S. Weigend, M. Mangeas and A. N. Srivastava, "Non–linear gated experts for time series: discovering regimes and avoid overfitting," **International Journal of Neural Systems**, vol. 6, pp. 373–399, 1995.