A Nonparametric Information Theoretic Approach for Change Detection in Time Series

Songlin Zhao and José C. Príncipe

Abstract— This paper presents an online nonparametric methodology based on the Kernel Least Mean Square (KLMS) algorithm and the surprise criterion, which is based on an information theoretic framework. Surprise quantifies the amount of information a datum contains given a known system state, and can be estimated online using Gaussian Process Theory. Based on this concept, we use the KLMS algorithm together with surprise criterion to detect regime change in nonstationary time series. We test the methodology on a synthesized chaotic time series to illustrate this criterion. The results show that surprise criterion is better than the conventional segmentation based on the error criterion.

I. INTRODUCTION

T HERE has been a recent surge of interest in tests for structure change problem in various areas, such as economics, audio processing, process control and sensor monitoring. Nowadays, common estimators include least squares (LS) [1], two stage least squares (2SLS) [2], maximum likelihood (ML) [3], likelihood ratio [4] and M-estimators [5]. All the estimators mentioned above use a linear or nonlinear function of the prediction error as a criterion to make the decision how to segment the time series.

In this paper, a new estimator of change detection is proposed based on the concept of surprise. Surprise is a concept that can be framed in statistical terms and is a subjective measure to estimate how a new datum affects an observer, in terms of differences between posterior and prior knowledge of a given world [6]. The definition of surprise was first proposed by Pfaffelhuber [7] as subjective information and later formalized by Palm [8]. More recently, Itti and Baldi [6] extended the concept to a Bayesian framework as the Kullback-Leibler (KL) divergence between the prior and posterior in a model class. Most recently, surprise was simplified to the negative log likelihood (NLL) of an observation given the learning machine's hypothesis [9], with the advantage that it can be estimated online using Gaussian Process Theory.

Even though the definition of surprise has been established, its application to time series segmentation and change detection is quite novel. Inspired by [9], this paper utilizes the NLL version of surprise in combination with the Kernel Least Mean Square (KLMS) algorithm [10] as an estimator to detect structure changes in data streams to search appropriate breakpoints. The KLMS finds a nonlinear optimal solution to the regression (filtering) problem by implementing a

Songlin Zhao, and José C. Príncipe are with the Department of Electrical and Computer Engineering, The University of Florida, Gainesville, USA (email: slzhao, principe@cnel.ufl.edu).

linear adaptive filter in a Reproducing Kernel Hilbert Space (RKHS). Since the optimal solution is obtained incrementally with a weighted combination of past values of the input using gradient descent learning, it is model free [11]. Assuming the time series $\{\mathbf{x}, f(\mathbf{x})\}$ is an observation of a Gaussian process, surprise is defined by a combination of the prediction error and the input variance, therefore, surprise contains more statistic information than estimators just utilizing prediction error and is consequently more informative.

In this paper, we assume that the time series is locally stationary and that there are abrupt changes amongst the locally stationary segments. We train KLMS algorithms independently for each segment, and assume that there is enough data for training. In a test set we compare the segmentation decisions based on the prediction error of the KLMS and the surprise using a likelihood ratio test [12]. This methodology can be easily extended to online training in a mixture model framework as done in [12].

The organization of this paper is as follows. In Section II, after a brief introduction of KLMS algorithm and surprise, a method to analysis data structure change based on KLMS training method and surprise criterion is interpreted. Afterwards, experiments for Lorenz chaotic time series are studied in Section III to support our method. Finally, Section IV summarizes the conclusions and future lines of work.

II. A NONPARAMETRIC INFORMATION THEORETIC METHOD

Suppose the goal is to analyze structure change of a scalar data $\{x_1, x_2, \ldots, x_i \ldots\}$ assumed locally stationary. For modeling, we train a predictor using the KLMS approach in one of the stationary segments. Once the filter is trained, let us call τ_n the state of the KLMS filter at time sample n.

A. Kernel Least-Mean-Square Algorithm

KLMS is a sample-by-sample update method for an adaptive filter in a reproducing kernel Hilbert space (RKHS). After mapping the input data \mathbf{u}_n into RKHS as $\varphi(\mathbf{u}_n)$, KLMS finds an appropriate weight Ω through stochastic gradient descent method [10], i.e.

$$\Omega_n = \eta \sum_{i=1}^{n-1} e_i \varphi(\mathbf{u}_i) \tag{1}$$

where η is the learning stepsize, e_i is prediction error at time sample *i* and φ is the mapping from input space to RKHS. Then by the kernel trick, the filter output can be computed

This work was partially supported by NSF grant ECCS 0856441.

efficiently,

$$y_n = \Omega_n^T \varphi(\mathbf{u}_n) = \eta \sum_{i=1}^{n-1} e_i \kappa(\mathbf{u}_i, \mathbf{u}_n)$$
(2)

where κ is a positive definite function also called kernel. KLMS algorithm is an online model free method, because the system output is just a linear combination of the past inputs with weighings given by the prediction error.

Furthermore, KLMS algorithm is a self-regularization method because a constraint on the solution norm $\|\Omega\|^2$ ensures well-posedness of this adaptive filter. The Tikhonov Regularization is a useful technique to prevent ill-posed problem in kernel adaptive filters [13]. It has been proved that utilizing the regularization term is equivalent to constraining the norm of the solution in LS problems [11]. Therefore, KLMS algorithm is well-posed in the sense of Hadamard, and does not need explicit regularization.

In conclusion, KLMS algorithm has many adavantages: simplicity, sequential learning approach, higher accuracy than linear filters, model-free, self-regularization and so on. This fact inspires us to utilize KLMS as the training approach.

B. A Brief Introduction of Surprise

Surprise is an information measure, which quantifies how much information a data contains given "a known model state". Defined as NLL of the conditional of the input data given the system state, surprise of a new data $\{\mathbf{u}_n, d_n\}$ can be written as

$$S_{\tau}(\mathbf{u}_n, d_n) = -\ln p(\mathbf{u}_n, d_n | \tau)$$
(3)

where \mathbf{u}_n is the input, d_n is corresponding desire, τ is the known model and $p(\mathbf{u}_n, d_n | \tau)$ is the posterior distribution of \mathbf{u}_n, d_n given the known model state. If $S_{\tau}(\mathbf{u}_n, d_n)$ is large, the new sample is "abnormal" for the current model. On the other hand, if $S_{\tau}(\mathbf{u}_n, d_n)$ is small, this new data is well expected by the current model. How to estimate $p(\mathbf{u}_n, d_n | \tau)$ is the key point. As a posterior distribution, $p(\mathbf{u}_n, d_n | \tau)$ can be written as

$$p(\mathbf{u}_n, d_n | \tau) = p(d_n | \mathbf{u}_n, \tau) p(\mathbf{u}_n | \tau)$$
(4)

In general, we assume the distribution of \mathbf{u}_n to be independent of the current model and if no a prior knowledge is available about the time series, we assign a uniform priori to $p(\mathbf{u}_n)$. That is,

$$p(\mathbf{u}_n, d_n | \tau) = p(d_n | \mathbf{u}_n, \tau) p(\mathbf{u}_n) \propto p(d_n | \mathbf{u}_n, \tau)$$
 (5)

Therefore, surprise defaults to $p(d_n | \mathbf{u}_n, \tau)$, i.e, the conditional probability of the new sample given the input data and the model state.

Gaussian Processes regression (GPR) is a good choice to estimate $p(d_n | \mathbf{u}_n, \tau)$ online. Assume the known model τ is learned by the data set $D = \{(\mathbf{u}_i, d_i) | i = 1, 2..., m\}$. As mentioned in [14], the prior distribution of system outputs follows a jointly Gaussian distribution,

$$[y(\mathbf{u}_1), \dots, y(\mathbf{u}_m)]^T \sim N(0, \tilde{\sigma}^2 \mathbf{I} + \mathbf{G}_m)$$
(6)

where \mathbf{G}_m is the covariance function.

$$\mathbf{G}_{m} = \begin{bmatrix} \kappa(\mathbf{u}_{1}, \mathbf{u}_{1}) & \dots & \kappa(\mathbf{u}_{m}, \mathbf{u}_{1}) \\ \vdots & \ddots & \vdots \\ \kappa(\mathbf{u}_{1}, \mathbf{u}_{m}) & \dots & \kappa(\mathbf{u}_{m}, \mathbf{u}_{m}) \end{bmatrix}$$
(7)

for all *i* and κ is kernel. $\tilde{\sigma}^2$ is the variance of the noise which is independent to system output. κ is the covariance function of the Gaussian process. According to the jointly Gaussian distribution property, for a new data $\{\mathbf{u}_n, d_n\}, n \neq 1, \ldots, m$, the posterior distribution of known model output given the input data \mathbf{u}_n and model is

 $p(\mathbf{y}_n | \mathbf{u}_n, \tau) \sim N(\overline{d}_n, \sigma_n^2)$

where

$$\bar{d}_n = \mathbf{h}_n^T [\tilde{\sigma}^2 \mathbf{I} + \mathbf{G}_m]^{-1} \mathbf{d}_m \tag{9}$$

(8)

$$\sigma_n^2 = \tilde{\sigma}^2 + \kappa(\mathbf{u}_n, \mathbf{u}_n) - \mathbf{h}_n^T [\tilde{\sigma}^2 \mathbf{I} + \mathbf{G}_m]^{-1} \mathbf{h}_n$$
(10)

where $\mathbf{h} = [\kappa(\mathbf{u}, \mathbf{u}_1), \dots, \kappa(\mathbf{u}, \mathbf{u}_m)]^T$ and $\mathbf{d} = [d_1, \dots, d_m]^T$.

Notice $\mathbf{h}_n^T [\tilde{\sigma}^2 \mathbf{I} + \mathbf{G}_m]^{-1} \mathbf{h}_n$ is the computational intensive part, scaling with $O(m^3)$. Such complexity is unacceptable in KLMS. In [9] it is shown that we can approximate σ_n by

$$\sigma_n^2 = \tilde{\sigma}^2 + \kappa(\mathbf{u}_n, \mathbf{u}_n) - \max_{\forall \mathbf{u}_i \in \{\mathbf{u}_1, \dots, \mathbf{u}_m\}} \frac{\kappa^2(\mathbf{u}_n, \mathbf{u}_i)}{\kappa^2(\mathbf{u}_i, \mathbf{u}_i)} \quad (11)$$

in which the computation complexity reduces to O(m). In conclusion,

$$S_{\tau}(\mathbf{u}_{n}, d_{n}) \approx -\ln p(d_{n}|\mathbf{u}_{n}, \tau)$$

$$= \ln \sigma_{n} + \frac{(d_{n} - \overline{d})_{n}^{2}}{2\sigma_{n}^{2}}$$

$$= \ln \sigma_{n} + \frac{e_{n}^{2}}{2\sigma_{n}^{2}}$$
(12)

Surprise is similar to conventional NLL, but applied to the posterior of the data given the system state. Owning to GPR, however, surprise can be estimated online tracking the change in the system state during operation.

C. Online Change Detection Based on KLMS and Surprise

In order to find online a change in statistical structure of a nonstationary time series, the prediction error is normally utilized. An extensive method present in [12] assumes that the prediction error of known system is Gaussian, estimates its mean and variance after the system is trained in the stationary segment, and then applies the generalized likelihood ratio (GLR) test to decide if the new error sample fits the model with a certain confidence interval. If not, the new sample is considered in the change region.

A similar procedure can be utilized using surprise instead of the prediction error, with some advantages. According to Eq.12, the predicted input variance and the prediction error affect the surprise measure together. Predicted input variance is important to judge how large the prediction error really is because if it is small and the error is large, it means that the new sample is different from the acquired model and it is likely an outlier. On the other hand, if the predicted input variance is large and the prediction error is also large, then there is not enough evidence to call the sample an outlier. For the error criterion, the second scenario generates false alarms.

According to the analysis in part II.B, large surprise value implies a change in statistical structure, so it is a compelling evidence to declare a change in statistics. In conclusion, our method can be expressed as follows: First, we choose a segment data $\{x_1, x_2, \ldots, x_m\}$ from one of the stationary regimes of the time series. Then a paired train sequence $\{\{\mathbf{u}_i, d_i\}, i = 1, 2, \ldots\}$ is established by assuming the previous points $[x_{i-(N-1)T}, x_{i-(N-2)T}, \ldots, x_{i-T}]^T$ belongs to the input \mathbf{u}_i at time *i* and x_i is the corresponding desire d_i , where N is predictor order and T is time delay. The KLMS algorithm is applied to the training sequence to establish the current model τ . Finally, the likelihood ratio test based on surprise detects the point of structure change. In the last procedure, we use a two-model GLR as the decision function firstly, with a fixed length window of M samples. The window is slided by R samples. This provides a coarse definition of the break point due to the need to evaluate the GLR test on a window of M samples. This coarse model applies an online nonparametric methodology to detect the breakpoint. If the GLR value is larger than threshold δ_{det} , a breakpoint detection occurs. Once a detection occurs, the three-model GLR is utilized for a fine breakpoint search near the detection point [12]. Even though the fine model is offline, in practice where online applications are required, we can approximate the time when the breakpoint is detected as the time when structure change happens. By repeating in this way, KLMS training procedure is actived to study the new model as soon as the breakpoint is detected.

The threshold is an extremely important parameter. Too high a threshold results in high miss probability $P_M(\delta)$ of breakpoints, while too small a threshold leads to high false-alarm probability $P_F(\delta)$. In this paper, statistics and confidence intervals are utilized to determine appropriate threshold. As soon as the training process is over, we could estimate the distribution of GLR for the data which has the same structure as the known system. Then, according to this distribution and acceptable confidence interval, an appropriate threshold is determined. In conclusion, details are shown in Algorithm. 1.

III. SIMULATION

The system of three ordinary differential equations abstracted by Lorenz is widely used as a benchmark data set for nonlinear learning methods. The system is generated from the following ordinary differential equations:

$$\dot{x} = \sigma(y - x)$$

$$\dot{y} = -xz + \gamma x - y \qquad (13)$$

$$\dot{z} = xy - bz$$

We generate two time systems, one with $\sigma = 16, \gamma = 45.92, B = 4$, another one with $\sigma = 10, \gamma = 28, B = \frac{8}{3}$, and

Algorithm 1

Estimate the current model τ using KLMS Apply τ to the training sequence to obtain σ_0 and a set of $\sigma_0^{\prime 2}$ s based on a sliding window of length M to estimate the distribution f of $Mlog \frac{\sigma_0^{\prime 2}}{\sigma_0^2}$ δ_{det} is estimated from f with confidence level α % now for the test sequence while $\{\mathbf{u}_n, d_n\}$ available do Segment $\{\{\mathbf{u}_{n-M+1}, d_{n-M+1}\}, ..., \{\mathbf{u}_n, d_n\}\}$ as a window $GLR = Mlog \frac{\sigma_1^2}{\sigma_0^2}$ if $GLR > \delta_{det}$ then % structure change happens, search breakpoint T $T = argmax[(T - n + M - 1)log \frac{\sigma_1^{\prime 2}}{\sigma_0^2}]$ $+(n - T + 1)log \frac{\sigma_1^{\prime 2}}{\sigma_0^2}]$ else No structure change, slide window at rate Rend if

end while

(*M* is the window size; $\sigma_0^2, \sigma_0'^2, \sigma_1'^2, \sigma_1'^2, \sigma_1''^2$ are the surprise variances of: whole training sequence, an arbitrary training sequence segment of length *M*, the current window time *n* on the test sequence, the subwindow between n - M + 1 to *T* and the complementary subwindow between *T* to *n*)

pick x dimension to build two time series with a sampling period of 0.01s. Zero mean Gaussian noise with a standard deviation of 0.02 is added to these two series to simulate noise. We segment 1000 samples from the first sequence as training data to estimate the known system. Then we extract 250 samples from two sequence respectively and concatenate them as an example test series. They are shown in Fig.1. The statistical structure of the first 250 test sample is the same as the training, while the last 250 sample has a different structure. We utilize KLMS to train the non linear model. According to [15], the problem setting for short time prediction is as follows: the previous 4 points $\mathbf{u}(i) = [x(i-40), x(i-30), x(i-20), x(i-10)]^T$ are used to predict the present sample x(i). The stepsize equals to 0.2 and a Gaussian kernel with kernel size of 1 is chosen. Then the likelihood ratio test based on prediction error and surprise are applied with a window length of 30 and sliding rate of 5.

TABLE I PERFORMANCE OF ERROR CRITERION AND SURPRISE CRITERION

	false alarm probability	average detection time
surprise	0.053%	265.4
error	1.2%	270

One hundred Monte Carlo simulations are run with different realizations of noise. Table-1 shows that the surprise criterion obtains much smaller false alarm probability and could detect the breakpoint earlier than the error criterion. Fig.2 is a representative result. Fig.2(a) shows the decision

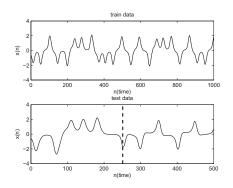
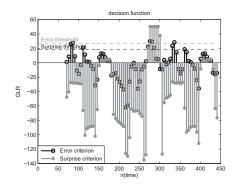
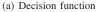
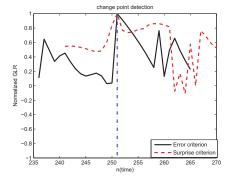


Fig. 1. Simulation data







(b) breakpoint detection and true breakpoint (vertical dotted)

Fig. 2. Performance of error criterion and surprise criterion

function value with the two model GLR for coarse segmentation (resolution is 5 samples) and Fig.2(b) the search for the optimal breakpoint of the two criteria. The surprise criterion detects a coarse change in statistics at n = 265, while the prediction error detects at n = 270. Therefore the surprise criterion detect regime change 5 samples earlier than the error criterion. The fine grain analysis within the window where the change is detected yields the most likely detection for both criterion at sample n = 251. Therefore both methods are able to provide the same segmentation in the fine mode, however the fine mode is not online. In practice where online application are required, the time when the breakpoint is detected is approximated by the breakpoint. Therefore we conclude that in applications where online applications are required, the surprise criterion is advantageous when compared with the prediction error criterion.

IV. CONCLUSIONS

This paper presents a nonparametric information theoretic approach using the KLMS algorithm and the surprise criterion, for structure change detection in nonstationary time series. As we show theoretically and experimentally, the method successfully detects online structure change and locates with high accuracy the change point when compared with the commonly used prediction error criterion. Because the fine scale analysis is noncausal, the best strategy to decrease the delay in detecting the regime change is to advance the data window one sample at a time instead of 5 as done in the paper. Moreover, the approximated surprise metric is easy to evaluate with the KLMS.

Although the preliminary testing was done in a synthetic scenario, we believe that the positive results grant further testing of the methodology in realistic situations. Besides, we applied the methodology in a very simple scenario (single regime change with the model trained off line), but surprise can be extended to the mixture of experts methodology that is able to cope with the online detection of multiple regime changes with a fixed number of models trained online. This extension will be left to future work.

REFERENCES

- [1] O. Bretscher, *Introduction to the Theory of Neural Computation*. Upper Saddle River NJ: Prentice Hall, 3rd ed, 1995.
- [2] J. D. Angrist and G. W Imbens, "Two-Stage Least Squares Estimation of Average Causal Effects in Models with Variable Treatment Intensity," *the American Statistical Association*, vol. 90, pp. 431-442, 1995.
- [3] F. W. Roush, "Discrete statistical models with social science applications : Erling B. Andersen, Amsterdam: North-Holland," *Mathematical Social Sciences*, vol. 2, pp. 215-216, 1981.
- [4] A. M. Mood, F. A. Graybill and D. C. Boes, Introduction to the Theory of Statistics, McGraw-Hill, 1974.
- [5] Sara A. van de Geer, Applications of empirical process theory. Cambridge University Press, PP. 247-262, 2000.
- [6] L. Itti and P. Baldi, "Bayesian surprise attracts human attention," Vision Research, vol. 49, no. 10, pp. 1295-1306, Jun. 2009.
- [7] E. Pfaffelhuber, "Learning and information theory," *International Journal of Neuroscience*, vol. 3, no. 2, pp. 83-88, 1972.
- [8] G. Palm, "Evidence, information, and surprise," *Biological Cybernetics*, vol. 42, no. 1, pp. 57-68, 1981.
- [9] W. Liu, I. Park and J.C. Principe, "An Information Theoretic Approach of Designing Sparse Kernel Adaptive Filters," *IEEE Transactions on Neural Networks*, vol. 20, no. 12, pp. 1950-1961, Dec. 2009.
- [10] W. Liu, P.P. Pokharel and J.C. Principe, "The Kernel Least-Mean-Square Algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 543-554, 2008.
- [11] W. Liu, J.C. Principe and S. Haykin, *Kernel Adaptive Filtering*, WILEY, 2010.
- [12] I. W. Sandberg, J. T. Lo, C.L. Fancourt, J.C. Principe, S. Katagiri and S. Haykin, *Nonlinear Dynamical Systems*, WILEY INTERSCIENCE, pp. 145-164, 2001.
- [13] Y. Engel, S Mannor, and R. Meir, "The kernel recursive least-squares algorithm," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2275 - 2285, 2004.
- [14] C.E. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*, Cambridge, MA: MIT Press, 2006.
- [15] Henry D.I. Abarbanel, *Analysis of obsearved chaotic data*. Springer, 1996.