Computational Statistics and Data Analysis 56 (2012) 327–344

Contents lists available at SciVerse ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Online wavelet-based density estimation for non-stationary streaming data

E.S. García-Treviño, J.A. Barria *

Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, United Kingdom

ARTICLE INFO

Article history:
Received 18 February 2011
Received in revised form 8 August 2011
Accepted 12 August 2011
Available online 25 August 2011

Keywords:
Probability density estimation
Orthogonal density estimators
Wavelet density estimators
Data streams modelling
Streaming data analysis

ABSTRACT

There has been an important emergence of applications in which data arrives in an online time-varying fashion (e.g. computer network traffic, sensor data, web searches, ATM transactions) and it is not feasible to exchange or to store all the arriving data in traditional database systems to operate on it. For this kind of applications, as it is for traditional static database schemes, density estimation is a fundamental block for data analysis. A novel online approach for probability density estimation based on wavelet bases suitable for applications involving rapidly changing streaming data is presented. The proposed approach is based on a recursive formulation of the wavelet-based orthogonal estimator using a sliding window and includes an optimised procedure for reevaluating only relevant scaling and wavelet functions each time new data items arrive. The algorithm is tested and compared using both simulated and real world data.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Traditional database and data processing systems are based on the storage and analysis of static records which generally have no predefined notion of time (Golab and Ozsu, 2003). In those models, conceived as *fixed static data* archives, data take the form of persistent relations that require persistent data storage as well as complex querying operations (Babcock et al., 2002; Golab and Ozsu, 2003). Recently, a new class of applications has emerged in which information occurs in the form of a sequence and a large amount of data is generated at a rapid rate; notable examples are: sensor networks, network traffic analysis, financial tickers, web clicks, transaction log analysis, etc. What is common in all these particular applications is that data arrive in a continuous, rapid and time varying fashion and an immediate processing and analysis of large *transient data streams* is required. In order to manage the issues of this rapidly changing data, new data model paradigms have been introduced in the literature; see for example the work of Babcock et al. (2002), Golab and Ozsu (2003) and Domingos and Hulten (2003). These new data models, referred by Babcock et al. (2002) as *data stream* models, are basically founded on the non-feasibility to store complete streams (which are considered potentially unbound in size) and as a result, time-based processing and querying operations are required to be performed as new data items arrive. Specifically, according to Golab and Ozsu (2003), a *data stream* is a real-time, continuous, ordered sequence of items whose order is implicit when is represented by the arrival time or explicit when it is indicated by timestamps.

In data processing and analysis systems, density estimation is a fundamental block, for both, fixed static data-based and streaming data-based applications. According to the data model paradigm in which they are found, i.e. fixed static or streaming, two main variants of density estimators can be distinguished: batch-processing techniques and online-processing methods, respectively. In batch-processing algorithms, the underlying density is obtained by processing all the

* Correspondence to: Department of Electrical and Electronic Engineering, Imperial College London, South Kensington, London SW7 2BT, United Kingdom. Tel.: +44 0 20 759 46275; fax: +44 0 20 759 46274.

E-mail address: j.barria@imperial.ac.uk (J.A. Barria).

0167-9473/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.csda.2011.08.009

data at the same time. On the other hand, in online-processing methods, data items are processed as they become available or as they arrive.

The problem of estimating density functions for fixed-static data models has been thoroughly studied in the literature; see for example Scott (1992), Vannucci (1995) and Silverman (1998). However, for the case of data streams and within the framework of nonparametric estimators, only the following publications have suggested possible online solutions: (Wegman and Marchette, 2003; Procopiuc and Procopiuc, 2005; Heinz and Seeger, 2005; Wegman and Caudle, 2006; Heinz, 2007; Boedihardjo et al., 2008; Caudle and Wegman, 2009). Note that the density estimation problem in the context of *data streams* is particularly different from the case of *fixed static data* models in basically three fundamental aspects, all of them implicitly related to the data stream paradigm previously described. First, it considers a limited memory storage, which means that the estimator cannot recompute the entire density using all the data samples available every time new data items arrive. Second, the time-based order of data items is relevant in this context, which means that the temporal relation of the data should be considered by the estimator. Third, the estimation should be fast enough to update the density estimate before new data arrive. For the three reasons mentioned above, a feasible data streams density estimator should process the incoming data in a recursive fashion considering the addition/inclusion of new data as well as the discounting of old data.

Wavelet-based estimators belong to the class of orthogonal series estimators, a general class of nonparametric methods introduced in the literature by Céncov (1962), from which the two most representative and applied algorithms are based on Fourier and Hermite orthogonal basis functions. The most relevant characteristic of traditional orthogonal estimators is their computation simplicity. In contrast, their major drawback is their inability to estimate local properties of the underlying density, since they are based on global basis functions (i.e. Fourier and Hermite basis functions). Since wavelets are localised basis functions (in time and frequency), this problem is overcome with wavelet-based estimators, which allow local learning and local manipulation of the estimated density. Furthermore, wavelet-based estimators offer more flexibility in terms of convergence and smoothness due to the availability of several families of orthogonal wavelet functions that can be used in the estimator. In general, as Safavi et al. (2004) pointed out, since wavelet-based estimators inherit the advantages of wavelets and multiresolution analysis, they are superior to other orthogonal estimators.

Wavelet-based estimators have been intensively investigated in the literature; see for example Vannucci (1995), Donoho et al. (1996), Herrick et al. (2001) and Safavi et al. (2004), however, most of the techniques available in the literature are based on a batch-processing concept and just few algorithms, such as the ones investigated by Heinz and Seeger (2005), Wegman and Caudle (2006) and Caudle and Wegman (2009), are based on the online processing framework. Particularly, Wegman and Caudle (2006) and Caudle and Wegman (2009) proposed a recursive wavelet-based density estimator that consists of two main parts. First, an initial estimate of the underlying density is obtained by the use of traditional wavelet density estimation techniques. Second, this initial estimate is recursively updated as new data arrive considering both the addition of new data as well as an exponential discounting of the old one. Specifically, the key idea behind the updating procedure relies on the theoretical fundamental that, in orthogonal series estimators the coefficients can be approximated by the expectation of the projections of each data item over the orthogonal series. Additionally, in Caudle and Wegman (2009), a static block discounting is also proposed in which the parameter that controls the block discounting procedure is determined according to the degree of stationarity of the data using the Kolmogorov–Smirnov test (KS-test). Although this general technique is simple and effective, it cannot be completely useful in applications in which arriving blocks or subsets of data require the same level of emphasis/importance. Consider, for instance, applications such as environmental monitoring, where air quality standards consider exposition time to pollutants and "running" estimates are required for some specific periods of time (e.g.the last 8 and 24 h, depending on the pollutant). On the other hand, the algorithm presented by Heinz and Seeger (2005) is derived from the framework for maintaining nonparametric estimators over data streams initially proposed by Blohsfeld et al. (2005) whose main idea is the initial partition of the data stream into blocks, then for each block, an estimator is constructed. The next step is the merging of all these "block" estimators in an overall estimator. Finally, this overall estimator is compressed in order to assure the consumption of a constant amount of memory. The major drawback of this algorithm is that, data items cannot be processed as they arrive, in real time fashion, instead, they are handled block by block, with density estimates available only after each block is completed. Furthermore, this approach does not consider any discounting procedure for old data, and for that particular reason it is not potentially useful for non-stationary data streams.

A novel online density estimator for non-stationary streaming data is addressed in this paper. The proposed algorithm is derived from: (1) the well documented wavelet-based density estimation framework for fixed static data studied by Vannucci (1995), Vidakovic (1999) and Herrick et al. (2001), (2) the recursive estimator proposed by Wegman and Caudle (2006) and Caudle and Wegman (2009), and (3) data stream model concepts described by Babcock et al. (2002) and Golab and Ozsu (2003). The approach presented in this paper considers the online updating of the estimator as well as the selective reevaluation of its coefficients for new arriving data items. The proposed estimator is fundamentally different with respect to the methods reported by Wegman and Caudle (2006) and by Caudle and Wegman (2009) by the fact that, it is based on sliding window concepts suitable for non-stationary streaming data, instead of considering landmarks windows which are suitable for stationary cases. In that sense, the estimator reported here is a novel estimator whose estimation capabilities are particularly oriented towards density estimation for non-stationary streaming data.

The contribution of this paper is then twofold. First, the technique proposed for the updating of the estimator coefficients, which includes both the inclusion/addition of new data items as well as the discount of old information using the concept of sliding windows, has not been applied in this context before. The second contribution is the optimisation procedure for the selective reevaluation of the estimator coefficients. Note that none of the work published so far addressing the issue

of wavelet-based density estimation in data streams has considered the fundamental difference between batch and online processing in the selective reevaluation of wavelet coefficients. It is important to highlight that the computational cost of such evaluation could be substantially reduced by considering that some of the estimator coefficients/parameters remain unaltered from one timestamp to the next one. This consideration, within the framework of wavelet-based orthogonal estimators, is completely new. Results reported in this work clearly show that the proposed estimator outperforms the technique proposed by Caudle and Wegman (2009) in the adaptation capability for cases in which the underlying distribution is changing over time. This improved capability potentially allows the use of this particular density estimation algorithm in the context of non-stationary applications.

The rest of the paper is organised as follows. In Section 2, orthogonal wavelet decomposition is briefly reviewed from a very general point of view. Additionally, fundamental concepts behind orthogonal series estimators and wavelet-based density estimators for static data is also introduced in this section. Afterwards, the algorithm proposed for the online density estimation is presented in Section 3. Section 4 includes both simulated and real-world data experiments performed to evaluate the proposed framework and their corresponding results. Finally, in Section 5, the conclusions of this work are presented.

2. Wavelets and batch-based wavelet density estimation

2.1. Wavelets

Wavelet analysis is a well-established discipline that has been widely applied in a great variety of applications. Among the most relevant ones, we could cite: data compression, signal filtering and denoising, image processing, as well pattern recognition and system identification. The basic concept in wavelet transforms, as Bruce et al. (2002) emphasises, is the projection of data onto a basis of wavelet functions in order to separate fine-scale and large-scale information. Particularly, data are decomposed into a series of shifted and scaled versions of a mother wavelet function to make possible the analysis of signals at different scales and resolutions.

In this work, the discrete wavelet transform (DWT) is employed. Its main features can be summarised as follows: (1) data is separated into wavelet details coefficients (fine-scale information) and approximation coefficients (large-scale information) by the projection of the data onto an orthogonal basis system; (2) such approximation and wavelet coefficients include all the information of the original signal. In a formal way, based on the DWT framework, a signal $f(x)$ is decomposed in an approximation and details to form a multiresolution analysis of the signal as:

$$f(x) = \sum_k c_{j_0,k}\phi_{j_0,k}(x) + \sum_{j_0 \le j}\sum_k d_{j,k}\psi_{j,k}(x), \quad j_0, j, k \in \mathbb{Z}, \tag{1}$$

where $c_{j_0,k}$ denote the approximation coefficient at resolution j_0, $d_{j,k}$ denotes the wavelet coefficient at resolution j, $\phi_{j_0,k}(n)$ is a scaling function and $\psi_{j,k}(n)$ is a wavelet function at resolution j. The coefficients $c_{j_0,k}$ and $d_{j,k}$ are computed according to:

$$c_{j_0,k} = \left\langle f(x), \phi_{j_0,k}(x)\right\rangle, \quad j_0, k \in \mathbb{Z}, \tag{2}$$

$$d_{j,k} = \left\langle f(x), \psi_{j,k}(x)\right\rangle, \quad j, k \in \mathbb{Z}, \tag{3}$$

where the operator $\langle . \rangle$ denotes the inner product in the space of square integrable functions $L^2(\mathbb{R})$. The dyadic DWT assumes scaling functions ϕ and wavelet functions ψ of the form:

$$\phi_{j_0,k}(x) = 2^{j_0/2}\phi(2^{j_0}x - k), \quad j_0, k \in \mathbb{Z}, \tag{4}$$

$$\psi_{j,k}(x) = 2^{j/2}\psi(2^{j}x - k), \quad j, k \in \mathbb{Z}. \tag{5}$$

Particularly, the process of obtaining the approximation coefficients $c_{j_0,k}$ and wavelet coefficients $d_{j,k}$ is called the discrete wavelet transform, while the process of reconstructing the signal, given such coefficients is named the inverse discrete wavelet transform (IDWT). In practice, the dyadic DWT can be implemented in a computationally efficient manner via the dyadic filter tree algorithm proposed by Mallat (1989), that is also referred as Mallat's cascade algorithm. The basic idea behind this fast algorithm is to represent the wavelet basis as a set of high-pass and low-pass filters in a filter bank (Percival and Walden, 2000).

2.2. Orthogonal series density estimators

In simple words, density estimation is the problem of constructing and estimating a probability density function from some given observed data. Density estimation is a well studied problem for which several solutions have been reported in the literature. Density estimation techniques can be classified into two main groups: parametric and nonparametric approaches. Among the latter, kernel estimators, orthogonal estimators and histograms have been the focus of the attention of the majority of the research community. Wavelet density estimators fall into the class of orthogonal series estimators and for the sake of completeness, in this section, some fundamental theoretical background of orthogonal series estimators is briefly reviewed, further details can be found in Vidakovic (1999).

According to Céncov (1962), an unknown square integrable density function can be expressed as a convergent series of orthogonal basis functions:

$$f(x) = \sum_j b_j \psi_j(x), \quad j \in \mathcal{J}, \tag{6}$$

where ψ_j is a complete orthonormal system of basis functions in $L^2(\mathbb{R})$, b_j is the coefficient of the jth basis function and $\mathcal{J}$ is an appropriate set of indices that belongs to $\mathbb{Z}$. In that context, and considering that $f(x)$ is a density function, Cêncov found that the coefficient b_j can be expressed as an expectation:

$$b_j = \langle f, b_j \rangle = \int \psi_j(x) f(x) \mathrm{d}x = E[\psi_j(X)], \quad j \in \mathcal{J}. \tag{7}$$

Consequently, if $X_1, X_2, \ldots, X_n$ are the realisations of a random variable X with an unknown square integrable density $f(x)$, the jth series coefficient in an orthogonal series estimator can be elegantly approximated by:

$$\hat{b}_j = \frac{1}{n} \sum_{i=0}^{n} \psi_j(X_i), \tag{8}$$

and the corresponding density by

$$\hat{f}(x) = \sum_j \hat{b}_j \psi_j(x), \quad j \in \mathbb{Z}. \tag{9}$$

2.3. Wavelet-based orthogonal series estimators (batch-processing approach)

Since wavelets are also orthogonal basis functions, wavelet density estimators follow the same concepts described in Eqs. (8) and (9), however, within the wavelet framework, the density can be represented as an orthogonal series of two different basis functions: scaling functions ϕ and wavelet functions ψ. If $X_1, X_2, \ldots, X_n$ are the realisations of a random variable X with an unknown square integrable density $f(x)$, then the scaling and wavelet coefficients can be approximated by

$$\hat{c}_{j_0,k} = \frac{1}{n} \sum_{i=0}^{n} \phi_{j_0,k}(X_i), \quad j_0, k \in \mathbb{Z}, \tag{10}$$

$$\hat{d}_{j,k} = \frac{1}{n} \sum_{i=0}^{n} \psi_{j,k}(X_i), \quad j, k \in \mathbb{Z}, \tag{11}$$

where $\hat{c}_{j_0,k}$ and $\hat{d}_{j,k}$ are the approximated or *empirical* coefficients for the scaling and wavelet functions, respectively. Following the same concept of Eq. (9), the wavelet density estimate is thus:

$$\hat{f}(x) = \sum_k \hat{c}_{j_0,k} \phi_{j_0,k}(x) + \sum_{j=j_0}^{j_0+J} \sum_k \hat{d}_{j,k} \psi_{j,k}(x) \quad j_0, j, k, J \in \mathbb{Z} \tag{12}$$

where j_0 indicates the coarsest scale or the lowest resolution of analysis, and J refers to the number of decomposition levels.

The rate of convergence of orthogonal series estimators has been shown to be asymptotically optimal (see Hall, 1986 for further details). Additionally, since wavelet basis functions possess good localisation properties, wavelet-based orthogonal series also have improved local approximation capabilities. Moreover, the precision of this type of estimators depends on three aspects: the shape of the density to be estimated, the number of data items considered for the estimation as well as the number of the decomposition levels in Eq. (12).

In the corresponding literature, different variants of the general estimator described by Eqs. (10)–(12) can be found (e.g. Masry, 1994; Donoho et al., 1996; Donoho and Johnstone, 1998; Herrick et al., 2001). Specifically, they mainly differ in two basic aspects. The first is the strategy that they follow to select which of the terms in the series expansion should be kept, and the second is the way the series coefficients are thresholded. By applying those strategies, the construction of linear and nonlinear estimators is possible: the former by keeping wavelet coefficients untouched and the latter with soft or hard coefficients thresholding strategies. Remember that, all these approaches are based on a fixed-static data model and consequently they address the density estimation problem from a batch processing perspective. Furthermore, since wavelets are not a positive δ-sequence, density estimates may take negative values in regions where the sample is sparse. Different solutions for this problem have been reported in the literature (i.e. taking the square root of the density Pinheiro and Vidakovic, 1997, using non-negative wavelets). A discussion on these procedures is outside of the scope of this paper. Details can be found in Vidakovic (1999).

Practical implementation

The practical implementation of the wavelet density estimator of Eqs. (10)–(12) makes use of two well known algorithms: (a) the recursive algorithm introduced by Daubechies and Lagarias (1992), and (b) Mallat's cascade algorithm (Mallat, 1989).

The Daubechies–Lagarias algorithm is a numerical method for the calculation of the scaling and wavelet values at a given point with a predefined precision. This algorithm is necessary since, for most of the compactly supported wavelet families, both scaling functions ϕ and wavelet functions ψ have no explicit or closed form representation. On the other hand, Mallat's cascade algorithm is a fast and optimised procedure for the implementation of the DWT based on the use of filter banks and nested spaces. A deep description of those procedures is not necessary for the understanding of the algorithm proposed in this work. For more details about these two procedures, the reader is referred to the original sources as well as to Vidakovic (1999).

In broad terms, the implementation of the wavelet-based estimator is performed by choosing in (12) the coarsest resolution j_0 and the number of decomposition levels J, then the scaling coefficients $\hat{c}_{j_0,k}$ and the wavelet coefficients $\hat{d}_{j,k}$ for $j \in \{j_0, j_0+1, \ldots, j_0+J\}$ can be obtained by the use of Mallat's cascade algorithm starting from the high resolution scaling coefficients $\hat{c}_{J,k}$, first obtained by applying the Daubechies–Lagarias algorithm over the original data.

3. Proposed online wavelet-based orthogonal series estimator

The recursive approach proposed in this paper is based on a sliding window that moves forward, replacing old items as new data items arrive. Note that, according to Babcock et al. (2002), sliding windows are a natural method for the analysis of data streams with the specific property of emphasising recent data; moreover, they are particularly useful in situations in which an excerpt of the stream is of interest at any given time (e.g. running hourly and daily data) Golab and Ozsu (2003).

The algorithm proposed considers the online updating of the density estimator as well as the selective reevaluation of only the meaningful estimator parameters for each arriving data item. For the case of the online implementation of the estimator, it can be divided in two major stages. First, an initial estimation of the density is computed using Eq. (12). Second, once the initial estimate has been obtained, the density is continuously updated in a recursive fashion as new data items arrive. In a formal way consider that, in streaming situations, the statistical and probabilistic structural properties of the data evolve over time. In that context, assume that $X_1, X_2, \ldots$ is an unbounded continuous sequence of data items, where X_k arrives before the data element X_{k+1} and in which the data elements are assumed to be i.i.d. random samples from an unknown probability density function $f(x)$ that evolves over time. For the initial estimation of the density, consider that the timestamp is initialised to $n = w$ and an initial estimate $\hat{f}_w(x)$ of $f(x)$ is computed in a batch fashion using Eq. (12) and considering the first w data items $X_1, X_2, \ldots, X_w$ covered by the sliding window of size w. Then, in the updating stage, the timestamp is increased by one to $n = w+1$ and the window is moved one step forward, as a new data item arrives. Here, the initial estimate is updated to obtain the estimate $\hat{f}_{w+1}(x)$ using the elements $X_2, \ldots, X_{w+1}$. Continuing in the same manner, and generalising, at a particular timestamp n an updated estimate $\hat{f}_n(x)$ of the density is obtained considering the sequence $X_{n-w+1}, \ldots, X_n$ of data items. Note that in its fundamental formulation, the procedure just described considers a window moving every time a new data item arrives but it could be easily modified and generalised in order to work in situations in which the obtention of density estimates is required after the arrival of a particular number of data items.

3.1. Recursive updating procedure

The updating of the estimator is performed by means of an iterative updating process of wavelet coefficients. In that sense, the framework proposed is inspired by the work initially reported by Wegman and Marchette (2003), which suggested a recursive method for updating kernel density estimators. This method was posteriorly adapted by Wegman and Caudle (2006) and Caudle and Wegman (2009) for its use in wavelet estimators.

3.1.1. Caudle and Wegman's approach

Specifically, Caudle and Wegman (2009) noted that, in the context of wavelet estimators, the estimator coefficients are approximated by taking the average of the basis functions evaluated at each data point according to Eqs. (10) and (11). Then, by considering the recursive implementation of that average, the orthogonal series coefficients can be iteratively updated, as new data items arrive, according to the equation:

$$\hat{b}_{j,n+1} = \frac{n}{n+1}\hat{b}_{j,n} + \frac{\varphi_j(X_{n+1})}{n+1}, \quad j, n \in \mathbb{N}, \tag{13}$$

where $\varphi_j(X_{n+1})$ either refers to a wavelet $\psi_{j,k}(x)$ or a scaling $\phi_{j,k}(x)$ functions and consequently $\hat{b}_{j,n}$ and $\hat{b}_{j,n+1}$ refer to their corresponding coefficients at timestamps n and $n+1$, respectively. Note that Eq. (13) is used in the context of landmark windows (windows with one fixed endpoint and one moving endpoint; see Golab and Ozsu, 2003). Additionally, if in Eq. (13) the term $\theta = n/(n+1)$ is considered, then (13) can be simply expressed as $\hat{b}_{j,n+1} = \theta\hat{b}_{j,n} + (1-\theta)\varphi_j(X_{n+1})$. Then, by choosing different values of θ, different weighting schemes can be implemented in order to adjust the emphasis or importance of new data with respect to the old one. It is important to notice that such weighting scheme is not able to assign the same level of importance to a specific subset of the arriving streaming data (see Fig. 1(a)), and as a result, it is not particularly suitable for applications which require density estimates for specific blocks of data, for instance, consider the case of environmental monitoring, in which hourly estimations are continuously needed. Moreover, results reported in Section 4 show that this

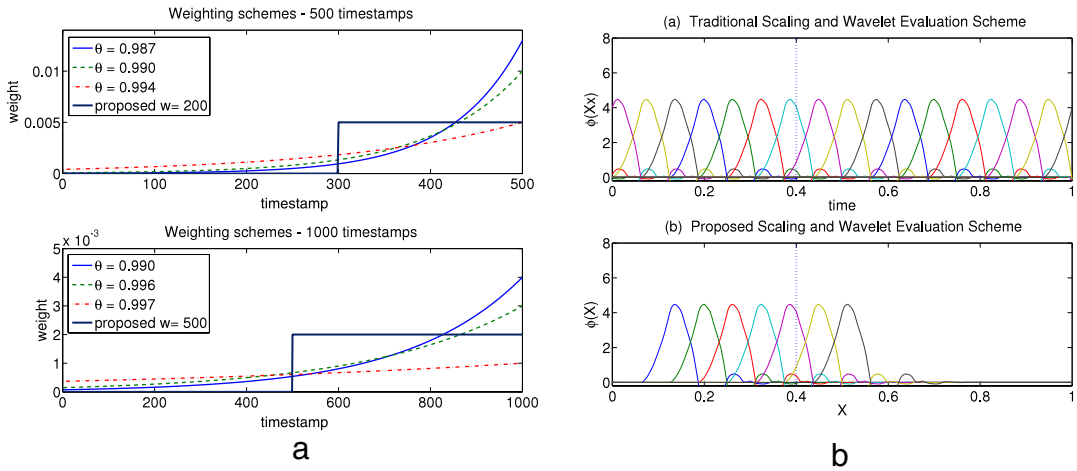


Fig. 1. (a) Weighting schemes and (b) scaling function evaluation for both the estimator of Caudle and Wegman (2009) and for the proposed in this work.

scheme does not have good adaptation capabilities in the context of non-stationary scenarios, which means that, if the underlying density is moderately changing over time, then the estimator will not able to perform the fast tracking of those changes.

3.1.2. Proposed approach

The recognition of the abovementioned difficulties motivated the updating scheme suggested in Caudle et al. (2011) which makes use of the concept behind Eq. (13) now applied in the context of sliding windows. Furthermore, in the proposed algorithm, Eq. (13) is adapted to be employed for both, the recursive addition of new data as well as the discounting of the old one. In its most basic form, and following the notation used in Eq. (13), the proposed updating algorithm can be expressed by the following equation:

$$\hat{b}_{j,n+1} = \hat{b}_{j,n} + \frac{\varphi_j(X_{n+1})}{w} - \frac{\varphi_j(X_{n-w+1})}{w}, \quad j, n, w \in \mathbb{N}, \tag{14}$$

where w is the window size and n denotes the timestamp. Here, X_{n+1} is the more recent data item arrived, and X_{n-w+1} expresses the oldest data item covered by the sliding window. Note that the first part of Eq. (14), related to the inclusion of new data items, is similar to the one proposed in (13), but in addition there is a third term that discounts old data. The basic idea of the proposed updating procedure is then twofold: (a) to incorporate into the approximated wavelet coefficients, and consequently into the density estimate in general, the contribution of new data items covered by the sliding window; and (b) to remove from the estimator coefficients calculation the old data items not any longer falling within the range of the sliding window. In its simplest form, Eq. (14) looks similar to (13). However, it is fundamentally different in its weighting scheme, which, as it was previously mentioned, assigns the same level of emphasis or importance to the data covered by the sliding window. This novel assigning method results in improved adaptation capabilities of the estimator. Furthermore, to make the proposed method suitable for real-time applications, in this paper, we additionally present a novel optimisation methodology not addressed before in the literature that considers the selective reevaluation of wavelet and scaling coefficients for arriving data items.

3.2. Selective reevaluation of wavelet coefficients

In this subsection, the proposed updating procedure previously presented is slightly modified in order to reduce the number of basis functions (scaling and wavelet functions) evaluated by the algorithm every time a new data item arrives. The aim of this modification is the optimisation of the computing resources used by the algorithm represented by Eq. (14), for this reason, the suggested strategy is basically the selective reevaluation/recalculation of just the relevant scaling and wavelet coefficients (coefficients that change at each particular time step) for every new processed data item. This modification is justified by the fact that, most of the series coefficients $\hat{b}_{j,n+1}$'s in Eq. (14) remain unaltered from time n to time $n+1$, namely, most of them are equal to $\hat{b}_{j,n}$'s. Taking into consideration this particular circumstance, inherent to the online estimator, Eq. (14) can be modified in order to evaluate just the scaling and wavelet functions that change for each timestamp.

Since the selective evaluation method proposed here is based on the effective support of scaling and wavelet functions at a given particular resolution, the formulation described below applies for compactly supported wavelet functions whose support is $2N - 1$ (e.g. Daubechies and Symlet families) where N is the order of the function. Note that the procedure can be generalised for other families of compactly supported wavelet functions by considering their corresponding support.

In a formal way, let ϕ and ψ be the scaling and wavelet functions of the Daubechies or Symlet families of order N whose support at scale $j = 0$ is $[0, 2N - 1]$. Note that, in a strictly theoretical way, the support of the wavelet function ψ is $[-N, N - 1]$. However, for its practical implementation it is normally assumed to be the same as its corresponding scaling function, that is $[0, 2N - 1]$; see Burrus et al. (1997) and Boggess and Narcowich (2009) for details. In this work, after following that practical consideration, the support of both ϕ and ψ can be generalised, within the dyadic wavelet framework, to any particular resolution j and translation k according to the following expression (Vannucci, 1995):

$$\mathbf{supp}\,\phi_{j,k} = \mathbf{supp}\,\psi_{j,k} = [2^{-j}k, 2^{-j}k + 2^{-j}(2N-1)] = [2^{-j}k, 2^{-j}(k + 2N - 1)]. \tag{15}$$

For convenience, it is assumed that the random variable X of interest is supported on $[0, 1]$. Note that in most of the problems involving density estimation, the random variable X of interest is compactly supported, that is, explicitly defined over a specific interval. For applications in which wavelet analysis is confined to a specific interval, a common practice to fix notations and simplify formulations is to assume data to be in the range $[0, 1]$. In that case, the wavelet basis for $L^2(\mathbb{R})$ is restricted to a basis for $L^2([0, 1])$ (Cohen et al., 1993).

Considering Eq. (15) and taking into account the practical implementation aspects of traditional batch-processing wavelet decomposition discussed in the previous section, Eq. (12) can be reduced to:

$$\hat{f}(x) = \sum_{k=-(2N-1)}^{2^{j_0}} \hat{c}_{j_0,k}\phi_{j_0,k}(x) + \sum_{j=j_0}^{j_0+J} \sum_{k=-(2N-1)}^{2^j} \hat{d}_{j,k}\psi_{j,k}(x) \quad j_0, j, J, k \in \mathbb{Z} \tag{16}$$

where j_0 is the lowest or coarsest resolution while J is the number of decomposition levels.

Then, according to Eq. (16), it is clear that, batch-processing wavelet estimators evaluate, at resolution j, $2^j + 2N$, basis functions (of either scaling or wavelet functions), each of them corresponding to a particular translation k, namely, $k = -(2N-1), \ldots, 0, \ldots, 2^j$. In that context, Eqs. (13) and (14) update the approximated coefficients, $\hat{b}_j$'s, $2^j + 2N$ times for each decomposition level j every time a new data item arrives by projecting them over each orthogonal base employed.

The selective reevaluation method proposed here modifies Eq. (14) in order to reevaluate at any particular timestamp only the estimator coefficients whose associated scaling and wavelet functions include the arriving data item within their support. For the opposite case, in which the arriving data item is outside of the support of the basis functions evaluated, their corresponding coefficients remain unaltered. The final expression for the updating procedure proposed is expressed in Eqs. (17)–(19). Notice that, in order to be clearer in the presentation of the selective reevaluation procedure, Eq. (17) has been split in two equations, one for the addition of new data (18) and other one for the discounting of the old one (19).

$$\hat{b}_{j,k,n+1} = \hat{b}_{j,k,n} + \hat{b}^{\text{addition}}_{j,k,n+1} - \hat{b}^{\text{discounting}}_{j,k,n+1}, \tag{17}$$

with:

$$\hat{b}^{\text{addition}}_{j,k,n+1} = \begin{cases} \dfrac{\varphi_{j,k}(X_{n+1})}{w} & \text{if } 2^j k \le X_{n+1} \le 2^j(k + 2N - 1), \\ 0 & \text{otherwise}, \end{cases} \tag{18}$$

$$\hat{b}^{\text{discounting}}_{j,k,n+1} = \begin{cases} \dfrac{\varphi_{j,k}(X_{n-w+1})}{w} & \text{if } 2^j k \le X_{n-w+1} \le 2^j(k + 2N - 1), \\ 0 & \text{otherwise}, \end{cases} \tag{19}$$

where $\varphi_{j,k}$ is used to refer to either the wavelet $\psi_{j,k}(x)$ or the scaling function $\phi_{j,k}(x)$ and $\hat{b}_{j,k,n}$ refer to their corresponding coefficient for timestamp n. Additionally, w is the window size and X_{n+1} and X_{n-w+1} are the most recent and oldest data items covered by the sliding window. Note that, since each basis function is defined by both the resolution j and the translation parameter k, the index k has been included. Furthermore, the proposed selective strategy for updating the coefficients is valid for both, Daubechies–Lagarias algorithm and Mallat's cascade algorithm which are the algorithms involved in the practical implementation of Eq. (12).

An important additional observation is that the proposed estimator does not require an initial estimate computed using Eq. (12), it can be simply initialised by assuming that each term of the set $\{X_{0-w-1}, \ldots, X_0\}$ is equal to zero and $\hat{b}_{j,k,0} = 0$. Pseudo-codes for the three wavelet-based estimators are shown in Algorithms 3.1–3.3. Specifically, Algorithm 3.1 shows the pseudo-code for the traditional batch estimators and is related to Eq. (16). Algorithm 3.2 corresponds to the estimator proposed by Caudle and Wegman (2009) and is based on Eq. (13) but in addition, it includes the discounting strategy explained in Section 3.1.1. Note that in Eq. (13) the translation parameter k is implicit, however in Algorithm 3.2 it is used. Algorithm 3.3 is the proposed estimator of Eqs. (17)–(19).

3.3. Implementation issues

The proposed updating procedure uses a fixed amount of memory with a buffer size depending on the number of data items covered by the sliding window no matter whether the data stream is unbounded in size. Regarding theoretical

criteria for the selection of an appropriate window size, note that the window size is application specific and it should be selected according to stochastic properties of the data and the specific requirements of the application (e.g. for time-based applications, it could consider specific minutes, hour or days).

In this paper, three different alternatives for the selection of the window size parameter are suggested. The first one considers the use of goodness-of-fit tests thoroughly investigated by Stephens (1965, 1970) and more recently by Tygert (2010), which are based on determining whether a given set of i.i.d. data points come from a specified probability density function. Within this framework, the most common tests used are the Kolmogorov–Smirnov and Kuiper's tests that focus on the size of the discrepancy between the cumulative distribution function of a specified probability density function and the empirical cumulative distribution function of i.i.d. data.

Specifically, as it is explained by Rohatgi and Saleh (1976), Kuiper's statistic is a variation of the well-known Kolmogorov–Smirnov statistic and is defined by:

$$V = D^+ + D^- = \max_x(\hat{F}(x) - F(x)) + \max_x(F(x) - \hat{F}(x)), \tag{20}$$

where F and $\hat{F}$ are the two cumulative distribution functions involved, specifically, F is the true distribution while $\hat{F}$ is the estimated one. In Eq. (20), D^+ and D^- are discrepancy statistics representing the maximum deviation above and below the two cumulative distribution functions being compared. Here, the window size may be selected according to a specified value of Kuiper's statistic or according to the results obtained from a hypothesis testing operation based on such statistic.

The second alternative comes from the work in the context of kernel density estimation in which the expected L_2 risk function defined by the mean integrated squared error (MISE) (Marron and Wand, 1992), is used as the optimality criterion for the selection of the bandwidth parameter. Hall and Patil (1995) investigated the properties of MISE in nonlinear, thresholded, wavelet-based density estimators and derived a formula to describe an asymptotically optimal empirical bandwidth selection rule. MISE is defined by the following equation:

$$\mathrm{MISE}(h) = E\int(\hat{F}_n(x;h) - F(x))^2\mathrm{d}x, \tag{21}$$

where n is the sample size, h is the bandwidth of the estimator. Note that the sample size n parameter can be interpreted, in our framework, as the window size, since it indicates the number of samples available (within the sliding window) to estimate the underlying density.

The third criterion suggested is useful for cases in which the underlying density is completely unknown, and then a possible solution might be approximating the empirical distribution function of the data with polynomials to a certain precision or error and then establishing a relationship between the polynomial degree obtained and the window size required to fit a density function with a smoothness defined by that specific polynomial degree. Future work is intended to assess a deep investigation of this particular issue in order to introduce a systematic procedure for the selection of this free parameter.

Algorithm 3.1: Batch-based Wavelet Density Estimator $(j_0, J, w, \mathbf{X}_n = \{X_1, \ldots, X_n\})$

```
for n ← 0 to w − 1
do { for j ← j_0 to j_0 + J
     do { for k ← −(2N − 1) to 2^j
          do { if n = 0
               then { b̂_{j,k,n} ← 0
                      comment: Initialise to zero the basis function's coefficient
               b̂_{j,k,n+1} ← b̂_{j,k,n} + φ_{j,k}(X_n)
               comment: Then update the corresponding basis function's coefficient
```

Algorithm 3.2: Recursive Wavelet Density Estimator (caudle) (j_0, J, n, X_{n+1})

```
for j ← j_0 to j_0 + J
do { for k ← −(2N − 1) to 2^j
     do { b̂_{j,k,n+1} ← (n/(n+1)) b̂_{j,k,n} + (1 − n/(n+1)) φ_{j,k}(X_{n+1})
          comment: For each new arriving data point evaluate all the basis functions
                   and update the corresponding coefficients
```

Algorithm 3.3: PROPOSED RECURSIVE WAVELET DENSITY ESTIMATOR $(j_0, J, w, n, X_{n+1}, X_{n-w+1})$

for $j \leftarrow j_0$ **to** $j_0 + J$
do
  for $k \leftarrow -(2N-1)$ **to** 2^j
  do
    if $X_{n+1} \geq 2^j k$ **and** $X_{n+1} \leq 2^j(k+2N-1)$
    comment: If the new arriving data point X_{n+1} falls within the support of the basis function
    then $\hat{b}^{\text{addition}}_{j,k,n+1} \leftarrow \frac{\varphi_{j,k}(X_{n+1})}{w}$
    comment: Then update the corresponding basis function's coefficient
    else $b^{\text{addition}}_{j,k,n+1} \leftarrow 0$
    if $X_{n-w+1} \geq 2^j k$ **and** $X_{n-w+1} \leq 2^j(k+2N-1)$
    comment: If the oldest data point X_{n-w+1} falls within the support of the basis function
    then $\hat{b}^{\text{discounting}}_{j,k,n+1} \leftarrow \frac{\varphi_{j,k}(X_{n-w+1})}{w}$
    comment: Then update the corresponding basis function's coefficient
    else $\hat{b}^{\text{discounting}}_{j,k,n+1} \leftarrow 0$
    $\hat{b}_{j,k,n+1} \leftarrow \hat{b}^{\text{addition}}_{j,k,n+1} - \hat{b}^{\text{discounting}}_{j,k,n+1}$

Regarding the computational complexity of the algorithm, since Eq. (12) is based on Mallat's algorithm, the number of computational operations (floating-point multiplications and additions) is linear with the length of the signal M, with a relatively small constant of linearity. It can be formally expressed in the following way. Let C_{DWT} be the complexity of a DWT of a data signal of length M, and considering that, after each scale the algorithm only operates on half of the output data, then it can be shown that $C_{\text{DWT}} = O(M) + C_{\text{DWT}}(M/2)$ which gives rise to the solution $C_{\text{DWT}} = O(M)$ (see Burrus et al., 1997 for details). In the proposed selective evaluation procedure, the reduction in the number of computations will depend on the wavelet basis employed and the support of the corresponding scaling and wavelet functions. For the case of orthogonal compactly supported wavelets (i.e. Daubechies, Symlets) whose support length is $2N-1$ (with N denoting the order of the wavelet or scaling function) the number of orthogonal basis evaluated is always equal to $2N-1$ for every level of decomposition j. This means that the computational complexity of the estimator is constant at each level of decomposition no matter whether a high number of orthogonal basis is used at that particular resolution.

In order to exemplify the method proposed, suppose that it is needed to evaluate the estimator of Eq. (16), for the data point $X = 0.4$ only using Symlet scaling functions of order $N = 4$ with a resolution $j = 5$. Note that, since we are interested in scaling functions, then only the first term of Eq. (16) is required, hence $j = j_0$. Batch-based algorithms evaluate $2^j + 2N = 40$ basis functions with the corresponding translation parameters $k = -(2N-1), \ldots, 2^j = \{-7, -5, \ldots, 32\}$ even when the data point to be evaluated is outside the support of the majority of the translated wavelets evaluated. In contrast, the algorithm proposed here is able to focus the computation only on all those basis functions whose translation parameters are meaningful, which are those whose coefficients are different from zero. For example, according to Eq. (18), they are $k = \{6, 7, \ldots, 12\}$. Note that for $k = 6$ and $k = 12$, the associated supports which are, according to Eq. (15), equal to $[2^{-5}(6), 2^{-5}(6 + 2(4) - 1)] = [0.1875, 0.4063]$ and $[2^{-5}(12), 2^{-5}(12 + 2(4) - 1)] = [0.3750, 0.5938]$, respectively, include the data point $X = 0.4$. Supports for scaling function with translation parameters $k = 5$ and $k = 13$ are $[2^{-5}(5), 2^{-5}(5 + 2(4) - 1)] = [0.1563, 0.3750]$ and $[2^{-5}(13), 2^{-5}(13 + 2(4) - 1)] = [0.4063, 0.6250]$, which in contrast do not include the data point we are interested in, i.e., $X = 0.4$. The proposed strategy for the evaluation of scaling and wavelet coefficients avoids wasting computing resources on the evaluation of the remaining 33 wavelets, $k = \{-7, \ldots, 5, 13, \ldots, 32\}$, whose corresponding coefficients will be zero after the evaluation. The difference between traditional coefficients evaluation scheme and the proposed procedure can be clearly observed in Fig. 1(b). The generalisation of the proposed estimator to higher dimensions can be found in the Appendix.

Concerning improvements in computational speed, the proposed estimator offers a notable speed advantage for problems involving medium and large window sizes, as can be noted in Fig. 2. The computer system that was used to generate the result reported in Fig. 2 was an Intel Core2 Duo CPU E6550, 2 GB of RAM, running on windows XP and the simulation environment is MATLAB R2010b. Specifically, we define the computation time ratio (used in Fig. 2(c)) as the number of times that the proposed algorithm is faster that its batch processing counterpart that is the computational time for batch algorithm divided by the computational time of the proposed approach. To exemplify the speed improvements consider, for instance, that a given application requires the online density estimation with a window size of 10,000 data samples, then, according to Fig. 2(c), the proposed estimator is approximately 5000 times faster than its batch-processing estimator counterpart. Note that, from the perspective of computational time, the approach reported by Caudle and Wegman (2009) is still a batch-based estimator that evaluates at each timestamp all the estimator coefficients even when this is done using a recursive scheme.

Note that the computation time ratio between the proposed technique against the batch-based approach increases exponentially as the window size increases. Also note that computation time of the proposed estimator is independent

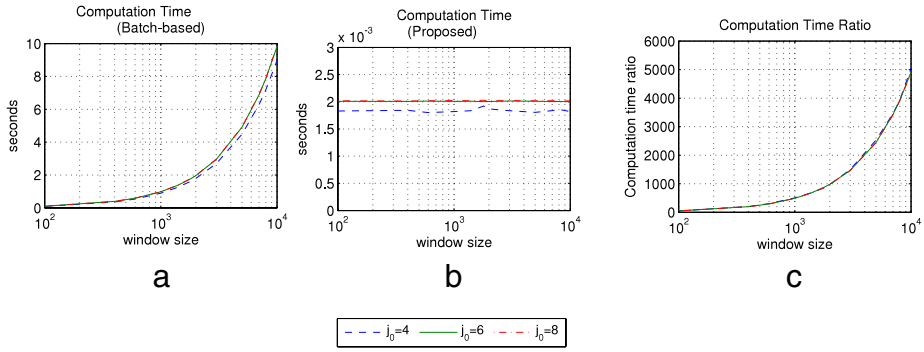


Fig. 2. Computation time for batch (a) and proposed (b) estimators and computation time ratio (c).

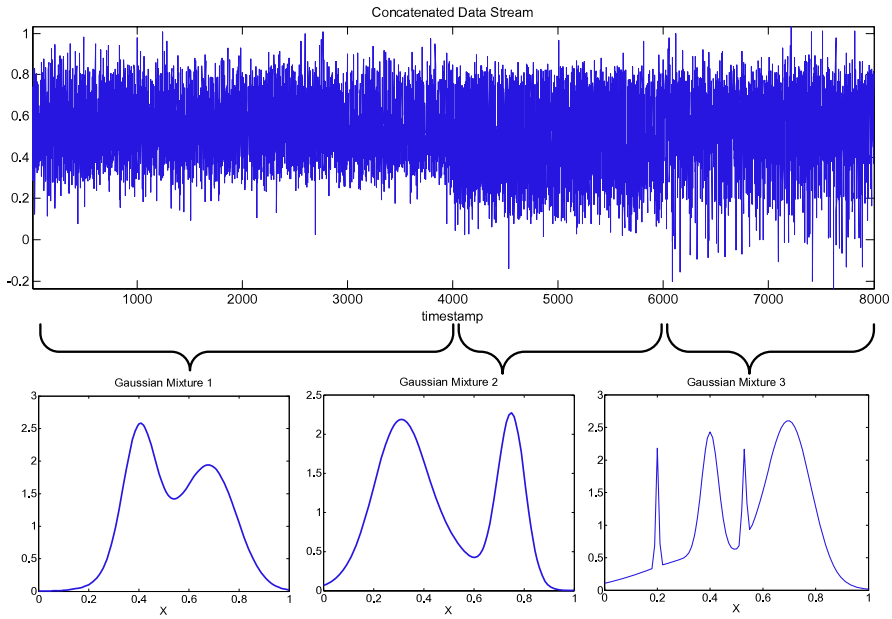


Fig. 3. Concatenated data stream used for the evaluation.

of the window size, since it only involves the evaluation of the newest and oldest data points covered by the sliding window.

4. Algorithm assessment

4.1. Simulation experiment

In order to compare the performance of the proposed online density estimator, initially we construct simulated streaming data using three different mixture distributions considering 4000, 2000 and 2000 samples from them, respectively. Note that, since the benchmark estimator (the approach proposed by Caudle and Wegman, 2009) used for performance comparisons requires an initial estimate of the underlying density, it is necessary to increase the number of timestamps evaluated for the first mixture distribution. One simulated data stream with 8000 samples is constructed by concatenating samples from the three mixture distribution (see Fig. 3). Then, seven scenarios are evaluated considering windows sizes of 200, 400, 800, 1000, 1200, 1600, and 2000 data items. Table 1 shows the three different mixture distributions used in the simulations and its corresponding timestamps in the concatenated data stream.

For the seven scenarios, the experiment consists of online density estimation of the concatenated data stream and the evaluation of the mean square error of the estimates at three particular timestamps (i.e. 4000, 6000, and 8000) in which the sliding window is covering samples from just one of the underlying distributions. The experiment is repeated 1000 times and the average of the mean square error (MSE) between the true underlying density and the estimated one for the 1000 cases is recorded. The averaged MSE's for the 1000 experiments at the selected timestamps are shown in Tables 2 and 3, for the estimator of Caudle and Wegman (2009) and for the algorithm proposed in this work, respectively. For the evaluation,

Table 1
Gaussian mixtures for the two scenarios evaluated.

	Timestamp	Gaussian mixture (with $N(\mu, \sigma^2)$)
1	1–4000	$N(0.4, 0.004)$, $N(0.5, 0.02)$ and $N(0.7, 0.01)$ with mixture parameters 0.3, 0.3, 0.4, respectively
2	4001–6000	$N(0.3, 0.01)$, $N(0.4, 0.03)$ and $N(0.75, 0.003)$ with mixture parameters 0.4, 0.3, 0.3, respectively
3	6001–8000	$N(0.4, 0.05)$, $N(0.4, 0.001)$, $N(0.2, 0.00003)$, $N(0.53, 0.00005)$ and $N(0.7, 0.007)$ with mixture parameters 0.3, 0.15, 0.025, 0.025 and 0.5 respectively

Table 2
Average of the MSE for the 1000 experiments for the estimator proposed in Caudle and Wegman (2009).

Evaluation results	θ value						
Timestamp	0.990	0.993	0.995	0.997	0.999	0.9995	0.9999
4000	5.05×10^{-8}	4.67×10^{-8}	4.36×10^{-8}	3.90×10^{-8}	3.37×10^{-8}	3.71×10^{-8}	5.26×10^{-8}
6000	39.9×10^{-8}	38.9×10^{-8}	39.4×10^{-8}	43.0×10^{-8}	64.7×10^{-8}	92.7×10^{-8}	185×10^{-8}
8000	137×10^{-8}	140×10^{-8}	143×10^{-8}	144×10^{-8}	133×10^{-8}	118×10^{-8}	93.5×10^{-8}

Table 3
Average of the MSE for the 1000 experiments for the proposed estimator.

Evaluation results	Window size						
Timestamp	200	400	800	1000	1200	1600	2000
4000	27.6×10^{-8}	14.2×10^{-8}	7.76×10^{-8}	6.44×10^{-8}	5.46×10^{-8}	4.35×10^{-8}	3.62×10^{-8}
6000	28.8×10^{-8}	15.5×10^{-8}	8.39×10^{-8}	7.06×10^{-8}	6.16×10^{-8}	5.03×10^{-8}	4.37×10^{-8}
8000	43.6×10^{-8}	28.8×10^{-8}	22.0×10^{-8}	20.5×10^{-8}	19.5×10^{-8}	18.3×10^{-8}	17.6×10^{-8}

the wavelet used in the estimator is the *daubechies* 4 (*db*4) from the family of orthogonal Daubechies wavelets, with the initial resolution set to $j_0 = 4$ and the number of decomposition levels $J = 0$.

As it is shown in Tables 2 and 3, for the estimates at timestamps 6000 and 8000 for the seven cases, the proposed approach outperforms the method suggested by Caudle and Wegman (2009). These results led us to conclude that the proposed algorithm has better estimation capabilities in non-stationary contexts in which the underlying density is changing over time. This result is supported by the fact that the proposed algorithm is based on sliding window concepts which allow full control of the emphasis/importance assigned to recent data, and as a consequence, its corresponding estimations are suitable to track changes in the underlying density function. In contrast, the method reported by Caudle and Wegman (2009), which is based on landmark windows, cannot properly manage the importance assigned to new data. The second observation is that, for the proposed approach, the error in the estimation decreases as the size of the window increases, due to a higher number of samples that are considered in the computation of the estimate. Also note that, in (Table 2), the estimator suggested by Caudle and Wegman (2009) presents the lowest mean square error for the timestamp 4000 with $\theta = 0.999$. The reason for this is that more data items are available for the estimation and Caudle's estimator, which is based on an exponential formulation, can include all of them for the computation of the estimate. In contrast, the number of data items used in the proposed sliding window approach is always fixed, with a maximum size of 2000 items for this example. It is important to note that, since an initial estimation based on a batch processing concept is required for both algorithms, the first mixture distribution consisted of 4000 data items, 2000 more than the second and third mixtures. Fig. 4(a) and (b) graphically show the results of one of the experiments, for the method reported by Caudle and Wegman (2009) and for the one introduced in this work, respectively. Additionally, it is important to take a closer look at the estimates when the sliding window is at the transition from a given density to another. This fact is important because it is directly related to the ability of the estimator to track changes in the underlying density. In Fig. 4(a) and (b), the density estimates reported for timestamps 6000 and 8000 clearly show the improved tracking capability of the reported approach, which adapts faster to the second and third underlying densities.

In order to evaluate, at any particular timestamp, the density estimates similarity/discrepancy with respect to the true underlying density or densities involved, Kuiper's statistic is analysed. Note that, in this context, a larger value of Kuiper's statistic V indicates larger discrepancy.

The experiment then involved obtaining Kuiper's statistic between the estimated density and the three mixture distributions of Table 1, for timestamps 2000–8000, and for both, the method suggested by Caudle and Wegman (2009) and the proposed estimator. The corresponding results are depicted in Fig. 5(a) and (b), respectively, in which resulting estimations are compared to the ideal situation that considers Kuiper's statistic among the underlying mixture distributions of Table 1.

Results from Tables 2 and 3 are confirmed by Fig. 5(a) and (b). Note that, for the method proposed, Kuiper's statistics for the three window sizes plotted (Fig. 5) are more sensitive with respect to the changes in the three underlying mixture distributions and, in that sense, the estimator increases its sensitivity to track changes in the underlying density as the size of the windows is decreased. On the other hand, for the estimator reported by Caudle and Wegman (2009), Kuiper's statistic shows that, for the three values of θ employed (which are the representative ones from Table 2), the fast adaptation

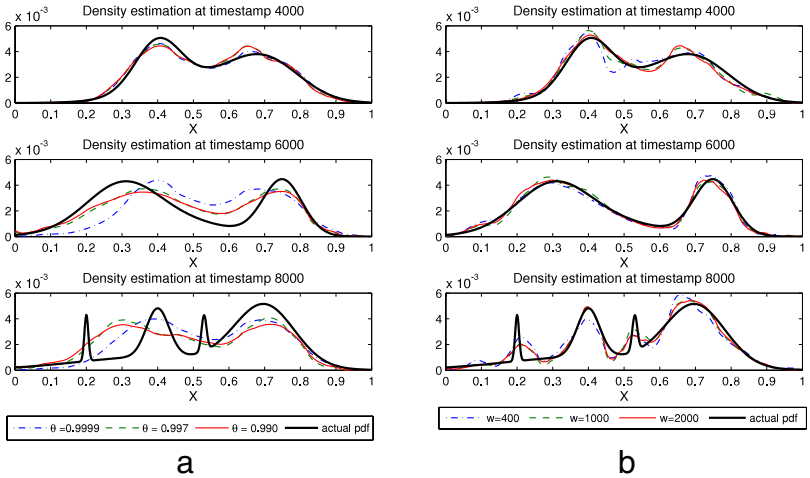


Fig. 4. Density estimation at timestamp 4000, 6000 and 8000 for (a) method reported by Caudle and Wegman (2009) and (b) for the proposed one.

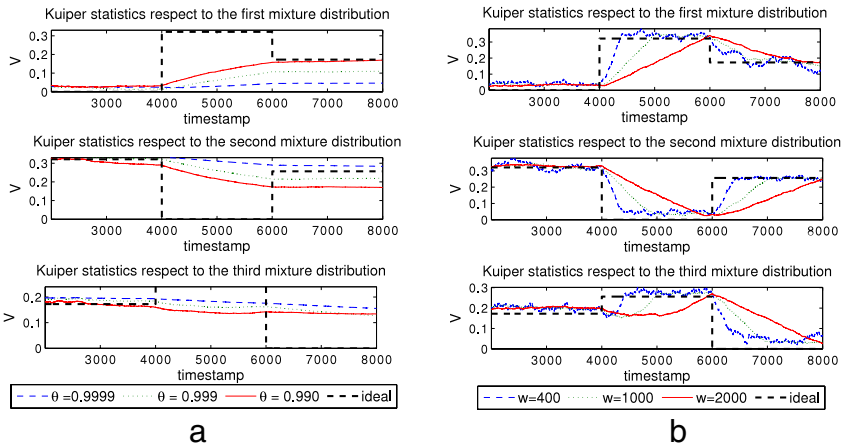


Fig. 5. Kuiper's statistics from timestamp 2000–8000 using different parameters. (a) Caudle and Wegman (2009) method (b) proposed method.

to changes in the underlying density is not possible. An important consequence of this fact is that, such an estimator is not suitable for moderately non-stationary scenarios. Note that, even when adjusting the value of the parameter θ, it is difficult to find an optimal balance between the adaptation and accuracy capabilities of the estimator. In contrast, according to results of Table 3 and Fig. 5, this problem is overcome in the proposed approach since the trade-off between the ability to track density changes and the precision of the estimation is controlled by the window size. The criterion for the selection of the window size is application specific, and in that sense, it can be selected according to which capability is more important in the application: (a) the sensitivity to adapt to new data or (b) the precision in the estimation. A short window is able to capture fast changes in the data, while a large window is suitable for a more precise estimation of the underlying density.

A relevant consideration for the density estimator presented in this work is that even though all data items covered by the sliding window are not processed, every time a new data item arrives to recompute the density estimate, they are still needed to be stored, and in that sense, as the size of the window increases, the storage required also increases. Additionally, regarding the convergence and precision of the procedure, it follows the same concepts discussed for its counterpart, the batch-based wavelet estimator, but it is important to highlight the fact that in the proposed case, the number of data items covered by the sliding window is the fundamental aspect that predefines the precision of the estimates.

Details concerning the interpretation of the relative values for Kuiper's statistics V of Fig. 5 are shown in Tables 4 and 5, where the averaged value for V for density estimates at timestamps 1–4000, 4001–6000 and 6001–8000 are presented. Note that, ideal values for V are shown within square brackets while the differences in percentage between V for the alternative and null hypothesis divided by the ideal value of V for the alternative hypothesis use round brackets. In our case, the null hypothesis corresponds to the situation in which $\hat{F}(x)$ is drawn from the underlying cumulative distribution $F(x)$ from Eq. (20). For example, ideally V with the underlying density $F(x)$ of Mixture 1 should be zero for the null hypothesis, that is, for timestamps 1–4000. On the contrary, V with $F(x)$ of Mixture 1 should be substantially greater than zero for the alternative hypothesis, ideally $V = 0.3212$ for timestamps 4001–6000 and $V = 0.1726$ for timestamps 6001–8000,

Table 4
Averaged Kuiper's statistic V for Caudle and Wegman (2009).

θ	0.9999			0.9990			0.9900		
Timestamps	1–4000	4001–6000	6001–8000	1–4000	4001–6000	6001–8000	1–4000	4001–6000	6001–8000
$F(x)$ of mixture 1	0.0329 [0] –	0.0471 [0.3212] (4.43%)	0.0610 [0.1726] (16.29%)	0.0179 [0] –	0.0739 [0.3212] (17.44%)	0.1187 [0.1726] (58.41%)	0.0266 [0] –	0.1138 [0.3212] (27.17%)	0.1684 [0.1726] (82.18%)
$F(x)$ of mixture 2	0.2817 [0.3212] (2.10%)	0.2885 [0] –	0.2646 [0.2565] (9.31%)	0.2322 [0.3212] (5.39%)	0.2496 [0] –	0.2044 [0.2565] (17.60%)	0.2233 [0.3212] (2.76%)	0.2145 [0] –	0.1678 [0.2565] (18.20%)
$F(x)$ of mixture 3	0.1704 [0.1726] (0.13%)	0.1859 [0.2565] (6.10%)	0.1702 [0] –	0.1379 [0.1726] (8.02%)	0.1727 [0.2565] (8.18%)	0.1517 [0] –	0.1265 [0.1726] (7.67%)	0.1491 [0.2565] (3.63%)	0.1398 [0] –

Table 5
Averaged Kuiper's statistic V for the proposed estimator.

w	200			400			800		
Timestamps	1–4000	4001–6000	6001–8000	1–4000	4001–6000	6001–8000	1–4000	4001–6000	6001–8000
$F(x)$ of mixture 1	0.0413 [0] –	0.2869 [0.3212] (76.48%)	0.2096 [0.1726] (97.54%)	0.0234 [0] –	0.2364 [0.3212] (66.30%)	0.2178 [0.1726] (112.61%)	0.0095 [0] –	0.1628 [0.3212] (47.75%)	0.2357 [0.1726] (131.08%)
$F(x)$ of mixture 2	0.2855 [0.3212] (63.74%)	0.0808 [0] –	0.2456 [0.2565] (64.28%)	0.2363 [0.3212] (40.77%)	0.1054 [0] –	0.2045 [0.2565] (38.65%)	0.1568 [0.3212] (2.32%)	0.1642 [0] –	0.1455 [0.2565] (7.28%)
$F(x)$ of mixture 3	0.1619 [0.1726] (51.53%)	0.2339 [0.2565] (62.76%)	0.0730 [0] –	0.1342 [0.1726] (26.54%)	0.2063 [0.2565] (45.97%)	0.0884 [0] –	0.0892 [0.1726] (25.43%)	0.1769 [0.2565] (17.09%)	0.1331 [0] –

to signal the difference between the empirical cumulative distribution of the data points and the underlying cumulative distribution $F(x)$ of Eq. (20). In order to show how the percentage of discrepancy is calculated, consider for instance that $F(x)$ is related to Mixture 1 and the proposed estimator is working with a window size of 200, then the percentage of discrepancy for timestamps 4001–6000 is $(0.2869 - 0.0413)/0.3212 * 100 = 76.48\%$ while for timestamps 6001–8000 is $(0.2096 - 0.0413)/0.1726 * 100 = 97.54\%$.

Regarding the percentage of discrepancy between alternative hypothesis with respect to the null hypothesis, note that, in general, their values (numbers within round brackets) are larger for the proposed estimator than for the approach reported by Caudle and Wegman (2009), except for the value obtained for timestamps 1–4000 with a window size of 800. Also note that, as the window size increases, the discrepancy between an alternative hypothesis with respect to the null hypothesis decreases. According to results reported in Tables 4 and 5, it is clear that there is a direct relationship between the window size of the estimator and the confidence to accept or reject a specific hypothesis. For this reason, in this paper, we are suggesting the use of Kuiper's statistic as a criterion for the selection of the window size parameter. More insights about how to solve this issue can be considered by analysing Fig. 6 where the values for Kuiper's statistic and for the MISE criterion evaluated over 100 trials are depicted.

The first observation about Fig. 6 is that both criteria, Kuiper's statistic and MISE, decrease asymptotically as the window size increases. Specifically, note that in Fig. 6(a) a window size of 500 (associated to point 1 in Fig. 6(a) would be suitable for Gaussian mixtures 1 and 2 if Kuiper's statistic V is expected to be 0.04 (see point 1 in Fig. 6(a)). On the other hand, a window size of 700 would fit for more complex mixtures like the Gaussian mixture 3 and the same expected V value (see point 2 in Fig. 6(a)). Regarding Fig. 6(b), if a MISE of 0.4×10^{-4} is specified, then a window size equal or larger than 300 data items would be appropriate for problems involving underlying densities such as the Gaussian mixture 1 or Gaussian mixture 2 (see point 1 in Fig. 6(b)). On the contrary, for cases in which the underlying density is similar to the more complex Gaussian mixture 3, then a proper window size may be 1000 data items (see point 1 in Fig. 6(b)).

4.2. Real-world data experiment: air pollution monitoring

In this subsection, the proposed estimator is tested using real-world data in the context of air pollution monitoring in the city of London, United Kingdom. The data used for the following experiment is publicly available and can be obtained from the London Air Quality Monitoring Network, LAQN (2010). The air pollution assessment is particularly important in urban environments, in which the air quality problem has become a severe and urgent issue that have a substantial impact on urban liveability and economic productivity. In that sense, the UK Government has established "Air Quality Standards" and "Air Quality Bands" for each of the major air pollutants (Sulphur Dioxide, Ozone, Carbon Monoxide, Nitrogen Dioxide and PM10 Particles) in order to have some benchmarks against which air pollution levels can be compared. As they are defined by LAQN (2010), air quality standards are set at a concentration, measured over a given time period, below which the effects

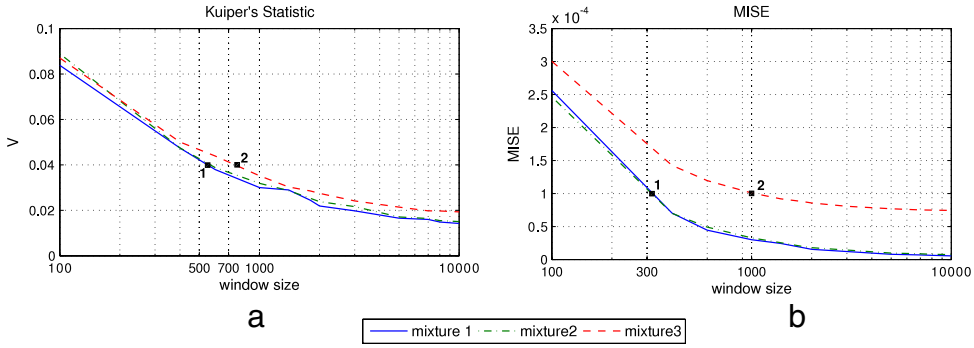


Fig. 6. Averaged Kuiper's statistic and averaged MISE over 100 trials.

Table 6
Air pollution bands and their corresponding thresholds, LAQN (2010).

Description	Low		Moderate		High		Very high
Ozone (ppb, hourly or 8 h running average)	<50	S	50–89	I	90–179	A	⩾ 180
PM10 Particles (μg m^{-3}, 24 h running average)	<63	S	63–94	I	95–127	A	⩾ 128

of pollution levels are considered acceptable for human health and for the environment. With respect to air pollution bands, pollution levels are classified into bands with the purpose of serving as a tool to help the public assess the possible health impacts of pollution above certain thresholds; see LAQN (2010) for details.

In this work, the attention is specifically focused on Ozone and PM10 particles. The reason for this choice is based on, first, the corresponding air quality standard and, second, on the availability of the data. Table 6 shows the relationship between the Air Pollution Bands and their corresponding thresholds for the two pollutants evaluated in the experiment. The first of these thresholds, is the "Standard Threshold" (S), which is based on the air quality standard for each pollutant. Further thresholds are the "Information" (I) and "Alert" (A) levels that are in line with EC directives on air quality. According to LAQN (2010), any concentration below the Standard threshold is described as a "low air pollution scenario" while a level between the Standard and Information thresholds is described as "moderate", in the same way, concentrations between the Information and Alert thresholds, and above the Alert threshold are denoted as "high" and "very high" air pollution scenarios, respectively. It is important to notice that, the time taken for exposure to a pollutant to cause adverse health effects varies from pollutant to pollutant, and in that sense, the times over which concentrations are evaluated are different. For Ozone, the maximum of the 8 h running and hourly mean is used to calculate the index value (Note that for the experiment, just the 8 h running average is used). In contrast, for PM10 particles the mean of 24 h running is considered. Additionally, note that the units for these two pollutants are different, Ozone is measured in points per billion (ppb) while PM10 particles in μg m^{-3}.

The experiment involved testing the discriminative capability of the estimator under different air quality bands over one year of data for both PM10 particles and Ozone. Specifically, the selected sites were North Kensington (for Ozone) and Cromwell Road (for PM10) that are two central locations in the city both inside the Royal Borough of Kensington and Chelsea. Additionally, the year 2006 is considered for Ozone while the year 2008 for PM10 particles both based on the fact that data related to those years are fully ratified and, in those years at least three of the air pollution scenarios previously described (i.e. low, moderate and high) were present. Fig. 7(a) and (b) show the data used for the experiment as well as their corresponding air quality standard and thresholds (dotted lines). It is important to mention that the sampling period for the data is 15 min and there are some missing information for the case of Ozone as it can be appreciated in Fig. 7(b).

The issue of computing density estimates with missing data items is directly related to the problem of density estimation with decreasing window size which was shown in Fig. 6(a). Depending on the complexity of the underlying density, reducing the number of data items available for the density estimation procedure impacts the precision of the estimate. The estimator design should take into account a certain margin of error in the estimate, to make it able to deal with a predefined amount of missing data. For our particular experiment, regions with missing data items covering more than the 5% of the window size are not considered for computation of the estimate (e.g. most of the month April 2008 for the case of Ozone) while regions with less than that percentage of missing data are computed in a normal way. Note that, for the case of the proposed estimator, when a new arriving data item is missing, the estimator coefficients $\hat{b}_{j,k,n+1}$ of Eq. (17) at timestamp $n+1$ is not modified and then is equal to $\hat{b}_{j,k,n}$. The same strategy is followed for discounting the missing data.

For the experiment, Kuiper's statistic defined in Eq. (20) and previously used for simulated data is also employed. Specifically, first, the underlying probability density function of the pollutants for the years described are obtained in an online and recursive fashion. Then, for each pollutant, densities corresponding to the same air quality bands (i.e. low, moderate and high) are grouped and averaged. Finally, the corresponding CDF's are calculated and Kuiper's statistic is applied among them. For the proposed estimator, only one window size was evaluated for each pollutant based on the

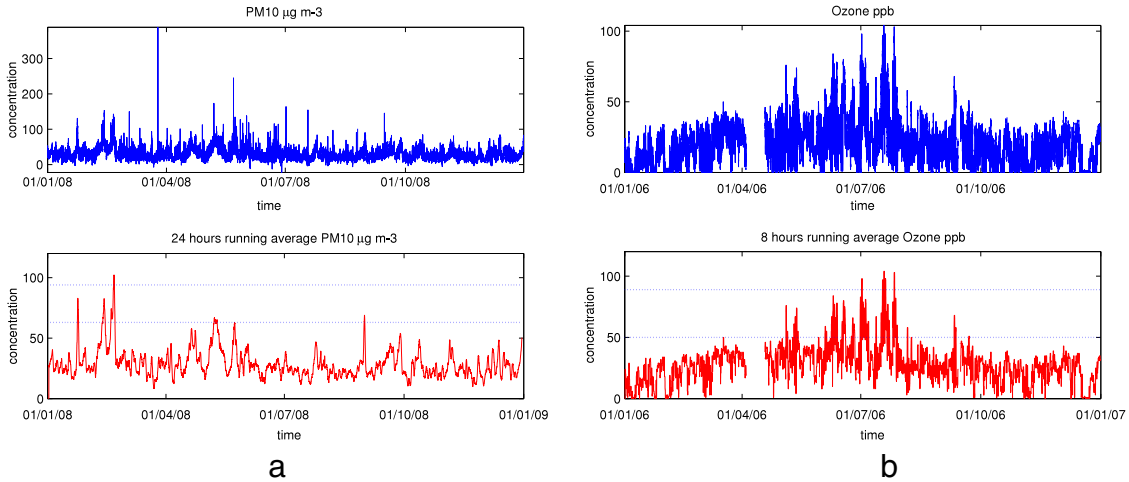
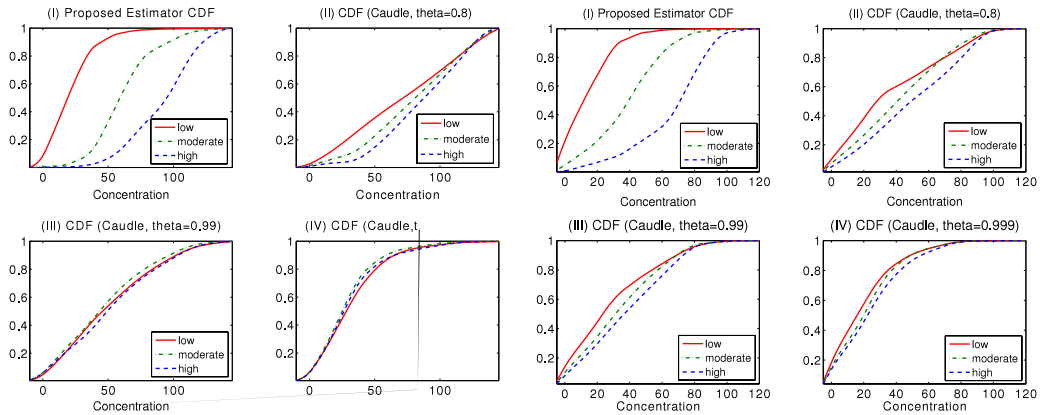


Fig. 7. (a) PM10 Particles concentrations for the whole year 2008 and their corresponding 24 h running averages at Cromwell Road's air pollution monitoring station; (b) Ozone concentrations for the whole year 2008 and their corresponding 8 h running averages at North Kensington's air pollution monitoring station, LAQN (2010).



(a) PM10 2008: (I) Proposed approach with a window size equal to 96 data items, (II)–(IV) Caudle and Wegman (2009) method using different θ values.

(b) Ozone 2006: (I) Proposed approach with a window size equal to 326 data items, (II)–(IV) Caudle and Wegman (2009) method using different θ values.

Fig. 8. Averaged cumulative density functions for low, moderate and high pollution scenarios.

air quality standard i.e. 32 data items for Ozone and 96 items for PM10, corresponding to 8 h and 24 h, respectively. In contrast, for the benchmark estimator reported by Caudle and Wegman (2009), three different values of theta are evaluated, i.e. 0.8, 0.99 and 0.999, considering the previous evaluation for simulated data. The experiment results are shown in Fig. 8(a) and (b) and presented in Table 7, where the label "low–moderate" is related to Kuiper's statistic between the low air pollution scenarios CDF and the moderate one. In that sense, labels "moderate–high" and "low–high" indicate Kuiper's statistic between moderate and high, and low and high air pollution scenarios, respectively. It is important to point out that the higher the value of Kuiper's statistic,the higher the difference and discrimination between densities from different air quality scenarios i.e. low, moderate and high. It can be noticed form Fig. 8(a) and (b) and Table 7 that the density estimation obtained with the proposed recursive approach have higher discrimination capabilities in terms of Kuiper's statistic and in that sense the estimator is able to clearly distinguish between different air pollution scenarios. Furthermore, the proposed estimator opens the possibility to introduce more robust air quality standards or strategies due to the fact that a probability density function offers more insights about the process or system under analysis than a single mean value.

One of the advantages of the proposed estimation technique is the fact that, in the particular context of air quality assessment the time frame is well defined. Remember that "air quality standards" consider not only concentration values, but also the persistence of pollution levels over a fixed and well defined period of time, e.g. 8 h for the case of Ozone and 24 h for PM10 particles. In contrast to the estimator introduced by Caudle and Wegman (2009) in which theta values adjust the relevance balance between old and new data, the proposed approach can estimate densities for a fixed period of time in which data within a period of time require the same level of importance. Furthermore, the proposed approach has shown to be particularly suitable for applications involving running averages, like the air pollution context.

Table 7
Kuiper's statistic among different air pollution scenarios.

Evaluation results		Kuiper's statistic		
Pollutant	Method	Low–Moderate	Moderate–High	Low–High
Ozone	Proposed (window = 32)	0.5223	0.5113	0.7886
	Caudle ($\theta = 0.8$)	0.1716	0.1209	0.2250
	Caudle ($\theta = 0.99$)	0.1222	0.0769	0.1771
	Caudle ($\theta = 0.999$)	0.0780	0.0607	0.1201
PM10	Proposed (window = 96)	0.7296	0.4804	0.8658
	Caudle ($\theta = 0.8$)	0.1477	0.0944	0.2332
	Caudle ($\theta = 0.99$)	0.0422	0.0653	0.0632
	Caudle ($\theta = 0.999$)	0.0766	0.0301	0.0577

5. Final remarks

In recent years, there has been an important emergence of applications involving streaming data. For these applications, an important issue to be addressed is the estimation of the underlying probability density of the data. In this paper, the problem of density estimation in the context of data streams is investigated following the concept of sliding windows and wavelet-based orthogonal series estimators. In that sense, a novel online algorithm for density estimation of data streams is proposed which incorporates a selective reevaluation procedure for the updating of estimator coefficients. The experimental results show that the method proposed is suitable for non-stationary applications where the corresponding probability density function is changing over time as well as for cases involving running average-based metrics. In such contexts, it outperforms recent solutions for density estimation for streaming data. Furthermore, the method proposed has good adaptation capabilities and it requires a fixed amount memory.

Future work will be focused on testing the algorithm in other real world data streams applications. Additionally, since the proposed method is able to track the evolution of a given density, future work will also investigate the suitability of the framework in online anomaly detection applications.

Acknowledgement

The first author gratefully acknowledges the financial support from the National Council on Science and Technology (CONACYT) Mexico.

Appendix. Extension to higher dimensions

In this section, the proposed recursive wavelet density estimator is extended to higher dimensions. For that purpose, both the construction of multidimensional multiresolution analysis and their corresponding multidimensional wavelets is briefly presented following the concept of *separable multiresolution approximations*. Details can be found in Mallat (1989), Ogden (1997) and Safavi et al. (2004). In this approach, the one-dimensional multiresolution analysis is extended to an m-dimensional multiresolution analysis by defining the m-dimensional vector space $\mathbf{V}_j^m$ as the tensor power of its one-dimensional counterpart V_j:

$$\mathbf{V}_j^m = V_j^{\otimes m} = \underbrace{V_j \otimes \cdots \otimes V_j}_{m}, \quad j, m \in \mathbb{Z}, \tag{A.1}$$

where $\otimes$ denotes tensor product and the subindex j refers to the resolution. In addition, $\mathbf{V}_j^m$ satisfies a m-dimensional multiresolution ladder in $L^2(\mathbb{R}^m)$,

$$\ldots \subset \mathbf{V}_{-2}^m \subset \mathbf{V}_{-1}^m \subset \mathbf{V}_0^m \subset \mathbf{V}_1^m \subset \mathbf{V}_2^m \ldots \tag{A.2}$$

with

$$\cap_{j\in\mathbb{Z}} \mathbf{V}_j^m = \{0\}; \qquad \overline{\cup_{j\in\mathbb{Z}} \mathbf{V}_j^m} = L^2(\mathbb{R}^m). \tag{A.3}$$

Eqs. (A.2) and (A.3) imply that the sequence of vector spaces $\mathbf{V}_j^m$ forms a multiresolution approximation of the space $L^2(\mathbb{R}^m)$ if and only if V_j is a multiresolution approximation of $L^2(\mathbb{R})$. In that sense, the basis functions for the vector space $\mathbf{V}_j^m$ are defined by:

$$\begin{aligned}\Phi_{j,k_1,k_2,\ldots,k_m}^m(x_1, x_2, \ldots, x_m) &= \phi_{j,k_1}(x_1)\phi_{j,k_2}(x_2)\ldots\phi_{j,k_m}(x_m)\\ &= 2^{mj/2}\Phi(2^j x_1 - k_1, 2^j x_2 - k_2, \ldots, 2^j x_m - k_m),\end{aligned} \tag{A.4}$$

where $\phi_{j,k_i}(x_i)$ is the one-dimensional scaling function associated to the vector space V_j, and orthonormal basis for $\mathbf{V}_j^m$ is given by $\{\Phi_{j,k_1,k_2,\ldots,k_m}^m(x_1, x_2, \ldots, x_m), j, k_1, k_2, \ldots, k_m \in \mathbb{Z}\}$. Additionally, recalling that, in the one-dimensional case, the complementary vector space W_j represents the detail signal between successive approximations $V_{j+1} = V_j \oplus W_j$, then, the

m-dimensional vector space $\mathbf{V}_{j+1}^m$ can be expressed as:

$$\mathbf{V}_{j+1}^m = V_{j+1}^{\otimes m} = \underbrace{V_{j+1} \otimes \cdots \otimes V_{j+1}}_{m} = \underbrace{(V_j \oplus W_j) \otimes \cdots \otimes (V_j \oplus W_j)}_{m} = \mathbf{V}_j^m \oplus \mathbf{W}_j^m, \quad j, m \in \mathbb{Z}, \tag{A.5}$$

where $\mathbf{W}_j^m$ is the orthogonal complement of $\mathbf{V}_j^m$ in $\mathbf{V}_{j+1}^m$. The detail space $\mathbf{W}_j^m$ consists of $2m-1$ orthogonal subspaces whose m-dimensional basis functions are given by combinations of tensor products between the one-dimensional approximation and detail spaces, V_j and W_j, respectively. The corresponding basis functions are then:

$$\begin{aligned}
&\Psi_{j,k_1,k_2,\ldots,k_m}^{m,1}(x_1, x_2, \ldots, x_m) = \phi_{j,k_1}(x_1)\psi_{j,k_2}(x_2)\ldots\psi_{j,k_m}(x_m),\\
&\Psi_{j,k_1,k_2,\ldots,k_m}^{m,2}(x_1, x_2, \ldots, x_m) = \psi_{j,k_1}(x_1)\phi_{j,k_2}(x_2)\ldots\psi_{j,k_m}(x_m),\\
&\vdots\\
&\Psi_{j,k_1,k_2,\ldots,k_m}^{m,2m-1}(x_1, x_2, \ldots, x_m) = \psi_{j,k_1}(x_1)\psi_{j,k_2}(x_2)\psi_{j,k_m}(x_m),
\end{aligned} \tag{A.6}$$

by considering $p = \{1, 2, \ldots, 2m-1\}$, Eq. (A.6) can be shortly expressed as:

$$\Psi_{j,k_1,k_2,\ldots,k_m}^{m,p}(x_1, x_2, \ldots, x_m) = 2^{mj/2}\Psi^p(2^j x_1 - k_1, 2^j x_2 - k_2, \ldots, 2^j x_m - k_m,), \tag{A.7}$$

where the set $\{\Psi_{j,k_1,k_2,\ldots,k_m}^{m,p}(x_1, x_2, \ldots, x_m), j, k_1, k_2, \ldots, k_m \in \mathbb{Z}\}$ constitutes an orthonormal basis for $\mathbf{W}_j^m$ in $L^2(\mathbb{R}^m)$. Based on the above m-dimensional multiresolution analysis, and defining m-dimensional random variable $X = [X_1, X_2, \ldots, X_m]$, then the m-dimensional version of the batch-processing based wavelet density estimator of Eq. (12) is then:

$$\hat{f}(x_1, x_2, \ldots, x_m) = \sum_{\mathbf{k}_h} \hat{c}_{j_0,\mathbf{k}_h} \Phi_{j_0,\mathbf{k}_h}^m(x_1, x_2, \ldots, x_m) + \sum_p \sum_{j=j_0}^{j=j_0+J} \leq j \sum_{\mathbf{k}_h} \hat{d}_{j,\mathbf{k}_h}^p \Psi_{j,\mathbf{k}_h}^{m,p}(x_1, x_2, \ldots, x_m), \tag{A.8}$$

where, for simplification purposes, $h = \{1, 2, \ldots, m\}$ and then $\mathbf{k}_h = \{k_1, k_2, \ldots, k_m\}$. Following Eqs. (10) and (11), the m-dimensional extension for the approximation of the scaling and wavelet coefficients is given by:

$$\hat{c}_{j_0,\mathbf{k}_h} = \frac{1}{n}\sum_{i=0}^{n} \Phi_{j_0,\mathbf{k}_h}^m(X_1^i, X_2^i, \ldots, X_m^i), \quad j_0 \in \mathbb{Z};\ \mathbf{k}_h \in \mathbb{Z}^m, \tag{A.9}$$

and

$$\begin{aligned}
\hat{d}_{j,\mathbf{k}_h}^1 &= \frac{1}{n}\sum_{i=0}^{n} \Psi_{j,\mathbf{k}_h}^{m,1}(X_1^i, X_2^i, \ldots, X_m^i),\\
\hat{d}_{j,\mathbf{k}_h}^2 &= \frac{1}{n}\sum_{i=0}^{n} \Psi_{j,\mathbf{k}_h}^{m,2}(X_1^i, X_2^i, \ldots, X_m^i),\\
&\vdots\\
\hat{d}_{j,\mathbf{k}_h}^{2m-1} &= \frac{1}{n}\sum_{i=0}^{n} \Psi_{j,\mathbf{k}_h}^{m,2m-1}(X_1^i, X_2^i, \ldots, X_m^i),
\end{aligned} \tag{A.10}$$

where $\hat{c}_{j_0,\mathbf{k}_h}$ and $\hat{d}_{j,\mathbf{k}_h}^1, \hat{d}_{j,\mathbf{k}_h}^2, \ldots, \hat{d}_{j,\mathbf{k}_h}^{2m-1}$ are the approximated coefficients for the scaling and wavelet functions, respectively. Finally, the recursive updating of the estimator coefficients of Eq. (17), can be extended for m-dimension according to the following equation:

$$\hat{b}_{j,\mathbf{k}_h,n+1} = \hat{b}_{j,\mathbf{k}_h,n} + \hat{b}_{j,\mathbf{k}_h,n+1}^{\text{addition}} - \hat{b}_{j,\mathbf{k}_h,n+1}^{\text{discounting}}, \tag{A.11}$$

with

$$\hat{b}_{j,\mathbf{k}_h,n+1}^{\text{addition}} = \begin{cases} \dfrac{\varphi_{j,\mathbf{k}_h}(X_1^{n+1}, X_2^{n+1}, \ldots, X_m^{n+1})}{\mathbf{w}}; & \text{if } 2^j k_1 \leq X_1^{n+1} \leq 2^j(k_1 + 2N - 1),\\ & \text{and } 2^j k_2 \leq X_2^{n+1} \leq 2^j(k_2 + 2N - 1),\\ & \vdots\\ & \text{and } 2^j k_m \leq X_m^{n+1} \leq 2^j(k_m + 2N - 1),\\ 0; & \text{otherwise} \end{cases} \tag{A.12}$$

and

$$\hat{b}^{\text{discounting}}_{j,\mathbf{k}_h,n+1} = \begin{cases} \dfrac{\varphi_{j,\mathbf{k}_h}(X_1^{n-w+1}, X_2^{n-w+1}, \ldots, X_m^{n-w+1})}{\mathbf{w}}; & \text{if } 2^j k_1 \le X_1^{n-w+1} \le 2^j(k_1 + 2N - 1), \\ & \text{and } 2^j k_2 \le X_2^{n-w+1} \le 2^j(k_2 + 2N - 1), \\ & \vdots \\ & \text{and } 2^j k_m \le X_m^{n-w+1} \le 2^j(k_m + 2N - 1), \\ 0; & \text{otherwise} \end{cases} \tag{A.13}$$

where $\varphi_{j,\mathbf{k}_h}$ is used to refer to either the m-dimensional wavelet $\Psi^{m,p}_{j,\mathbf{k}_h}(x)$ or its corresponding m-dimensional scaling function $\Phi^m_{j,\mathbf{k}_h}(x)$ and $\hat{b}_{j,\mathbf{k}_h,n}$ refer to their corresponding coefficient for the timestamp n. Additionally, $\mathbf{w}$ is the size of a m-dimensional sliding window, where $X_1^{n+1}, X_2^{n+1}, \ldots, X_m^{n+1}$ and $X_1^{n-w+1}, X_2^{n-w+1}, \ldots, X_m^{n-w+1}$ are the most recent and oldest m-dimensional data items covered by the sliding window.

References

Babcock, B., Babu, S., Datar, M., Motwani, R., Widom, J., 2002. Models and issues in data stream systems. In: Proceedings of the Twenty-First ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems. ACM, Madison, Wisconsin, pp. 1–16.

Blohsfeld, B., Heinz, C., Seeger, B., 2005. Maintaining nonparametric estimators over data streams. In: The German Database Conference "Datenbanksysteme in Büro. Technik und Wissenschaft" BTW, pp. 385–404.

Boedihardjo, A.P., Lu, C.-T., Chen, F., 2008. A framework for estimating complex probability density structures in data streams. In: Proceeding of the 17th ACM Conference on Information and Knowledge Management. ACM, Napa Valley, California, USA, pp. 619–628.

Boggess, A., Narcowich, F., 2009. A First Course in Wavelets with Fourier Analysis. Wiley.

Bruce, L., Koger, C., Li, J., 2002. Dimensionality reduction of hyperspectral data using discrete wavelet transform feature extraction. IEEE Transactions on Geoscience and Remote Sensing 40, 2331–2338.

Burrus, C., Gopinath, R., Guo, H., Odegard, J., Selesnick, I., 1997. Introduction to Wavelets and Wavelet Transforms: A Primer. Prentice Hall, Upper Saddle River, NJ.

Caudle, K.A., Fowler, G.O., Jager, L.R., Ruth, D.M., 2011. Discounting older data. Wiley Interdisciplinary Reviews: Computational Statistics 3, 30–33.

Caudle, K.A., Wegman, E., 2009. Nonparametric density estimation of streaming data using orthogonal series. Computational Statistics & Data Analysis 53, 3980–3986.

Céncov, N., 1962. Evaluation of an unknown distribution density from observations. Soviet Mathematics Doklady 3, 1559–1562.

Cohen, A., Daubechies, I., Vial, P., 1993. Wavelets on the interval and fast wavelet transforms. Applied and Computational Harmonic Analysis 1 (28), 54–81.

Daubechies, I., Lagarias, J., 1992. Two-scale difference equations II. Local regularity, infinite products of matrices and fractals. SIAM Journal on Mathematical Analysis 23, 1031–1079.

Domingos, P., Hulten, G., 2003. A general framework for mining massive data streams. Journal of Computational and Graphical Statistics 12, 945–949.

Donoho, D., Johnstone, I., 1998. Minimax estimation via wavelet shrinkage. The Annals of Statistics 26, 879–921.

Donoho, D., Johnstone, I., Kerkyacharian, G., Picard, D., 1996. Density estimation by wavelet thresholding. The Annals of Statistics 24, 508–539.

Golab, L., Ozsu, M.T., 2003. Issues in data stream management. In: ACM Special Interest Group on Managment of Data SIGMOD Rec., vol. 32, pp. 5–14.

Hall, P., 1986. On the rate of convergence of orthogonal series density estimators. Journal of the Royal Statistical Society. Series B (Methodological) 48, 115–122.

Hall, P., Patil, P., 1995. Formulae for mean integrated squared error of nonlinear wavelet-based density estimators. The Annals of Statistics 23, 905–928.

Heinz, C., 2007. Density estimation over data streams. Ph.D. Thesis. Philipps University Marburg.

Heinz, C., Seeger, B., 2005. Wavelet density estimators over data streams. In: SAC'05: Proceedings of the 2005 ACM Symposium on Applied Computing. ACM, New York, NY, USA, pp. 578–579.

Herrick, D., Nason, G., Silverman, B., 2001. Some new methods for wavelet density estimation. Sankhya the Indian Journal of Statistics, Series A 63, 394–411.

LAQN, 2010. The London air quality network. http://www.londonair.org.uk [accessed on June-2010].

Mallat, S., 1989. A theory for multiresolution signal decomposition: the wavelet representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 11, 674–693.

Marron, J.S., Wand, M.P., 1992. Exact mean integrated squared error. The Annals of Statistics 20, 712–736.

Masry, E., 1994. Probability density estimation from dependent observations using wavelets orthonormal bases. Statistics and Probability Letters 21, 181–194.

Ogden, R., 1997. Essential Wavelets for Statistical Applications and Data Analysis. Birkhauser.

Percival, D.B., Walden, A.T., 2000. Wavelets Methods of Time Series Analysis. Cambridge University Press.

Pinheiro, A., Vidakovic, B., 1997. Estimating the square root of a density via compactly supported wavelets. Computational Statistics & Data Analysis 25, 399–415.

Procopiuc, C., Procopiuc, O., 2005. Density estimation for spatial data streams. In: Advances in Spatial and Temporal Databases. pp. 109–126.

Rohatgi, V., Saleh, A., 1976. An Introduction to Probability and Statistics. John Wiley and Sons Inc., New York

Safavi, A., Chen, J., Romagnoli, J., 2004. Wavelet-based density estimation and application to process monitoring. AIChE Journal 43, 1227–1241.

Scott, D., 1992. Multivariate Density Estimation: Theory, Practice, and Visualization. Wiley-Interscience.

Silverman, B., 1998. Density Estimation for Statistics and Data Analysis. Chapman and Hall, CRC.

Stephens, M.A., 1965. The goodness-of-fit statistic v_n: distribution and significance points. Biometrika 52, 309–321.

Stephens, M.A., 1970. Use of the Kolmogorov–Smirnov, cramer-von mises and related statistics without extensive tables. Journal of the Royal Statistical Society. Series B (Methodological) 32, 115–122.

Tygert, M., 2010. Statistical tests for whether a given set of independent, identically distributed draws comes from a specified probability density. Proceedings of the National Academy of Sciences 107, 16471–16476. doi:10.1073/pnas.1008446107.

Vannucci, M., 1995. Nonparametric density estimation using wavelets. Technical Report 95-26. Institute of Statistics and Decision Sciences, Duke University. Available: http://www.isds.duke.edu.

Vidakovic, B., 1999. Statistical Modeling by Wavelets. Wiley, New York.

Wegman, E., Caudle, K., 2006. Density estimation from streaming data using wavelets. In: Compstat 2006-Proceedings in Computational Statistics, pp. 231–242.

Wegman, E., Marchette, D., 2003. On some techniques for streaming data: a case study of internet packet headers. Journal of Computational and Graphical Statistics 12, 893–914.