

TIME SERIES CLUSTERING FOR NOVELTY DETECTION: AN EMPIRICAL APPROACH

LEONARDO AGUAYO*, GUILHERME A. BARRETO*

**Federal University of Ceará, Department of Teleinformatics Engineering
Av. Mister Hull, S/N, Center of Technology, Campus of Pici
CP 6007, CEP 60455-970 Fortaleza, Ceará, Brazil*

Emails: laguayo@deti.ufc.br, guilherme@deti.ufc.br

Abstract— This paper presents some results of DANTE project: **D**etection of **A**nomalies and **N**ovelties in **T**ime **s**ERies with self-organizing networks, whose goal is to devise and evaluate self-organizing models for detecting novelties or anomalies in univariate time series. The methodology to detect novelty consists in finding non-parametric confidence intervals, who are computed from the quantization errors obtained at the training phase, used at the testing phase as decision thresholds for classifying data samples as novelties/anomalies. We compared the performance achieved among variations of the self-organizing neural architectures using as input patterns a non-stationary data series composed by distinct dynamical regimes.

Keywords— Anomaly Detection, Time Series, Local Linear Models, Self-Organizing Maps.

Resumo— Este trabalho apresenta resultados do projeto DANTE: **D**etecção de **A**nomalias e **N**ovidades em séries **T**emporais. O objetivo do projeto é avaliar o desempenho de diversas redes auto-organizadas ao detectar anomalias/novidades em padrões de dados dinâmicos. A metodologia consiste em se determinar intervalos de confiança não-paramétricos a partir de erros de quantização obtidos na fase de treinamento, usados posteriormente na fase de teste como limiares de decisão para classificar amostras como sendo uma anomalia ou novidade. Realizou-se uma comparação do desempenho de variantes da rede SOM, utilizando-se como padrões de entrada uma série não-estacionária composta de regimes dinâmicos distintos.

Keywords— Detecção de Novidades, Séries Temporais, Modelos Lineares Locais, Mapas Auto-Organizáveis.

1 Introduction

Novelty detection¹ methods comprise computational procedures developed to handle the difficult problem of finding data samples which appears to be, in some sense, inconsistent or incompatible with a previous model for a data set. In recent years, it has been observed an increasing number of applications of the Self-Organizing Map (SOM) to such a problem (Sarasamma and Zhu, 2006; Barreto et al., 2005; Lee and Cho, 2005; Singh and Markou, 2004), most of them dealing with static data only, i.e. data for which the temporal dimension is an unimportant source of information.

However, several real-world applications provide data in a time-ordered fashion, usually in the form of successive measurements of the magnitude from one or several variables of interest, giving rise to time series data. In industry, for example, many process monitoring procedures involves measuring various sensor readings continuously in time to track the state of the monitored system (Zorriassatine et al., 2005; Jamsa-Jounela et al., 2003; Alhoniemi et al., 1999). In financial market, stock time series may present patterns that can guide an investor in his/her investment decisions in short or long-term horizons.

Anomaly detection in time series is particularly challenging due to the usual presence of

noise, inserted by the measurement device, as well of deterministic features - such as trend and seasonality - that can mask the character of novelty that may be present in data. Inherent non-stationary processes, such as regime-switching time series, also impose additional limitations on time series modeling. Furthermore, some time series, e.g. econometric ones, may have relatively few samples, restricting the amount of data available to extract information about its behavior. Finally, time-critical applications, such as fault detection and surveillance, requires on-line anomaly detection.

Traditional approaches, such as statistical parametric modeling and hypothesis testing (Markou and Singh, 2003a) can be successfully used to model static (i.e. memoryless) patterns, as these techniques assume some degree of stationarity of the data. On one hand, linear stationary dynamic processes can be handled by standard Box-Jenkins ARMA time series models. On the other hand, highly nonlinear and nonstationary dynamic patterns, such as chaotic or regime-switching time series, require a more powerful approach in terms of learning and computational capabilities.

At this point the use of artificial neural networks (ANNs) have shown to be useful due to their capability to act as general purpose nonlinear system identifier, generalizing the acquired knowledge to unknown data. Most of the ANN-based methods rely on supervised ANN models,

¹Depending on the research field, anomaly detection also comes under several designations, namely, anomaly detection, outlier detection, fault detection and condition monitoring.

such as MLP and RBF architectures (Markou and Singh, 2003b; Fancourt and Principe, 2004). However, a major drawback of such models in performing anomaly detection in time series is the asymmetry on the size of training data: labeled data for training may not be always unavailable or may be costly to collect. A plausible solution relies on the use of clustering algorithms to find subsets of data with similar temporal structure (Liao, 2005; Liao, 2007).

However, despite the recent interest in applying unsupervised learning on time series (B. Hammer, 2004; B. Hammer, 2005), few clustering-based algorithms for anomaly detection have been proposed to date. In particular, considering the usage of SOM algorithm as a clustering tool for anomaly detection systems, the former assertion is even stronger. Most of the SOM-based approaches usually converts the time series into a non-temporal representation (e.g. spectral features computed through Fourier transform) and use it as input to the usual SOM (Wong et al., 2006). Another common approach is to use fixed-length tapped delay lines at the input of the SOM, again converting the time series into a spatial representation (Fu et al., 2001).

Since the early 1990's, several temporal variants of the SOM algorithm have been proposed (see (Barreto and Araújo, 2001) for a review) with the aim of performing better than static clustering methods when dealing with time series data. However, to the best of our knowledge, such temporal SOMs have never been used for anomaly/novelty detection purposes.

From the exposed, the aim of this paper is to evaluate the performance of several self-organizing networks in the detection of anomalies/novelty in dynamic data patterns. For this purpose, we first describe clustering-based approaches which uses variations of the well-known SOM architecture, such as the Kangas' model (Kangas et al., 1990), TKM-Temporal Kohonen Map (Chappell and Taylor, 1993) and RSOM-Recursive SOM (Koskela et al., 1998). Additionally, the performance of the Fuzzy ART network (Carpenter et al., 1991) is also evaluated. All these algorithms were trained on-line and computer simulations carried out to compare their performances.

The remainder of the paper is divided as follows. In Section 2 we describe the self-organizing algorithms used in this work to perform anomaly/novelty detection in time series. In this section, we also present in detail the decision-support methodology used to run the simulations. In Section 4 the numerical results and comments on the performance of all the simulated algorithms are reported. The paper is concluded in Section 5.

2 Time Series Clustering for Anomaly Detection

In this section we describe self-organizing approaches adapted to perform anomaly detection in time series. In this paper we limit the scope of our description to prototype-based clustering algorithms. Here is assumed that the algorithms are trained on-line as the data is collected, and input vectors are built through a fixed-length window, sliding over the time series of interest. Thus, at time step t , the input vector is given by

$$\mathbf{x}^+(t) = [x(t) \ x(t-1) \ \dots \ x(t-p+1)]^T, \quad (1)$$

where $p \geq 1$ is the memory-depth parameter. Weight updating is allowed for a fixed number of steps, T_{max} . The first three algorithms to be described are based on the SOM algorithm, while the third one belongs to the family of ART (Adaptive Resonance Theory) architectures. Once the networks are trained, decision thresholds are computed based on the quantization errors for the SOM-based methods. ART-based models have an intrinsic novelty-detection mechanism, which can also be used for anomaly detection purposes.

2.1 Standard SOM

Usual SOM training is carried out using the vector $\mathbf{x}^+(t)$ as input. Thus, the winning neuron, $i^*(t)$, is given by

$$i^*(t) = \arg \min_{\forall i} \|\mathbf{x}^+(t) - \mathbf{w}_i(t)\|, \quad i = 1, \dots, Q, \quad (2)$$

where Q is the number of neurons and t denotes the current iteration of the algorithm. Accordingly, the weight vectors are updated by the following learning rule:

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + \eta(t)h(i^*, i; t)[\mathbf{x}^+(t) - \mathbf{w}_i(t)], \quad (3)$$

where $h(i^*, i; t)$ is a gaussian function which control the degree of change imposed to the weight vectors of those neurons in the neighborhood of the winning neuron:

$$h(i^*, i; t) = \exp\left(-\frac{\|\mathbf{r}_i(t) - \mathbf{r}_{i^*}(t)\|^2}{\sigma^2(t)}\right), \quad (4)$$

where $\sigma(t)$ defines the radius of the neighborhood function at iteration t , and $\mathbf{r}_i(t)$ and $\mathbf{r}_{i^*}(t)$ are the coordinates of neurons i and i^* in the output array, respectively. The learning rate, $0 < \eta(t) < 1$, should decay with time to guarantee convergence of the weight vectors to stable states. In this paper, we use $\eta(t) = \eta_0 (\eta_T/\eta_0)^{-(t/T_{max})}$, where η_0 is the initial value of η , and η_T is its final value after T_{max} training iterations. The variable $\sigma(t)$ should decay in time in a similar fashion.

2.2 Kangas' Model

Several SOM-based algorithms for time series clustering have been proposed but they have not been used for anomaly detection purposes yet. Kangas' model (Kangas et al., 1990) is one of the simplest temporal SOM algorithms available. The underlying idea of Kangas' model consists in performing a first-order IIR filtering on the input vector $\mathbf{x}^+(t)$ as follows:

$$\bar{\mathbf{x}}(t) = (1 - \lambda)\bar{\mathbf{x}}(t - 1) + \lambda\mathbf{x}^+(t), \quad (5)$$

where $0 < \lambda < 1$ is a memory decay -or memory depth- parameter. The filtered vector $\bar{\mathbf{x}}(t)$ is then presented to the SOM algorithm, which follows its usual training procedure.

2.3 TKM-Temporal Kohonen Map

The TKM model places the temporal memory at the output of the SOM network, by the maintenance of the activation $a_i(t)$ for each neuron:

$$a_i(t) = \lambda a_i(t - 1) - \frac{1}{2}\|\mathbf{x}(t) - \mathbf{w}_i(t)\|^2, \quad (6)$$

with $0 < \lambda < 1$ as the same memory depth defined in the Kanga's model. Now, the winner i^* is the one with the highest value for the activation, i.e, it satisfies

$$a_{i^*}(t) = \max_i \{a_i(t)\}, \quad i = 1, 2, \dots, Q, \quad (7)$$

and its weights are updated as in Eq. 3.

2.4 RSOM-Recurrent SOM

In this variant, the difference $\mathbf{x}'(t) = \mathbf{x}^+(t) - \mathbf{w}_i(t)$ contains the memory of the past status. Defining

$$\mathbf{y}_i(t) = \lambda \mathbf{x}'(t) + (1 - \lambda) \mathbf{y}_i(t - 1) \quad (8)$$

and the winner as

$$i^*(t) = \arg \min_i \{\mathbf{y}_i(t)\}, \quad i = 1, 2, \dots, Q, \quad (9)$$

the learning rule is now

$$\mathbf{w}_i(t + 1) = \mathbf{w}_i(t) + \alpha(t)h(i^*, i; t)\mathbf{y}_i(t), \quad (10)$$

where the memory is now taken into account when updating the weights of the winner. Note also that Eq. (8) has the same IIR structure of the Eq. 5.

2.5 The Fuzzy ART Algorithm

This paper also evaluates the performance of the Fuzzy ART algorithm (Carpenter et al., 1991) on anomaly detection in time series, due to its simplicity of implementation and low computational cost. The input vector $\mathbf{x}^+(t)$ is presented to a

competitive layer of Q neurons. The winning neuron i^* is selected if its *choice function* T_{i^*} is the highest one among all neurons:

$$i^*(t) = \arg \max_{\forall i} \{T_i(t)\}, \quad (11)$$

where the choice function T_i is computed as follows:

$$T_i(t) = \frac{|\mathbf{x}^+(t) \wedge \mathbf{w}_i(t)|}{\varepsilon + |\mathbf{w}_i(t)|}, \quad (12)$$

where $0 < \varepsilon \ll 1$ is a very small constant, and $|\mathbf{u}|$ denotes the L_1 -norm of the vector \mathbf{u} . The symbol \wedge denotes the component-wise minimum operator, i.e.

$$x_j^+(t) \wedge w_{ij}(t) \equiv \min \{x_j^+(t), w_{ij}(t)\}. \quad (13)$$

The next step involves a test for *resonance*. If

$$\frac{|\mathbf{x}^+(t) \wedge \mathbf{w}_{i^*}(t)|}{|\mathbf{x}^+(t)|} \geq \rho, \quad (14)$$

the weights of the winning neuron $i^*(t)$ are updated as follows:

$$\mathbf{w}_{i^*}(t + 1) = \beta (\mathbf{x}^+(t) \wedge \mathbf{w}_{i^*}(t)) + (1 - \beta) \mathbf{w}_{i^*}(t) \quad (15)$$

where the constants $0 < \rho < 1$ and $0 < \beta < 1$ are the *vigilance parameter* and the learning rate, respectively.

If the resonance test for the current winning neuron $i^*(t)$ fails, then another neuron is selected as the winner, usually the one with the second highest value for $T_i(t)$. If this neuron also fails, then the one with the third highest value for $T_i(t)$ is selected, and so on until one of the selected winning neurons $i^*(t)$ matches Eq. 14. If none of the existing prototype vectors resonates with the current input vector, then the input vector is declared *novel* and turned into a new prototype vector.

The parameter ρ controls the sensitivity of the Fuzzy ART algorithm to new input vectors. If $\rho \rightarrow 1$, more prototypes are created in the competitive layer, increasing the number of false alarms (false positives). If $\rho \rightarrow 0$, the number of prototypes decreases, increasing the number of missed detection (false negatives).

3 Detection Methodology

Unlike the Fuzzy ART algorithm, the SOM-based methods previously described do not have an intrinsic mechanism to detect novel or anomalous data. However, it has become common practice (Sarasamma and Zhu, 2006; Barreto et al., 2005; Alhoniemi et al., 1999) to use the quantization error

$$e_q(\mathbf{x}^+, \mathbf{w}_{i^*}; t) = \|\mathbf{x}^+(t) - \mathbf{w}_{i^*}(t)\|, \quad (16)$$

as a measure of the degree of proximity of $\mathbf{x}^+(t)$ to a statistical representation of normal behavior

encoded in the weight vectors of the SOM variants. Once the network has been trained, we present the training data vectors once again to this network. From the resulting quantization errors $\{e_q(\mathbf{x}^+, \mathbf{w}_{i^*}; t)\}_{t=1}^N$, computed for all training vectors, we compute decision thresholds for the anomaly detection tests. For a successfully trained network, the sample distribution of these quantization errors should reflect the 'known' or 'normal' behavior of the input variable whose time series model is being constructed.

Several procedures to compute decision thresholds have been developed in recent years, most of them based on well-established statistical techniques (Hodge and Austin, 2004), but we apply the method recently proposed in (Barreto et al., 2005). For a given significance level α , we are interested in an interval within which we can certainly find a percentage $100(1-\alpha)$ (e.g. $\alpha = 0.05$) of normal values of the quantization error. Hence, we compute the lower and upper limits of this interval as follows:

- **Lower Limit** (τ^-): This is the $100\frac{\alpha}{2}$ th percentile² of the distribution of quantization errors associated with the training data vectors.
- **Upper Limit** (τ^+): This is the $100(1-\frac{\alpha}{2})$ th percentile of the distribution of quantization errors associated with the training vectors.

Once the decision interval $[\tau^-, \tau^+]$ has been computed, any anomalous behavior of the time series can be detected on-line by means of the simple rule:

$$\begin{array}{ll} \text{IF} & e_q(\mathbf{x}^+, \mathbf{w}_{i^*}; t) \in [\tau^-, \tau^+] \\ \text{THEN} & \mathbf{x}^+(t) \text{ is } \mathbf{NORMAL} \\ \text{ELSE} & \mathbf{x}^+(t) \text{ is } \mathbf{ABNORMAL} \end{array} \quad (17)$$

4 Simulations

The feasibility of the proposed approach was evaluated using input signals derived from four different dynamic systems, and three of them were realizations of chaotic series. The first one is composed by the x component of Lorenz equations

$$\dot{x} = \sigma_L(y - x), \quad \dot{y} = x(\alpha_L - z) - y, \quad \dot{z} = xy - \epsilon_L z, \quad (18)$$

which exhibits chaotic dynamics for $\sigma_L = 10$, $\alpha_L = 28$ and $\epsilon_L = 8/3$. The second and third cases as well two different Mackey-Glass series, with distinct τ delays:

$$\dot{x} = Rx(t) + P \frac{x(t-\tau)}{(1+x(t-\tau))^{10}}, \quad (19)$$

²The percentile of a distribution of values is a number N_α such that a percentage $100(1-\alpha)$ of the sample values are less than or equal to N_α .

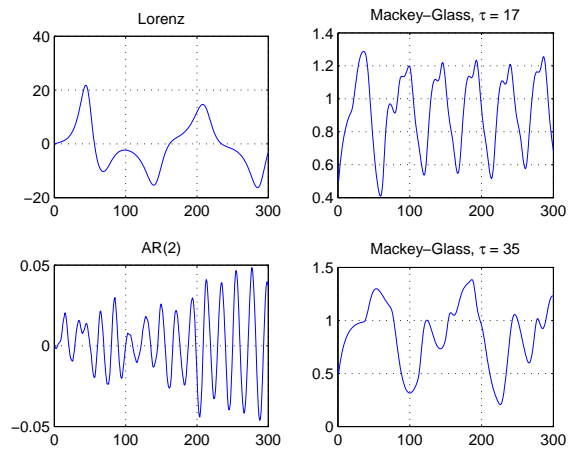


Figure 1: Samples of time series used in the simulations.

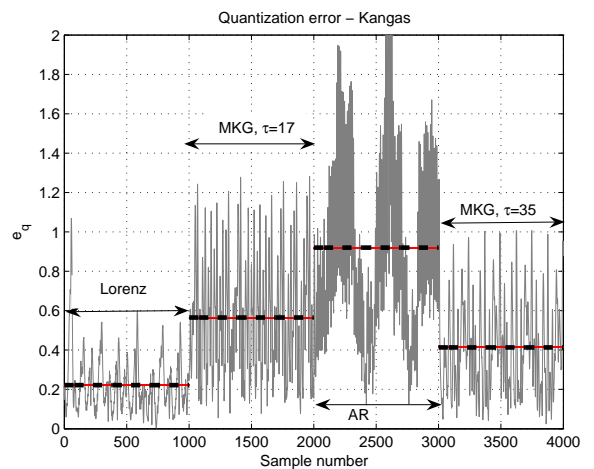


Figure 2: Quantization error $e_{i^*}(t)$. Each testing sequence has $k = 1000$ samples.

with $P = 0.2$, $R = -0.1$ and $\tau = 17$ or $\tau = 35$. The fourth case is an autoregressive process AR(2):

$$x(n+1) = 1.9x(n-1) - 0.99x(n-2) + n(t), \quad (20)$$

with $n(t)$ is a random sample from a gaussian white noise process with zero mean and variance $\sigma_n = 10^{-3}$. Figure 1 depicts 300 samples of each signal.

The novelty detection experiment was designed to perform the on-line detection of an anomalous signal, after training the networks with a sequence considered **NORMAL**. This role was assigned to the Lorenz series, leaving the Mackey-Glass and the AR process as representatives of an **ABNORMAL** behavior. Only for purposes of clarity on the presentation of results, all different testing sequences were presented sequentially: a set of k samples from each series was used as input to the four networks, followed by k samples of the next series.

Figure 2 shows the quantization error $e_{i^*}(t)$

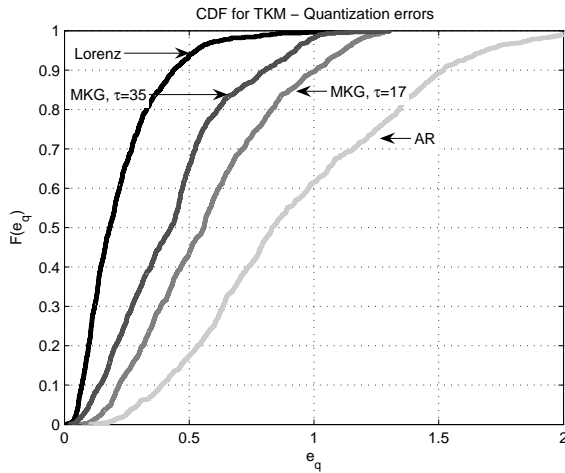


Figure 3: Cumulative distributions for quantization errors.

collected from the winning neuron i^* for the Kanga model, when the first $k = 1000$ samples of the training set consisted of samples generated by the Lorenz equations. It is possible to notice (i) the low prediction error for the first k samples, showing the feasibility of the proposed method described in Section 3; and (ii) that when a different pattern is presented, the quantization error is higher.

It is illustrative to observe the cumulative distribution function (CDF) of the quantization errors for the studied networks. Taken as example, Figure 3 depicts the CDFs for $e_{i^*}(t)$ obtained from all the different testing sequences for the TKM network, where it is possible to verify that **ABNORMAL** behavior results in distributions with higher variance. The same general behavior was observed for all the networks, but the SOM.

A comparative analysis of the performances of the presented models can be achieved straightforwardly using the percentages of true positive (TP) and false positive (FP) ratios. Here, a true positive is the correct detection of an **ABNORMAL** sample $x(t)$, when the testing signal belongs to novel pattern, and a false positive is the uncorrect novelty detection when a testing sample belongs to the training set (i.e. it has an already modeled dynamics).

The point with coordinates (FP, TP) is a point in *Receiver Operating Characteristic* (ROC) space, and can be used to visually identify good and bad classifiers. For instance, a perfect binary classifier should achieve the (0,1) point at ROC space. Now, if we change the percentile N_α , the decision interval $[\tau^-, \tau^+]$ at Eq. 17 is modified, and a set of points in ROC space can be derived, allowing to verify the performance of the classifiers under different degrees of tolerance for the prediction error.

Additionally, different configurations for the

input signals and network setup were analyzed, using variations on general parameters, such as the size of memory-depth p , noise variance σ_ε^2 , number of neurons Q , as well specific parameters, such as Fuzzy-ART ρ and β and memory decay λ . Figure 4 shows a typical result for the comparative performance among the networks, obtained with $Q = 20$ neurons and memory-depth parameter set to $p = 10$.

For the set of time series used in this paper, all SOM variations achieved similar performance, followed closely by Fuzzy-ART network. It worth notice that the memory structures introduced by Kanga's model, TKM and RSOM results in the great difference on the performance obtained, when compared to the standard SOM model, which has no internal structure to handle temporal series. Fuzzy-ART also performed closely to SOM variations, even with no explicit structure to process time-series. Its inherent novelty detection procedure, defined by a dissimilarity measure by Eq. (14) can explain the good performance. Finally, the proposed method appears as a feasible candidate to perform efficient novelty detection, using the methodology defined at the Section 3.

The optimal performance of a binary classifier may be used as a reference to choose the optimal thresholds for the interval $[\tau^-, \tau^+]$. Figure 5 shows the distance of each (fp,tp) point for each network to the best possible performance at the (0,1) coordinates. Defining the optimum point as the minimal distance between the mentioned points, the best threshold may be calculated off-line if the data sets for **ABNORMAL** and **NORMAL** samples are at disposal.

5 Conclusions

This paper introduced some results of the DANTE project, whose goal is to devise and evaluate self-organizing models for detecting novelties or anomalies in univariate time series. Non-parametric confidence intervals are computed from the quantization errors obtained at the training phase and used as decision thresholds for detecting novelties/anomalies. We compared the performance achieved among variations of the self-organizing neural architectures, as well the Fuzzy-ART performance, to the same problem. Future work on the subject investigates mechanisms of dynamically set the thresholds $[\tau^-, \tau^+]$, leading to self-adjustment of optimal performance.

Acknowledgments: The authors thank CAPES/PRODOC for its financial support.

Referências

Ahloniemi, E., Hollmén, J., Simula, O. and Vesanto, J. (1999). Process monitoring and modeling us-

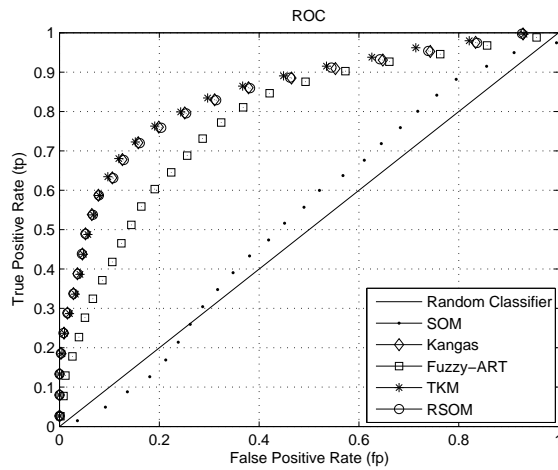


Figura 4: ROC curves for the SOM variations and Fuzzy-ART.

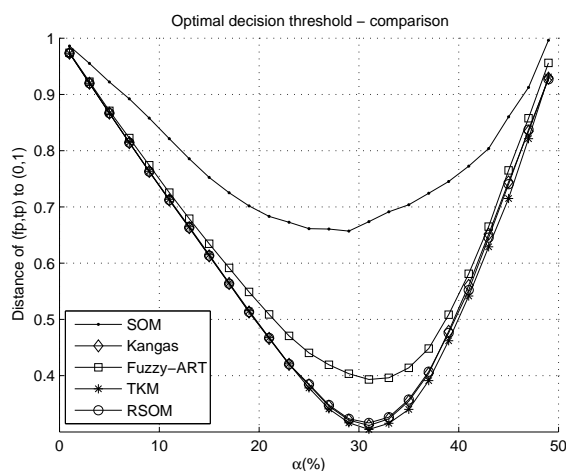


Figura 5: Optimal choice of N_α , using the distance of (fp, tp) point to (0,1).

ing the self-organizing map, *Integrated Computer Aided Engineering* **6**(1): 3–14.

- B. Hammer, A. Micheli, A. M. (2004). Recursive self-organizing network models, *Neural Networks* **17**: 1061–1086.
- B. Hammer, A. Micheli, N. A. M. (2005). Self organizing maps for time series, pp. 115–122.
- Barreto, G. A. and Araújo, A. F. R. (2001). Time in self-organizing maps: An overview of models, *International Journal of Computer Research* **10**(2): 139–179.
- Barreto, G. A., Mota, J. C. M., Souza, L. G. M., Frota, R. A. and Aguayo, L. (2005). Condition monitoring of 3G cellular networks through competitive neural models, *IEEE Transactions on Neural Networks* **16**(5): 1064–1075.
- Carpenter, G. A., Grossberg, S. and Rosen, D. B. (1991). Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system, *Neural Networks* **4**(6): 759–771.
- Chappell, G. J. and Taylor, J. G. (1993). The temporal Kohonen map, *Neural Networks* **6**(3): 441–445.

Fancourt, C. L. and Principe, J. C. (2004). On the use of neural networks in the generalized likelihood ratio test for detecting abrupt changes in signals, *Proceedings of the 2000 IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN'00)*, Vol. 3, pp. 243–248.

Fu, T. C., Chung, F. L., Ng, V. and Luk, R. (2001). Pattern discovery from stock time series using self-organizing maps, *Workshop Notes of KDD2001 Workshop on Temporal Data Mining*, pp. 27–37.

Hodge, V. J. and Austin, J. (2004). A survey of outlier detection methodologies, *Artificial Intelligence Review* **22**(2): 85–126.

Jamsa-Jounela, S. L., Vermasvuori, M., Enden, P. and Haavisto, S. (2003). A process monitoring system based on the kohonen self-organizing maps, *Control Engineering Practice* **11**(1): 83–92.

Kangas, J. A., Kohonen, T. K. and Laaksonen, J. (1990). Variants of self-organizing maps, *IEEE Transactions on Neural Networks* **1**(1): 93–99.

Koskela, T., Varsta, M., Heikkonen, J. and Kaski, K. (1998). Time series prediction using recurrent SOM with local linear models, *International Journal of Knowledge-based Intelligent Engineering Systems* **2**(1): 60–68.

Lee, H.-J. and Cho, S. (2005). SOM-based novelty detection using novel data, *Lecture Notes on Computer Science* **3578**: 359–366.

Liao, T. W. (2005). Clustering of time series data - a survey, *Pattern Recognition* **38**(11): 1857–1874.

Liao, T. W. (2007). A clustering procedure for exploratory mining of vector time series, *Pattern Recognition* **40**: 2250–2562.

Markou, M. and Singh, S. (2003a). Novelty detection: a review - part 1: Statistical approaches, *Signal Processing* **83**: 2481–2497.

Markou, M. and Singh, S. (2003b). Novelty detection: A review - part 2: Neural network based approaches, *Signal Processing* **83**: 2499–2521.

Sarasamma, S. T. and Zhu, Q. A. (2006). Min-max hyperellipsoidal clustering for anomaly detection in network security, *IEEE Transactions on Systems, Man and Cybernetics* **B-36**(4): 887–901.

Singh, S. and Markou, M. (2004). An approach to novelty detection applied to the classification of image regions, *IEEE Transactions on Knowledge and Data Engineering* **16**(4): 1041–1347.

Wong, M., Jack, L. and Nandi, A. (2006). Modified self-organising map for automated novelty detection applied to vibration signal monitoring, *Mechanical Systems and Signal Processing* **20**(3): 593–610.

Zorriassatine, F., Al-Habaibeh, A., Parkin, R., Jackson, M. and Coy, J. (2005). Novelty detection for practical pattern recognition in condition monitoring of multivariate processes: a case study, *International Journal of Advanced Manufacturing Technology* **25**(9-10): 954–963.