Int. J. Appl. Math. Comput. Sci., 2018, Vol. 28, No. 3, 559–567
DOI: 10.2478/amcs-2018-0043

REGRESSION FUNCTION AND NOISE VARIANCE TRACKING METHODS FOR DATA STREAMS WITH CONCEPT DRIFT

MACIEJ JAWORSKI [a]

[a] Institute of Computational Intelligence
Częstochowa University of Technology, Armii Krajowej 36, 42-200 Częstochowa, Poland
e-mail: maciej.jaworski@iisi.pcz.pl

Two types of heuristic estimators based on Parzen kernels are presented. They are able to estimate the regression function in an incremental manner. The estimators apply two techniques commonly used in concept-drifting data streams, i.e., the forgetting factor and the sliding window. The methods are applicable for models in which both the function and the noise variance change over time. Although nonparametric methods based on Parzen kernels were previously successfully applied in the literature to online regression function estimation, the problem of estimating the variance of noise was generally neglected. It is sometimes of profound interest to know the variance of the signal considered, e.g., in economics, but it can also be used for determining confidence intervals in the estimation of the regression function, as well as while evaluating the goodness of fit and in controlling the amount of smoothing. The present paper addresses this issue. Specifically, variance estimators are proposed which are able to deal with concept drifting data by applying a sliding window and a forgetting factor, respectively. A number of conducted numerical experiments proved that the proposed methods perform satisfactorily well in estimating both the regression function and the variance of the noise.

Keywords: data streams, concept drift, Parzen kernels, regression, variance estimation.

1. Introduction

In recent years data stream mining have become a very important research area (Alippi *et al.*, 2017; Bifet *et al.*, 2010; Nikulin, 2016; Shaker and Hüllermeier, 2014). It is due to the significant increase of data amounts which need to be processed in various fields of human activity. A data stream can be understood as a sequence in data elements which constantly arrive at the system. The size of the stream can be potentially unlimited and there is no possibility to store all the elements in memory. Moreover, the data elements often arrive at the system with very high rates. In light of the above-mentioned characteristics, traditional data mining algorithms cannot be directly applied to streaming data. Additionally, in the streaming scenario the distribution of data values can often change over time, which is called the 'concept drift' (Ditzler *et al.*, 2015; Gama *et al.*, 2014; Zliobaite *et al.*, 2014). It is desired that algorithms designed for data streams be able to react to such changes.

Although the number of the algorithms for data stream mining is not as large as in the case of traditional data mining, in the recent decade there has been a considerable progress in this field. The most successful seem algorithms based on decision trees (Domingos and Hulten, 2000; Jaworski *et al.*, 2017; Rutkowski *et al.*, 2015; 2013; Weinberg and Last, 2017) and ensemble methods (Pietruczuk *et al.*, 2017; Wang *et al.*, 2003). They are mainly devoted to data classification problems.

In this paper, however, nonparametric methods for the regression problem are considered (Krzyżak and Pawlak, 1984; 1987; Rafajłowicz, 1987; 1989; Rutkowski and Gałkowski, 1994; Greblicki and Pawlak, 2008; Györfi *et al.*, 2002; Mzyk, 2007; Andrzejewski *et al.*, 2013; Rao, 2014; Duda *et al.*, 2017; 2018). Additionally, we also consider the problem of estimating the noise variance. To the best of our knowledge, no research was conducted on this issue in the case of data streams. It should be noted that the problem of variance estimation is very important. Sometimes it is of deep interest to know the noise variance of the considered signal, e.g., in estimating the risk of economic investments. It can be also used for determining the confidence band for the estimation of the regression function (Hart, 1997) as well as for evaluating the goodness of fit (Carroll and Ruppert, 1988), and in controlling the amount of smoothing (Gasser *et al.*, 1991). The confidence interval determination is a very important issue in many real-world applications, e.g., in turbulence modeling considered by Ruppert *et al.* (1997) or in the analysis of financial time series as shown by Fan and Yao (1998). It is also of interest in covariance structure estimation for non-stationary longitudinal data (Diggle and Verbyla, 1998) or in nonparametric regression with log-normal errors (Shen and Brown, 2006).

Motivated by the aforementioned wide spectrum of possible applications, we propose estimators for both regression function and the noise variance, working in streaming scenario. It is assumed that the data stream is a sequence of pairs (X_n, Y_n), $n = 1, 2, \ldots,$ where X_n are i.i.d. univariate random variables with some unknown probability distribution. The probability density function of variables X_n is denoted by $f(x)$. In the case of estimation of the regression function we assume that the target random variables Y_n depend on variables X_n through the following model:

$$Y_n = \phi_n(X_n) + Z_n, \quad n = 1, 2, \ldots, \tag{1}$$

where the random variables Z_n are random variables with zero mean and unknown variance

$$\mathrm{Var}\,(Z_n) = s_n. \tag{2}$$

This means that in this case, both the function $\phi_n(x)$ and the variance of noise s_n can change over time. It should be pointed out that model (1) assumes that the noise is independent of random variables X_n. In the case of estimating the noise variance we also consider a more general problem in which the above-mentioned dependence is included. However, the function $\phi_n(x)$ cannot change in time any more since the variance estimator does not work well otherwise. Therefore, the model considered in the problem of noise variance estimation can be expressed as follows:

$$Y_n = \phi(X_n) + Z_n(X_n), \quad n = 1, 2, \ldots, \tag{3}$$

$$\mathrm{Var}\,(Z_n(x)) = s_n(x), \tag{4}$$

where $s_n(x)$ is any function which we will be trying to estimate. The variance, besides being dependent on X, can still change over time.

In brief, the main results and novelties of this paper can be summarized as follows:

- Nonparametric estimators for the time-changing regression function under nonstationary noise are proposed (model (1)).
- Nonparametric estimators are designed in which the noise variance $\mathrm{Var}(Y|x)$ is estimated. The variance may change over time and depend on X, whereas the regression function is stationary (model (3)).
- The estimators for models (1) and (3) are proposed in two variants. They apply traditional methods which are often used to deal with concept-drifting data: the sliding window and the forgetting factor.

All the considered estimators are based on the Parzen kernel approach, which is often used in various nonparametric estimation methods.

The rest of this paper is organized as follows. In the next two sections, the problem of nonparametric regression function estimation is recalled, starting from estimators for stationary data in Section 2. The modifications of commonly known methods using a sliding window and a forgetting factor are presented in Section 3. In Section 4 the noise variance estimator is introduced and precisely explained. Two variants of the estimator are proposed: one with the sliding window and the other one containing the forgetting factor. The results of numerical simulations are presented in Section 5. Section 6 concludes the paper and outlines possible ideas for future work.

2. Regression function estimation

The problem of estimating the regression function was investigated in the literature very broadly for stationary data, i.e., when $\phi_n(x) \equiv \phi(x)$. A proper estimator of regression can help in predicting the values of the function for incoming data X. Hence, the aim of regression is to overcome the noise and extract the information about the mean signal from noisy data.

At points x for which the probability density function $f(x) \neq 0$ function $\phi(x)$ can be expressed as follows:

$$\phi(x) = \frac{\phi(x) f(x)}{f(x)} = \frac{R(x)}{f(x)}. \tag{5}$$

An estimator of function $\phi(x)$ can then be proposed as the ratio of two estimators: one for function $R(x)$ and the second for density function $f(x)$,

$$\widehat{\phi}_n(x) = \frac{\widehat{R}_n(x)}{\widehat{f}_n(x)}. \tag{6}$$

In this paper recursive estimators of functions $R(x)$ and $f(x)$ will be considered (estimator (6) is then called semirecursive). The reason is that only recursive estimators are applicable in the data stream scenario. In this case the data can be processed incrementally. These estimators were first proposed by Greblicki (1974):

$$\widehat{R}_n(x) = \frac{n-1}{n}\widehat{R}_{n-1}(x) + Y_n \frac{1}{h_n n} K\left(\frac{x - X_n}{h_n}\right), \tag{7}$$

$$\widehat{f}_n(x) = \frac{n-1}{n}\widehat{f}_{n-1}(x) + \frac{1}{h'_n n} K'\left(\frac{x - X_n}{h'_n}\right) \tag{8}$$

where $K(u)$, $K'(u)$ are kernel functions (Parzen, 1962). The sequences h_n, h'_n are called bandwidths and should satisfy the following conditions:

$$\lim_{n\to\infty} h_n = 0, \quad \lim_{n\to\infty} nh_n = \infty, \tag{9}$$

$$\lim_{n\to\infty} h'_n = 0, \quad \lim_{n\to\infty} nh'_n = \infty. \tag{10}$$

There are many possible kernel functions. In this work an example of a one-dimensional (i.e., $u \in \mathbb{R}$) kernel is considered, i.e., the Epanechnikov kernel (Epanechnikov, 1969) given by

$$K(u) = \begin{cases} 0.75\left(1-u^2\right), & |u| \le 1, \\ 0, & |u| > 1. \end{cases} \tag{11}$$

The estimator (6) works well for static data even if the noise variance grows to infinity (with an appropriately low rate). However, it is not suited to deal with data for which the function $\phi_n(x)$ changes over time.

Let us further assume that the sequences h_n and h'_n from the estimators (7) and (8), respectively, are the same, i.e., $h_n \equiv h'_n$, $n = 1, 2, \ldots$. Moreover, assume that the same kernels are used for both estimators, i.e., $K(u) \equiv K'(u)$. Then the estimator (6) can be rewritten in the following form:

$$\widehat{\phi}_n(x) = \frac{\sum_{i=1}^{n} Y_i \frac{1}{h_i} K\left(\frac{x-X_i}{h_i}\right)}{\sum_{i=1}^{n} \frac{1}{h_i} K\left(\frac{x-X_i}{h_i}\right)}. \tag{12}$$

The most frequently considered form of bandwidth sequence h_n is

$$h_n = Dn^{-H}, \tag{13}$$

where $D > 0$ and $0 < H < 1$. In this paper only the bandwidths expressed by (13) will be considered. Based on the estimator (12) for static data, in the next section we will present regression function estimators able to deal with concept drifting data.

3. Concept drift handling

It is easily seen that the estimator (12) is a weighted average of subsequent values of Y_i, i.e.,

$$\widehat{\phi}_n(x) = \frac{\sum_{i=1}^{n} w_i Y_i}{\sum_{i=1}^{n} w_i}, \tag{14}$$

where w_i, $i = 1, \ldots, n$, are some weights which take different forms in different kinds of estimators (they depend on the chosen kernel and bandwidth sequence parameters). This general form can form a basis for estimators able to handle data with concept drift, i.e., data that are drawn according to the time-varying function $\phi_n(x)$. It should be noted that Rutkowski (2004) and Pietruczuk *et al.* (2014) analyzed an estimator of a time-varying function $\phi_n(x)$ which applied the ideas of stochastic approximation. Its convergence under certain assumptions was proved as well. In this estimator only the formula for the estimator of $R(x)$ is changed whereas the estimator of $f(x)$ remains the same as (8). Therefore, the above-mentioned regression function estimator cannot be expressed by the general weighted average scheme (14). Two other types of estimators able to handle concept drift will be presented below (one with a sliding window approach and the second which applies the forgetting mechanism). These estimators are heuristic (i.e., their convergence is not proved), but they can be expressed using the general formula (14). The weight w_i of the i-th datum changes over time with n.

3.1. Sliding window approach.

In the case of a sliding window of a size W, the appropriate estimator of the regression function is given by

$$\begin{aligned} \overline{\phi}_n(x;W) &= \frac{\sum_{i=n-W+1}^{n} Y_i \frac{1}{\overline{h}_i(W)} K\left(\frac{x-X_i}{\overline{h}_i(W)}\right)}{\sum_{i=n-W+1}^{n} \frac{1}{\overline{h}_i(W)} K\left(\frac{x-X_i}{\overline{h}_i(W)}\right)} \\ &= \frac{\overline{R}_n(x;W)}{\overline{f}_n(x;W)} \end{aligned} \tag{15}$$

Estimators $\overline{R}_n(x;W)$ and $\overline{f}_n(x;W)$ can be expressed using recursive formulas respectively as follows:

$$\begin{aligned} &\overline{R}_n(x;W) \\ &= \begin{cases} \overline{R}_{n-1}(x;W) + \frac{Y_n K\left(\frac{x-X_n}{\overline{h}_n(W)}\right)}{\overline{h}_n(W)}, & n \le W, \\ \overline{R}_{n-1}(x;W) + \frac{Y_n K\left(\frac{x-X_n}{\overline{h}_n(W)}\right)}{\overline{h}_n(W)} \\ -\frac{Y_{n-W} K\left(\frac{x-X_{n-W}}{\overline{h}_{n-W}(W)}\right)}{\overline{h}_{n-W}(W)}, & n > W, \end{cases} \end{aligned} \tag{16}$$

$$\begin{aligned} &\overline{f}_n(x;W) \\ &= \begin{cases} \overline{f}_{n-1}(x;W) + \frac{K\left(\frac{x-X_n}{\overline{h}_n(W)}\right)}{\overline{h}_n(W)}, & n \le W, \\ \overline{f}_{n-1}(x;W) + \frac{K\left(\frac{x-X_n}{\overline{h}_n(W)}\right)}{\overline{h}_n(W)} \\ -\frac{K\left(\frac{x-X_{n-W}}{\overline{h}_{n-W}(W)}\right)}{\overline{h}_{n-W}(W)}, & n > W. \end{cases} \end{aligned} \tag{17}$$

To have the form of the bandwidth sequence analogous to (13), an effective number of data $M(n;W)$ should be introduced. This quantity expresses how many data elements contributes to the current value of the estimator. In the case of the sliding window we have

$$M(n;W) = \min\{n, W\}. \tag{18}$$

Therefore,

$$\overline{h}_n(W) = D\,[M(n;W)]^{-H} = D\left(\min\{n,W\}\right)^{-H}. \tag{19}$$

3.2. Forgetting factor approach. If the forgetting mechanism with a forgetting factor $0 < \lambda < 1$ is applied, then the corresponding estimator of the regression function is given by

$$\begin{aligned}\widetilde{\phi}_n(x;\lambda) &= \frac{\sum_{i=1}^{n} Y_i \lambda^{n-i} \frac{1}{\widetilde{h}_i(\lambda)} K\left(\frac{x-X_i}{\widetilde{h}_i(\lambda)}\right)}{\sum_{i=1}^{n} \lambda^{n-i} \frac{1}{\widetilde{h}_i(\lambda)} K\left(\frac{x-X_i}{\widetilde{h}_i(\lambda)}\right)} \\ &= \frac{\widetilde{R}_n(x;\lambda)}{\widetilde{f}_n(x;\lambda)}.\end{aligned} \tag{20}$$

The recursive formulas for estimators are

$$\begin{aligned}\widetilde{R}_n(x;\lambda) &= \lambda \widetilde{R}_{n-1}(x;\lambda) \\ &\quad + Y_n \frac{1}{\widetilde{h}_n(\lambda)} K\left(\frac{x-X_n}{\widetilde{h}_n(\lambda)}\right),\end{aligned} \tag{21}$$

$$\begin{aligned}\widetilde{f}_n(x;\lambda) &= \lambda \widetilde{f}_{n-1}(x;\lambda) \\ &\quad + \frac{1}{\widetilde{h}_n(\lambda)} K\left(\frac{x-X_n}{\widetilde{h}_n(\lambda)}\right).\end{aligned} \tag{22}$$

The effective number of data $M(n;\lambda)$ in the case of the forgetting factor approach is slightly more complicated. The i-th datum contributes to the value of the estimator with weight λ^{n-i}. Hence it can be considered as an incomplete data element, i.e., a fraction λ^{n-i} of an element. Therefore the effective number of data is a partial sum of the geometric series

$$\begin{aligned}M(n;\lambda) &= 1 + \cdots + \lambda^{n-1} = \sum_{i=0}^{n-1} \lambda^{n-i} \\ &= \frac{1-\lambda^n}{1-\lambda}.\end{aligned} \tag{23}$$

Analogously to formulas (13) and (19), we have

$$\widetilde{h}_n(\lambda) = D\,[M(n;\lambda)]^{-H} = D\left(\frac{1-\lambda^n}{1-\lambda}\right)^{-H}. \tag{24}$$

4. Noise variance estimation

As was stated previously, the aim of regression is to overcome noise and extract information about the mean signal from noisy data. On the other hand, knowledge about the noise variance can be helpful in other kinds of tasks, i.e., in estimating the interval of possible values of Y for newly incoming data X. Whereas the estimation of the regression function was investigated in the literature for both stationary and non-stationary data, i.e., for $\phi(x)$ or $\phi_n(x)$, to the best of our knowledge, estimation of the variance was considered only for the stationary case, mainly within the designed experimental setting in which properly chosen values of argument x were used instead of the random variables X_i. The first variance estimators which did not require the estimation of the regression function were proposed by von Neumann (1941). This kind of estimators is called difference-based and they were further improved by Gasser *et al.* (1986). Hall and Carroll (1989) considered a different approach in which the average of squared residuals from fitting to function $\phi(x)$ was taken into account. The application of kernel methods to estimate the variance was investigated by Brown and Levine (2007). Dai *et al.* (2015) considered the case of repeated measurements of data (i.e., multiple measurements of Y_n for the same value of x).

Analogously to (14), it seems at first sight that the general form of the noise variance estimator could be expressed as the following weighted average:

$$\widehat{\sigma^2}_n(x) = \frac{\sum_{i=1}^{n} w_i \left(Y_i - \widehat{\phi}_n(x)\right)^2}{\sum_{i=1}^{n} w_i}. \tag{25}$$

where, as previously, the appropriate forms of weights correspond to estimators based on sliding windows or a forgetting mechanism. Such estimators would approximate the mean squared deviation between the Y_i variables and the value of the regression function estimator at point x. In other words, they would try to estimate the quantity $E[(Y_i - E\,[Y_i|x])^2\,|x]$. However, what is really needed is the estimation of $E[(Y_i - E\,[Y_i|X_i = x])^2\,|x]$. The random variable X_i corresponding to Y_i may be far away from x and, in consequence, the quantity $(Y_i - \widehat{\phi}_n(x))^2$ may differ much from $(Y_i - E\,[Y|X_i = x])^2$, especially if the function $\phi(x)$ has a large value of the derivative at point x. Therefore, it might be better to directly estimate the desired square difference $E[(Y_i - E\,[Y_i|X_i = x])^2\,|x]$. The resulting noise variance estimators can be thus proposed as follows. In the case of the sliding window approach the estimators are

$$\begin{aligned}&\overline{\Psi^2}_n(x;W)\\&=\frac{\overline{r^2}_n(x;W)}{\overline{f}_n(x;W)}\\&=\frac{\sum\limits_{i=n-W+1}^{n}\left(Y_i-\overline{\phi}_i(X_i;W)\right)^2\frac{1}{\overline{h}_i(W)}K\left(\frac{x-X_i}{\overline{h}_i(W)}\right)}{\sum\limits_{i=n-W+1}^{n}\frac{1}{\overline{h}_i(W)}K\left(\frac{x-X_i}{\overline{h}_i(W)}\right)}.\end{aligned}\tag{26}$$

Estimator $\overline{f}_n(x;W)$ is the same as (17) and $\overline{r^2}_n(x;W)$ is recursively given by

$$\begin{aligned}&\overline{r^2}_n(x;W)\\&=\begin{cases}\overline{r^2}_{n-1}(x;W)\\+\frac{\left(Y_n-\overline{\phi}_n(X_n;W)\right)^2K\left(\frac{x-X_n}{\overline{h}_n(W)}\right)}{\overline{h}_n(W)}, & n\leq W,\\\overline{r^2}_{n-1}(x;W)\\+\frac{\left(Y_n-\overline{\phi}_n(X_n;W)\right)^2K\left(\frac{x-X_n}{\overline{h}_n(W)}\right)}{\overline{h}_n(W)}\\-\frac{\left(Y_{n-W}-\overline{\phi}_{n-W}(X_{n-W};W)\right)^2K\left(\frac{x-X_{n-W}}{\overline{h}_{n-W}(W)}\right)}{\overline{h}_{n-W}(W)} & n>W.\end{cases}\end{aligned}\tag{27}$$

In the case of the forgetting factor approach, the corresponding estimators are

$$\begin{aligned}&\widetilde{\Psi^2}_n(x;\lambda)\\&=\frac{\widetilde{r^2}_n(x;\lambda)}{\widetilde{f}_n(x;\lambda)}\\&=\frac{\sum\limits_{i=1}^{n}\lambda^{n-i}\left(Y_i-\widetilde{\phi}_i(X_i;\lambda)\right)^2\frac{1}{\widetilde{h}_i(\lambda)}K\left(\frac{x-X_i}{\widetilde{h}_i(\lambda)}\right)}{\sum\limits_{i=1}^{n}\lambda^{n-i}\frac{1}{\widetilde{h}_i(\lambda)}K\left(\frac{x-X_i}{\widetilde{h}_i(\lambda)}\right)},\end{aligned}\tag{28}$$

$$\begin{aligned}&\widetilde{r^2}_n(x;\lambda)\\&=\lambda\widetilde{r^2}_{n-1}(x;\lambda)\\&\quad+\left(Y_n-\widetilde{\phi}_n(X_n;\lambda)\right)^2\frac{1}{\widetilde{h}_n(\lambda)}K\left(\frac{x-X_n}{\widetilde{h}_n(\lambda)}\right).\end{aligned}\tag{29}$$

The estimator $\widetilde{f}_n(x;\lambda)$ is obviously the same as in (22).

5. Simulation results

All experiments considered in this paper were conducted on synthetic datasets. Unfortunately, there is no real dataset which could allow measuring the performance of proposed algorithms for noise variance estimation on real data. None of the existing real sets contains information about the variance of noise according to which the data were generated. However, we believe that the experiments on synthetic data presented further in this section will help in gaining insight into how the proposed estimators work on any kind of datasets including the real ones. In each experiment the dataset consisted of 100000 elements where the random variables X_i were drawn from the uniform distribution in interval $[\min,\max]=[-3,3]$. Each estimator was evaluated on a set of $N_p=101$ equidistant points x_j, $j=1,\ldots,101$, i.e.,

$$x_j=\min+\frac{j\,(\max-\min)}{N_p-1}=-3+0.06j.\tag{30}$$

To evaluate the performance of any regression function, the mean squared error estimator can be applied,

$$\text{MSE}\left(\widehat{\phi}_n(x)\right)=\sqrt{\frac{1}{N_p}\sum_{j=1}^{N_p}\left(\widehat{\phi}_n(x_j)-\phi_n(x_j)\right)^2}.\tag{31}$$

In the case of variance estimators, an analogous measure can be defined, i.e.,

$$\text{MSE}\left(\widehat{\sigma^2}_n(x)\right)=\sqrt{\frac{1}{N_p}\sum_{j=1}^{N_p}\left(\widehat{\sigma^2}_n(x_j)-s_n(x_j)\right)^2}.\tag{32}$$

5.1. Regression function estimation.

In the first experiment the performance of the proposed estimators (15) and (20) was examined in the case where the function $\phi_n(x)$ changes over time. Then the model (1) was considered with noise variables (2). The following function $\phi_n(x)$ was analyzed:

$$\phi_n(x)=n^{\beta}2\cos(2x-3)\sin(x+2),\tag{33}$$

where the coefficient β was set to 0.1. The noise variables Z_n were sampled from the Gaussian distribution with zero mean and the variance was set as the following increasing sequence:

$$s_n=n^{\alpha},\tag{34}$$

with α set to 0.15. The Epanechnikov kernel given by (11) was applied and the parameters D and H of sequences $\overline{h}_n(W)$ and $\widetilde{h}_n(\lambda)$ were set to $D=2$ and $H=0.3$.

At the beginning of the experiment, the most appropriate values of λ and W were found. In preliminary simulations we tested 60 different values of λ in interval $[0.999,1]$ as well as 60 different sliding window sizes W in interval $[1000,35000]$. We wanted to choose the values of parameters which would provide high estimation quality not only at the end of the simulation (i.e., after $n=100000$ data elements processed in this case) but during the whole simulation. Therefore, another performance

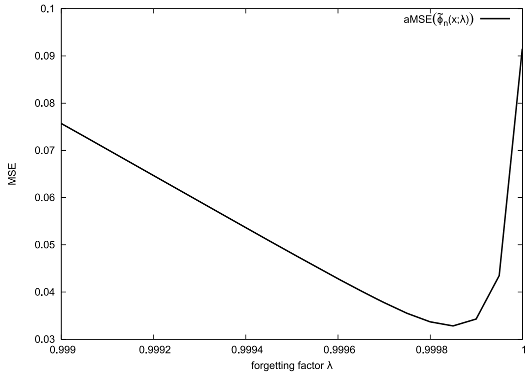

Fig. 1. Dependence of the averaged MSE on the value of λ for the regression function estimator with the forgetting factor in the case of non-stationary variance s_n and the non-stationary function given by (33).

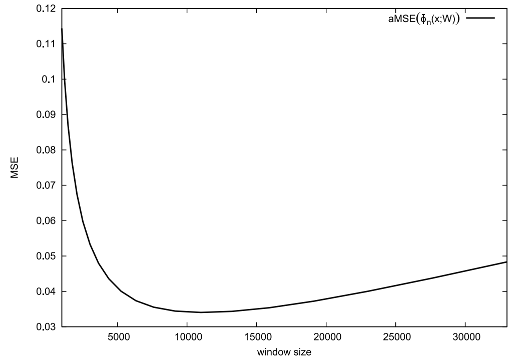

Fig. 2. Dependence of the average MSE on W for the regression function estimator with the sliding window in the case of non-stationary variance s_n and the non-stationary function given by (33).

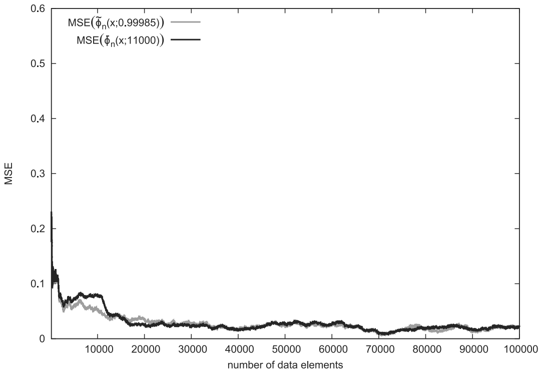

Fig. 3. MSE as a function of the number of processed data elements for the regression function estimators $\overline{\phi}_n(x;11000)$ and $\widetilde{\phi}_n(x;0.99985)$ in the case of non-stationary noise variance s_n and the non-stationary function given by (33).

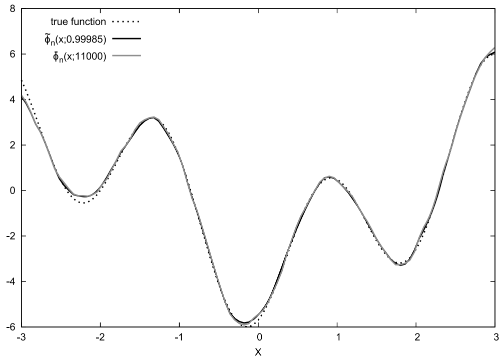

Fig. 4. Fit of the regression function estimators $\overline{\phi}_n(x;11000)$ and $\widetilde{\phi}_n(x;0.99985)$ to the true function $\phi_n(x)$ given by (33) after processing $n = 100000$ data elements.

measure of any estimator $\widehat{\phi}_n(x)$ is introduced, i.e., average mean square error (aMSE) which is defined as the arithmetic mean of the MSE results obtained for all values of n,

$$a\text{MSE}\left(\widehat{\phi}_n(x)\right) = \frac{1}{n}\sum_{q=1}^{n}\text{MSE}\left(\widehat{\phi}_q(x)\right). \tag{35}$$

The dependence between the obtained values of aMSE and the forgetting factor λ for estimator $\widetilde{\phi}_n(x;\lambda)$ is depicted in Fig. 1. Similarly, the values of aMSE as a function of the sliding window size W for estimator $\overline{\phi}_n(x;W)$ is presented in Fig. 2. The obtained results demonstrate that the most effective values of the forgetting factor and the sliding window are $\lambda = 0.99985$ and $W = 11000$. Therefore, two different estimators were investigated in the further part of the experiment, $\overline{\phi}_n(x;11000)$ and $\widetilde{\phi}_n(x;0.99985)$. The resulting MSE values for the considered estimators are presented in Fig. 3. In Fig. 4 the fits of the estimators are compared with the true function $\phi_n(x)$ given by (33).

It seems that the estimators with the sliding window and the forgetting factor are comparable. Despite the fact that both the variance and the regression function $\phi_n(x)$ were increasing with n, the estimators are almost indistinguishable from the true function.

The multiplicative polynomial non-stationarity used in the presented simulation was chosen arbitrarily as an example. The estimators also deal well with other types of non-stationarities. As another example, the following function with the ‘moving’ argument is considered:

$$\phi_n(x) = \frac{10(x - cn)}{1 + (x - cn)^2}, \tag{36}$$

where $c = 0.00003$. As previously, the noise variables are normally distributed with zero mean and the variance given by (34) with $\alpha = 0.15$. The best values of $W = 2100$ and $\lambda = 0.999$ were chosen experimentally. The

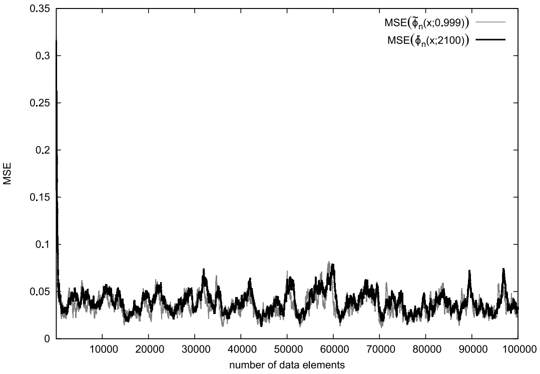

Fig. 5. MSE as a function of the number of processed data elements for the regression function estimators $\overline{\phi}_n(x; 2100)$ and $\widetilde{\phi}_n(x; 0.999)$ in the case of non-stationary noise variance s_n and the non-stationary function given by (36).

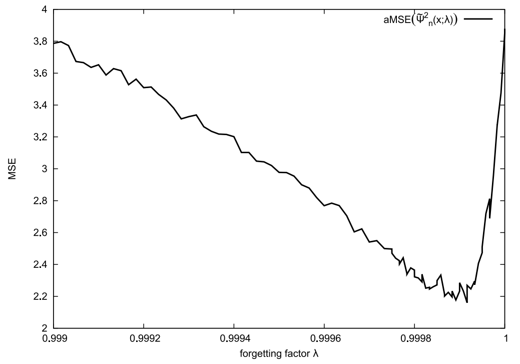

Fig. 6. Dependence of the averaged MSE on λ for the variance estimator with the forgetting factor in the case of non-stationary variance $s_n(x)$ given by (38) with $\alpha = 0.15$.

obtained results are shown in Fig. 5.

5.2. Variance estimation. In this subsection the performance of proposed estimators (26) and (28) was examined in the case where the noise variance depends on x, i.e., model (3) was considered with the noise variables (4). The following function $\phi(x)$ was considered:

$$\phi(x) = 10\,\mathrm{atan}(x) + 10. \tag{37}$$

The noise variables were normally distributed zero mean and the variance

$$s_n(x) = n^{\alpha}\left(0.5\sin(2x) + 5\right). \tag{38}$$

The parameter α was set to 0.15. The Epanechnikov kernel given by (11) was applied and the parameters D and H of sequences $\overline{h}_n(W)$ and $\widetilde{h}_n(\lambda)$ were set to $D = 2$ and $H = 0.3$. As was previously done for the case of regression function estimation, satisfactory values of λ and W were found first. In preliminary simulations we tested 60 different values of λ in interval $[0.999, 1]$ as well as 60 different sliding window sizes W in interval $[1000, 36000]$. The dependence between the obtained values of aMSE (calculated analogously as in (35)) and the forgetting factor λ for estimator $\widetilde{\Psi^2}_n(x;\lambda)$ is depicted in Fig. 6. Similarly, the values of aMSE as a function of the sliding window size W for estimator $\overline{\Psi^2}_n(x;W)$ is presented in Fig. 7.

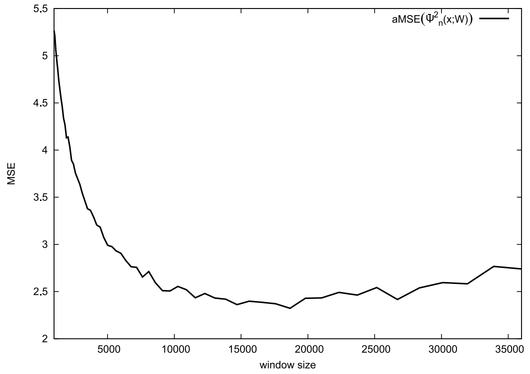

Fig. 7. Dependence of the average MSE on W for the variance estimator with the sliding window in the case of non-stationary variance $s_n(x)$ given by (38) with $\alpha = 0.15$.

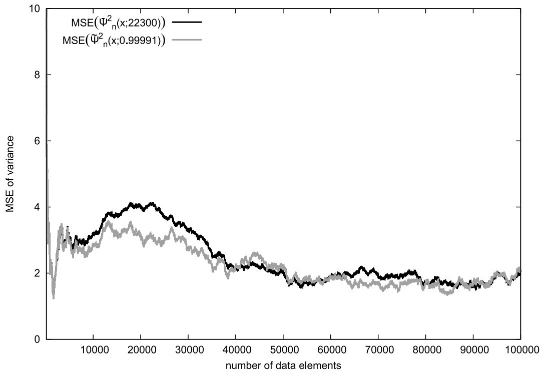

Fig. 8. MSE as a function of the number of processed data elements for the variance estimators $\overline{\Psi^2}_n(x; 22300)$ and $\widetilde{\Psi^2}_n(x; 0.99991)$ in the case of non-stationary noise variance $s_n(x)$ given by (38) with $\alpha = 0.15$.

According to the obtained results, for the further part of the experiment the following values of the forgetting factor and the sliding window size were chosen: $\lambda = 0.99991$ and $W = 22300$. Summarizing, two different estimators were investigated: $\overline{\Psi^2}_n(x; 22300)$ and $\widetilde{\Psi^2}_n(x; 0.99991)$. The resulting MSE values obtained for the estimators considered are depicted in Fig. 8 whereas Fig. 9 demonstrates the fits of the estimators to the true variance.

It can be seen that both the estimators with the sliding window and the forgetting factor keep around the true variance—they track satisfactorily well the increasing

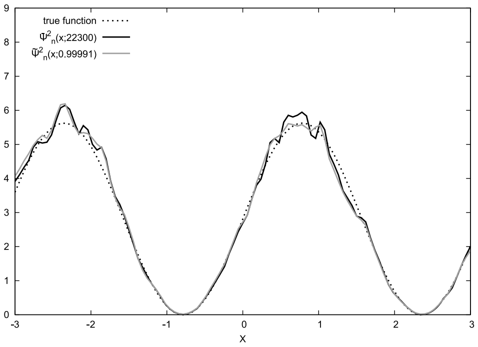

Fig. 9. Fit of the variance estimators $\overline{\Psi^2}_n(x;22300)$ and $\widetilde{\Psi^2}_n(x;0.99991)$ to true variance $s_n(x)$ given by (38) with $\alpha = 0.15$ after processing $n = 100000$ data elements.

value of the true variance. The estimators provide comparable results after processing a sufficiently large number of data elements. At the beginning of the data stream, the estimator with the sliding window is slightly worse than the one with the forgetting factor. This is due to the fact that by the time when the sliding window have been filled up, estimators (16) and (17) behave like stationary estimators (7) and (8).

6. Conclusions

In this paper nonparametric algorithms based on semirecursive kernel estimates for recovering of regression function and noise variance were considered. The estimators were proposed in two variants. They contain commonly known tools which allow dealing with concept drifting data, i.e., the sliding window and the forgetting factor. In a series of numerical experiments, the proposed methods proved to perform satisfactorily well in estimating both the regression function and the variance of the noise.

The conducted research opens up several possible directions for future work. In the case of variance estimation, the time-varying variance was considered, but the function $\phi(x)$ was stationary. It would be worth working on proposing appropriate estimators in the case in which both the variance and the function $\phi(x)$ change over time. Another idea is to propose new estimators in which the Parzen kernels are replaced by application of some orthogonal series, e.g., Hermit's or Fourier's ones.

Acknowledgment

This work was supported by the Polish National Science Centre under the grant no. 2014/15/B/ST7/05264.

References

Alippi, C., Boracchi, G. and Roveri, M. (2017). Hierarchical change-detection tests, *IEEE Transactions on Neural Networks and Learning Systems* **28**(2): 246–258.

Andrzejewski, W., Gramacki, A. and Gramacki, J. (2013). Graphics processing units in acceleration of bandwidth selection for kernel density estimation, *International Journal of Applied Mathematics and Computer Science* **23**(4): 869–885, DOI: 10.2478/amcs-2013-0065.

Bifet, A., Holmes, G., Kirkby, R. and Pfahringer, B. (2010). MOA: Massive online analysis, *Journal of Machine Learning Research* **11**: 1601–1604.

Brown, L.D. and Levine, M. (2007). Variance estimation in nonparametric regression via the difference sequence method, *Annals of Statistics* **35**(5): 2219–2232.

Carroll, R.J. and Ruppert, D. (1988). *Transformation and Weighting in Regression*, CRC Press, Boca Raton, FL.

Dai, W., Ma, Y., Tong, T. and Zhu, L. (2015). Difference-based variance estimation in nonparametric regression with repeated measurement data, *Journal of Statistical Planning and Inference* **163**: 1–20.

Diggle, P.J. and Verbyla, A.P. (1998). Nonparametric estimation of covariance structure in longitudinal data, *Biometrics* **54**(2): 401–415.

Ditzler, G., Roveri, M., Alippi, C. and Polikar, R. (2015). Learning in nonstationary environments: A survey, *IEEE Computational Intelligence Magazine* **10**(4): 12–25.

Domingos, P. and Hulten, G. (2000). Mining high-speed data streams, *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Boston, MA, USA*, pp. 71–80.

Duda, P., Jaworski, M. and Rutkowski, L. (2017). Knowledge discovery in data streams with the orthogonal series-based generalized regression neural networks, *Information Sciences* **460–461**: 497–518.

Duda, P., Jaworski, M. and Rutkowski, L. (2018). Convergent time-varying regression models for data streams: Tracking concept drift by the recursive Parzen-based generalized regression neural networks, *International Journal of Neural Systems* **28**(02): 1750048.

Epanechnikov, V.A. (1969). Non-parametric estimation of a multivariate probability density, *Theory of Probability & Its Applications* **14**(1): 153–158.

Fan, J. and Yao, Q. (1998). Efficient estimation of conditional variance functions in stochastic regression, *Biometrika* **85**(3): 645–660.

Gama, J., Zliobaite, I., Bifet, A., Pechenizkiy, M. and Bouchachia, A. (2014). A survey on concept drift adaptation, *ACM Computing Surveys (CSUR)* **46**(4): 44:1–44:37.

Gasser, T., Kneip, A. and Köhler, W. (1991). A flexible and fast method for automatic smoothing, *Journal of the American Statistical Association* **86**(415): 643–652.

Gasser, T., Sroka, L. and Jennen-Steinmetz, C. (1986). Residual variance and residual pattern in nonlinear regression, *Biometrika* **73**(3): 625–633.

Greblicki, W. (1974). *Asymptotically Optimal Algorithms of Recognition and Identification in Probabilistic Conditions*, BI Wrocław University of Technology, Wrocław, (in Polish).

Greblicki, W. and Pawlak, M. (2008). *Nonparametric System Identification*, Cambridge University Press, Cambridge.

Györfi, L., Kohler, M., Krzyzak, A. and Walk, H. (2002). *A Distribution-Free Theory of Nonparametric Regression*, Springer, New York, NY.

Hall, P. and Carroll, R.J. (1989). Variance function estimation in regression: The effect of estimating the mean, *Journal of the Royal Statistical Society: Series B (Methodological)* **51**(1): 3–14.

Hart, J. (1997). *Nonparametric Smoothing and Lack-of-Fit Tests*, Springer, New York, NY.

Jaworski, M., Duda, P. and Rutkowski, L. (2017). New splitting criteria for decision trees in stationary data streams, *IEEE Transactions on Neural Networks and Learning Systems* **PP**(99): 1–14.

Krzyzak, A. and Pawlak, M. (1984). Almost everywhere convergence of a recursive regression function estimate and classification, *IEEE Transactions on Information Theory* **30**(1): 91–93.

Krzyzak, A. and Pawlak, M. (1987). The pointwise rate of convergence of the kernel regression estimate, *Journal of Statistical Planning and Inference* **16**: 159–166.

Mzyk, G. (2007). Generalized kernel regression estimate for the identification of Hammerstein systems, *International Journal of Applied Mathematics and Computer Science* **17**(2): 189–197, DOI: 10.2478/v10006-007-0018-z.

Nikulin, V. (2016). Prediction of the shoppers loyalty with aggregated data streams, *Journal of Artificial Intelligence and Soft Computing Research* **6**(2): 69–79.

Parzen, E. (1962). On estimation of probability density function and mode, *Annals of Mathematical Statistics* **33**: 1065–1076.

Pietruczuk, L., Rutkowski, L., Jaworski, M. and Duda, P. (2014). The Parzen kernel approach to learning in non-stationary environment, *Proceedings of the International Joint Conference on Neural Networks (IJCNN), Beijing, China*, pp. 3319–3323.

Pietruczuk, L., Rutkowski, L., Jaworski, M. and Duda, P. (2017). How to adjust an ensemble size in stream data mining?, *Information Sciences* **381**: 46–54.

Rafajlowicz, E. (1987). Nonparametric orthogonal series estimators of regression: A class attaining the optimal convergence rate in L2, *Statistics and Probability Letters* **5**(3): 219–224.

Rafajlowicz, E. (1989). Reduction of distributed system identification complexity using intelligent sensors, *International Journal of Control* **50**(5): 1571–1576.

Rao, B.P. (2014). *Nonparametric Functional Estimation*, Academic Press, Cambridge, MA.

Ruppert, D., Wand, M.P., Holst, U. and Hössjer, O. (1997). Local polynomial variance-function estimation, *Technometrics* **39**(3): 262–273.

Rutkowski, L. (2004). Generalized regression neural networks in time-varying environment, *IEEE Transactions on Neural Networks* **15**: 576–596.

Rutkowski, L. and Galkowski, T. (1994). On pattern classification and system identification by probabilistic neural networks, *International Journal of Applied Mathematics and Computer Science* **4**(3): 413–422.

Rutkowski, L., Jaworski, M., Pietruczuk, L. and Duda, P. (2015). A new method for data stream mining based on the misclassification error, *IEEE Transactions on Neural Networks and Learning Systems* **26**(5): 1048–1059.

Rutkowski, L., Pietruczuk, L., Duda, P. and Jaworski, M. (2013). Decision trees for mining data streams based on the McDiarmid's bound, *IEEE Transactions on Knowledge and Data Engineering* **25**(6): 1272–1279.

Shaker, A. and Hüllermeier, E. (2014). Survival analysis on data streams: Analyzing temporal events in dynamically changing environments, *International Journal of Applied Mathematics and Computer Science* **24**(1): 199–212, DOI: 10.2478/amcs-2014-0015.

Shen, H. and Brown, L.D. (2006). Non-parametric modelling of time-varying customer service times at a bank call centre, *Applied Stochastic Models in Business and Industry* **22**(3): 297–311.

von Neumann, J. (1941). Distribution of the ratio of the mean square successive difference to the variance, *Annals Mathematic of Statistics* **12**(4): 367–395.

Wang, H., Fan, W., Yu, P.S. and Han, J. (2003). Mining concept-drifting data streams using ensemble classifiers, *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'03, Washington, DC, USA*, pp. 226–235.

Weinberg, A.I. and Last, M. (2017). Interpretable decision-tree induction in a big data parallel framework, *International Journal of Applied Mathematics and Computer Science* **27**(4): 737–748, DOI: 10.1515/amcs-2017-0051.

Zliobaite, I., Bifet, A., Pfahringer, B. and Holmes, G. (2014). Active learning with drifting streaming data, *IEEE Transactions on Neural Networks and Learning Systems* **25**(1): 27–39.

Maciej Jaworski was born in Częstochowa, Poland, in 1985. He received the MSc degree in theoretical physics from Jagiellonian University, Cracow, Poland, in 2009, and the MSc degree in applied computer science from the AGH University of Science and Technology, Cracow, in 2011. In 2015 he obtained the PhD degree in computer science at the Institute of Computational Intelligence, Częstochowa University of Technology. His current research interests include computational intelligence, data stream mining, neural networks and deep learning.

Received: 16 February 2018
Revised: 10 April 2018
Accepted: 4 May 2018