Self-tuning density estimation based on Bayesian averaging of adaptive kernel density estimations yields state-of-the-art performance

Christofer L. Bäcklin, Claes Andersson, Mats G. Gustafsson

PII: S0031-3203(18)30006-2
DOI: 10.1016/j.patcog.2018.01.008
Reference: PR 6417

To appear in: *Pattern Recognition*

Received date: 1 May 2017
Revised date: 30 December 2017
Accepted date: 7 January 2018

Highlights

- A new method called ADEBA for multivariate adaptive density estimation is presented.
- A simulation study shows that ADEAB is competitve to currently dominating methods.
- ADEBA is simple and computes much faster than Gaussian mixture modeling.
- Further improvements can be made by incorporating application-specific prior knowledge into ADEBA.
- Implementations of ADEBA are publicly available for R.

Self-tuning density estimation based on Bayesian averaging of adaptive kernel density estimations yields state-of-the-art performance

Christofer L. Bäcklin, Claes Andersson, Mats G. Gustafsson

Department of Medical Sciences, Cancer Pharmacology and Computational Medicine, Uppsala University, Uppsala 751 85, Sweden

Abstract

Non-parametric probability density function (pdf) estimation is a general problem encountered in many fields. A promising alternative to the dominating solutions, kernel density estimation (KDE) and Gaussian mixture modeling, is adaptive KDE where kernels are given individual bandwidths adjusted to the local data density. Traditionally the bandwidths are selected by a non-linear transformation of a pilot pdf estimate, containing parameters controlling the scaling, but identifying parameters values yielding competitive performance has turned out to be non-trivial.

We present a new self-tuning (parameter free) pdf estimation method called adaptive density estimation by Bayesian averaging (ADEBA) that approximates pdf estimates in the form of weighted model averages across all possible parameter values, weighted by their Bayesian posterior calculated from the data.

ADEBA is shown to be simple, robust, competitive in comparison to the current practice, and easily generalize to multivariate distributions. An implementation of the method for R is publicly available.

Keywords: adaptive density estimation, variable bandwidth, Bayesian model averaging, square root law, multivariate, univariate

2010 MSC: 62G07, 62F15

1. Introduction

The problem of estimating a probability density function (pdf) based on a sample of data has received much attention in a wide range of applications. Some of them are summarized in reviews by [40, 52], including one by Chacon [10] on the closely related problem of estimating the derivative of the pdf. KDE may be regarded as a family of methods that is commonly used to perform pdf estimation, with the perhaps most well known member being the Parzen window estimator [30]. The most popular KDE members today are based on selecting a single global bandwidth h as in the univariate method proposed by Sheather and Jones [41]. In the less well developed multivariate case there is no outstanding most popular method for global bandwidth (or bandwidth matrix) selection, but the most popular approach seems to be Gaussian mixture modeling using a fixed number of Gaussians.

Although the use of a global constant bandwidth, or bandwidth matrix in the multivariate case, is expected to result in difficulties for certain types of underlying pdf this approach has not yet been replaced by the more flexible alternative to use adaptive (also called variable) bandwidth kernel density estimation (AKDE) where each kernel is given an individual bandwidth. For the type of AKDE studied here, each individual bandwidth is determined by means of a rough pilot pdf estimate $\hat{p}_0(x)$, obtained using some robust estimation procedure, that is allowed to influence the local bandwidth via a non-linear transformation. The mathematical expression for the AKDE methods considered in this work is shown in (3), which illustrates how the fixed global bandwidth h used in standard KDE is replaced by a local data dependent value $h(x_i)$ which is a non-linear transformation (function) of data point x_i.

The particular non-linear transformation used in this work is $h(x_i) = \frac{\alpha}{(\hat{p}_0(x_i))^\beta}$ where the α parameter specifies global scaling and β specifies the influence of the pilot pdf [43]. So far this approach has not yielded competitive performance because it has turned out to be a non-trivial or perhaps impossible task to find an algorithm that always choose the values of these parameters sufficiently well to outperform the global alternative. One natural and attractive approach in this context is the use of Bayesian inference to obtain promising/optimal parameter values for α and β as suggested by Brewer [9] and Wu et al. [51]. One may also note that in recent years, several Bayesian approaches for the selection of a single global bandwidth have been proposed [54, 44, 22, 56, 57, 58].

Based on the assumption that in many practical situations there will be no single pair (α,β) of parameter values that will yield satisfying results in AKDE, we have studied to what extent Bayesian model averaging (BMA) [20] could be used to improve the situation. As demonstrated below with both theory and simulation results, the resulting new method denoted *adaptive density estimation by Bayesian averaging* (ADEBA) is a new type of pdf estimation method that does not require the user to specify any parameter values and that is simple, robust, and competitive in comparison with the state-of-the-art. Moreover, the method is not restricted to the univariate case but also enables competitive estimation in arbitrary many dimensions within reasonable numerical complexity. The study is novel and innovative as it explores an until known largely overlooked possibility to use Bayesian model averaging to make AKDE self-tuned. Previously Bayesian approaches have been used but only to find the most probably/promising hyperparameter values. Moreover, this possibility of Bayesian averaging is not trivial to explore because it has to employ a tailor made leave-one out procedure introduced here which is required to avoid overfitting and obtain an successful algorithm. The resulting algorithm is attractive as it is self-tuning (it has eliminated all user defined parameters).

1.1. Historical background and applications

KDE was originally introduced in a technical report from 1951 [14, reprint] to approximate optimal pattern classification based on likelihood ratios, thus minimizing the probability of misclassification (see also the valuable historical review and acknowledgment of this work by Silverman and Jones [42]). Another notable early report is the article by Sebestyen [38] where KDE is used to build Bayes optimal classifiers. This work was published the same year as the first article describing the already mentioned univariate Parzen-window estimator, and 3 years before the first article explicitly addressing multivariate KDE [27]. In recent years, the same basic idea has been used to create probabilistic classifiers [53, 32]; combine multiple binary classifiers into multi-class classifiers [37]; quantify the likelihood of relations between entities in structured knowledge bases [6]; and automate image annotation [23]. KDE has also been used in various ways for image segmentation and object detection [11, 12, 31, 50]. Extending into the temporal dimension, related on-line methods based on KDE has been used for background modeling, detection and tracking of moving object in video [55, 45, 34, 4].

Also worth mentioning is the possibility to use KDE for estimation of mutual information [15], which is useful for detecting non-linear associations in data sets and may be used for variable selection [49, 24].

2. Theoretical and algorithmic results

2.1. Background definitions and terminology

Let $D = \{x_1, ..., x_n\}$ be a sample of n independent, identically distributed univariate data points from an unknown pdf $p(x)$. The standard univariate KDE then has the form

$$\hat{p}(x|D,h) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{1}$$

where K is a kernel function satisfying $\int K(x)dx = 1$ and h is a constant smoothing parameter known as the bandwidth [35, 30]. It is well known that the asymptotic convergence of $\hat{p}(x|D,h)$ towards the underlying $p(x)$ for large data sets is greatly affected by the choice of h whereas the selection of K is far less important [40]. Therefore, much work has been spent on trying to find an optimal choice of h based on D that makes KDE practically successful. Since 1991, the bandwidth selector suggested by Sheather and Jones [41] (SJ) has remained state-of-the-art due to its solid theoretical foundation as well as its overall good performance. For a more recent attempt to improve the SJ selector in cases when the pdf deviates severely from normality, see the work by Liao et al. [26].

A natural extension of this standard KDE with potential of yielding better estimates is adaptive KDE (AKDE), also known as variable KDE [47, 36], which is obtained whenever h is no longer a global constant. Following Terrell and Scott [47], the AKDE methods can be divided into two main families. One is the family of *balloon estimators*, which can be written

$$\hat{p}(x|D,h) = \frac{1}{nh(x)} \sum_{i=1}^{n} K\left(\frac{x_i - x}{h(x)}\right). \tag{2}$$

Although this family is advantageous for asymptotic analyzes, its major drawback is that this pdf estimate typically does not integrate to one and thus is not a proper pdf. The second family of AKDEs is the *sample smoothing estimators* which can be written

$$\hat{p}(x|D,h) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{h(x_i)} K\left(\frac{x - x_i}{h(x_i)}\right) \tag{3}$$

and where there is one individual bandwidth $h(x_i)$ per data point. This kind of estimator was first proposed by Breiman et al. [8] who suggested choosing $h(x_i)$ proportional to the Euclidean distance $d_{i,k} = ||x_i - x_{i,k}||$ from x_i to its kth nearest other sample point.

$$h(x_i) = \alpha_k d_{i,k} \tag{4}$$

This was followed by the influential work by Abramson [1] suggesting to choose $h(x_i)$ as $h(x_i) = \frac{1}{\hat{p}_0(x_i)^{1/2}}$ where $\hat{p}_0(x_i)$ is a first rough estimate of the desired pdf value $p(x_i)$ called the pilot estimate. This suggestion is known as the *the square-root law* and resulted in the perhaps the most well known member of this family of AKDEs.

The sample smoothing estimates always integrate to one but suffers from the major problem of *non-locality*, which means that the density estimate at one point x might be heavily influenced by data points x_i that are far away and thus not in the neighborhood of x. Therefore, a refinement of this approach is to introduce one parameter α that controls the global scaling of the kernels and another parameter β that controls how much influence the pilot $\hat{p}_0(x)$ has on the final estimate. Following Silverman [43, Section 5.3.1], this can be formalized by expressing the bandwidth at x_i as

$$h_i(\alpha, \beta) = h(x_i, \alpha, \beta) = \left[\frac{\alpha}{\hat{p}_0(x_i)}\right]^{\beta} . \tag{5}$$

The recommendation of Silverman is to choose α as the geometric mean of the pilot evaluated at the data points available,

$$\alpha = \left[\prod_{i=j}^{n} \hat{p}_0(x_j)\right]^{1/n} , \tag{6}$$

and to select $\beta = 0.5$ as in the Abramson estimator. For ADBEA, we choose a slightly different parametrization (7) compared to (5) to avoid the dependency between the parameters.

$$h(x_i, \alpha, \beta) = \frac{\alpha}{(\hat{p}_0(x_i))^{\beta}} \tag{7}$$

2.2. Adaptive density estimation by Bayesian averaging (ADEBA)

In the work presented here, we have investigated the potential gains of going beyond this kind of AKDEs by means of BMA. We are assuming the two-parameter model 95

$$\begin{aligned} \hat{p}(x|D, \alpha, \beta) &= \hat{p}(x|D, \{h_i(\alpha, \beta)\}_{i=1}^{n}) && (8) \\ &= \frac{1}{n} \sum_{i=1}^{n} \frac{1}{h_i(\alpha, \beta)} K\left(\frac{x - x_i}{h_i(\alpha, \beta)}\right) , && (9) \end{aligned}$$

and determine the final estimates by averaging across all possible parameter values (AKDE models) weighted by the Bayesian posterior $p(\alpha, \beta|D)$ as

$$\hat{p}(x|D) = \int_{\alpha} \int_{\beta} \hat{p}(x|D, \{h_i(\alpha, \beta)\}_{i=1}^{n}) p(\alpha, \beta|D) d\beta d\alpha \; . \tag{10}$$

Notably, this results in a pdf estimation method that does not have any user defined parameters at all as it comes with a built-in averaging across the parameters α and β. Since this estimation method is based on BMA of AKDE models we denote it *adaptive density estimation by Bayesian averaging* (ADEBA).

As outlined in detail below, one key to make this approach successful is to use the data points $D = \{x_1, ..., x_n\}$ not only as parameters in the model but also for estimating the support (likelihood) of other data points. This is achieved by rewriting the posterior $p(\alpha, \beta|D)$ by the means of Bayes' theorem as $p(\alpha, \beta|D) = p(D|\alpha, \beta)p(\alpha, \beta)/P(D)$ and approximating the value of the likelihood $p(D|\alpha, \beta)$ via a procedure similar to cross validation, in which one or several data point at a time are held out from the calculations. This type of cross-validation procedure was originally described in the context of KDE independently by Habbema et al. [17], Duin [13], Hall [18].

As has been confirmed in a large simulation study described below, ADEBA is simple, robust, and competitive. We have also successfully extended and applied this method to multi-dimensional distributions where it was found to outperform conventional Gaussian mixture modeling. An implementation of the method is provided as a package for R [33] that can be downloaded from the Comprehensive R Archive Network (CRAN). To our best knowledge, this is the first reported example of this kind of pdf estimator, and one among only a few pdf estimators for multivariate distributions reported together with a ready to use implementation.

2.3. Derivation of ADEBA

In this subsection the derivation and implementation of ADEBA is presented. Firstly, approximative Bayesian inference required to implement ADEBA is presented (Section 2.3.1), followed by a generalization from the univariate to multivariate distributions (Section 2.3.2), then the role of the pilot estimate $\hat{p}_0(x)$ is discussed in the context of generating a sequence of different ADEBA estimates (Section 2.3.3), and finally the algorithm used in the simulations (Section 3) is presented in pseudocode together with a complexity analysis (Section 2.3.4).

2.3.1. Approximative Bayesian inference

As briefly explained above, in ADEBA the basic idea is to determine the pdf estimate $\hat{p}(x|D)$ in (10). This is achieved by approximating it with standard summation as

$$\hat{p}(x|D) \approx \Delta\alpha\Delta\beta \sum_{\alpha}\sum_{\beta} \hat{p}(x|D, \{h_i(\alpha,\beta)\}_{i=1}^{n}) p(\alpha,\beta|D) \ . \tag{11}$$

$p(\alpha,\beta|D)$ is approximated by the following leave-one-out calculation

$$\begin{aligned} p(\alpha,\beta|D) &= p(D|\alpha,\beta)\frac{p(\alpha,\beta)}{p(D)} \qquad (12)\\ &\approx \left[\prod_{i=1}^{n} p(x_i|D\setminus x_i,\alpha,\beta)\right]\frac{p(\alpha,\beta)}{p(D)} \\ &= \left[\prod_{i=1}^{n}\frac{1}{n-1}\sum_{\substack{j=1\\ j\neq i}}^{n}\frac{1}{h_j(\alpha,\beta)}K\left(\frac{x_i-x_j}{h_j(\alpha,\beta)}\right)\right]\frac{p(\alpha,\beta)}{p(D)} \qquad (13)\end{aligned}$$

The values of $\alpha, \beta\Delta\alpha, \Delta\beta$ are selected using a grid search procedure explained in Section 2.3.4. Here $D \setminus x_i$ denotes the set of data points remaining when x_i has been removed from D, $p(\alpha,\beta)$ reflects our prior uncertainty about the parameter values and $p(D)$ is the conventional normalization constant. Thus the factor $p(x_i|\alpha,\beta)$ in the likelihood $p(D|\alpha,\beta) = \prod_{i=1}^{n} p(x_i|\alpha,\beta)$ is approximated as

$$p(x_i|\alpha,\beta) \approx \hat{p}(x_i|D\setminus x_i,\alpha,\beta) \tag{14}$$

where $\hat{p}(x_i|D\setminus x_i,\alpha,\beta)$ is the value at x_i for the kernel estimate obtained using the dataset $D \setminus x_i$ and the fixed parameter values α and β:

$$\hat{p}(x_i|D\setminus x_i,\alpha,\beta) = \frac{1}{1-n}\sum_{\substack{j=1\\ j\neq i}}^{n}\frac{1}{h_j(\alpha,\beta)}K\left(\frac{x_i-x_j}{h_j(\alpha,\beta)}\right) . \tag{15}$$

Leaving the data point x_i out when trying to approximate the support $p(x_i|\alpha,\beta)$ is intuitive and even necessary. If not, then there would be infinite support for zero bandwidth as $\lim_{\alpha\to 0} p(\alpha,\beta|D) = \infty$.

Each factor in (13) can be approximated more generally by the following k-fold leave out calculation:

$$p(\alpha,\beta|D,k) = \left[\prod_{l=1}^{k}\prod_{x_i\in D_l}\frac{1}{n_l}\sum_{x_j\notin D_l}\frac{1}{h_j(\alpha,\beta)}K\left(\frac{x_i-x_j}{h_j(\alpha,\beta)}\right)\right]\frac{p(\alpha,\beta)}{p(D)} \ , \tag{16}$$

where D_l denotes one among k non-overlapping subsets of D obtained by random splitting.

As always when it comes to Bayesian inference of this kind, when the number of examples n grows the products in (13) and (16) will dominate in comparisons to the prior. This would make the prior have little influence of the final estimate unless it has a pathological shape, e.g. zero in some regions, or D is very small ($n \lessapprox 25$). If no prior knowledge is available or assumptions can be made about the true values of α and β a reasonable choice is to assign a constant prior that does not favor any particular choice more than any other. A simulation study presented in supplementary section 1 further support the use of a constant prior by comparing it to two other commonly used priors designed to convey objectivity. In the simulations presented in Section 3 the prior $p(\alpha, \beta)$ for the parameters α and β was therefore set to be constant in the semi-infinite region $\{(\alpha, \beta)|\alpha > 0 \text{ and } \beta \geq 0\}$. The exact value of the constant used is irrelevant due to the normalization by $p(D)$ in (16). The reason for only accepting $\alpha > 0$ and $\beta \geq 0$ is that $\alpha \leq 0$ corresponds to invalid non-positive bandwidths and $\beta < 0$ would reverse the local factor producing larger bandwidths in dense areas and smaller bandwidths in sparse areas, which is not desirable. Note that $\beta = 0$ is perfectly acceptable and is another way of producing the non-adaptive estimator DEBA described below (Section 2.3.3).

2.3.2. Extension to multivariate KDE

It is straightforward to extend ADEBA to the multivariate pdf estimation problem in $p > 1$ dimensions. In this case we simply replace the univariate sample smoothing estimator (3) by the p-dimensional extension

$$\hat{p}(\mathbf{x}|D, h) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{(h(\mathbf{x}_i))^p} K_p \left(\frac{\mathbf{x} - \mathbf{x}_i}{h(\mathbf{x}_i)} \right) \quad . \tag{17}$$

$K_p(\mathbf{x})$ denotes a multivariate kernel function, often selected to be a multivariate Gaussian distribution function. This results in the following generalization of

(13):

$$p(\alpha,\beta|D) = \left[\prod_{i=1}^{n} \frac{1}{n-1} \sum_{\substack{j=1 \\ j\neq i}}^{n} \frac{1}{h_j^p} K_p \left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2}{h_j} \right) \right] \frac{p(\alpha,\beta)}{p(D)}. \tag{18}$$

In order to keep the calculations and the number of adjustable parameters down in our simulation experiments we have used the spherical unit variance Gaussian kernel function $K_p(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}} \exp\left(-\frac{1}{2}||\mathbf{x}||^2\right)$. No new parameters are introduced so the time complexity stays at $O(n^2)$ on any dataset. The time it takes to evaluate the collection of weighted kernels that constitutes the estimate over a grid to produce a density function surface or volume will of course increase rapidly with dimension, along with the number of grid points. This problem is not unique to ADEBA but exist for any estimator of multivariate distributions as it occurs when rendering and evaluating the function and not while estimating it.

To reduce the impact of scale and rotation of the distributions to be estimated, we pre-transformed each sample to have the identity matrix as its covariance matrix by means of principal component analysis [19, chapter 14.5] before estimating its pdf. The estimates were then reverse transformed back to the orignal space when rendered.

2.3.3. Pilot estimates and sequences of ADEBA estimates

In order to compute adaptive bandwidths (7) a pilot estimate $\hat{p}_0(x)$ of the density is needed. This pilot should resemble the unknown density to guide the adaptive factor, but even if it is somewhat misleading the estimated density will still most likely be an improvement over the pilot and can in turn be used as new pilot, allowing us to iterate until satisfaction. As illustrated in Figure 1, consider a series of iterative ADEBA based estimates $\hat{p}_t(x|D)$ each using a pilot function $\hat{p}_{0,t}(x)$ where t denotes the iteration number. If no prior knowledge is available, an objective choice of initial pilot would be $\hat{p}_{0,1}(x) = 1$ and appropriate subsequent pilots would be $\hat{p}_{0,t}(x) = \hat{p}_{t-1}(x|D)$. Thus the kernel estimator $\hat{p}_1(x|D)$ comes without any adaptive contribution, as all kernels are identical having bandwidth $h = \alpha$. This special case will therefore be referred to as *density estimation by Bayesian averaging* (DEBA) without the "A" for adaptive. The next estimator $\hat{p}_2(x|D)$, obtained using $\hat{p}_{0,2}(x) = \hat{p}_1(x|D)$ as pilot, has adaptive bandwidth like the AKDE of Abramson and will therefore be referred to as the ADEBA-1 estimator. It is of course possible to continue iterating and producing estimates but this opportunity is beyond the current scope.

This choice to use the non-adaptive estimator DEBA as pilot for ADEBA also follows the conventional approach dating back to Abramson [1], who chose a fixed bandwidth estimator as pilot for the square-root-law estimator.

2.3.4. Algorithm

The algorithm used to make ADEBA estimates is outlined in Algorim 1 and 2. Apart from the sample of data X the user must also provide a set B of equidistant β values and an interger value T that specify the number of iterative estimates (see Section 2.3.3 and Figure 1), which is $T = 1$ for DEBA and $T = 2$ for all ADEBA variants. The parameter grid used in the approximation (11) is found by an iterative search procedure implemented in the subroutine FindIntegrationGrid. For each $\beta \in B$, a suitable set A_β of equidistant α values and corresponding weights W_β is identified (line 6 of Algorithm 2). The weights are proportional to (12) but are scaled differently for each A_β in order to compensate for the different spacing of α values and to sum to 1 across all A_β.

The computational complexity of the internal functions is as follows (using $k = n$ for the cross-validation in (12) and uniform prior $p(\alpha, \beta) = 1$). Initialization is of $O(np^2+n^2p+n^3)$ due to the use of the R function `prcomp` for PCA, which is based on singular value decomposition [16]. FindIntegrationGrid is of $O((2n_{\alpha 1} + n_{\alpha 2})n_\beta n^2 p)$, $O((2n_{\alpha 1} + n_{\alpha 2})n_\beta)$ from the grid search and $O(n^2 p)$ from the weight calculation. The main function EstimateDensity is therefore of $O(np^2 + n^2 p + n^3 + T(2n_{\alpha 1} + n_{\alpha 2})n_\beta n^2 p)$. This may be considerably higher than SJ but still acceptable for practical use, as demonstrated in Section 3.

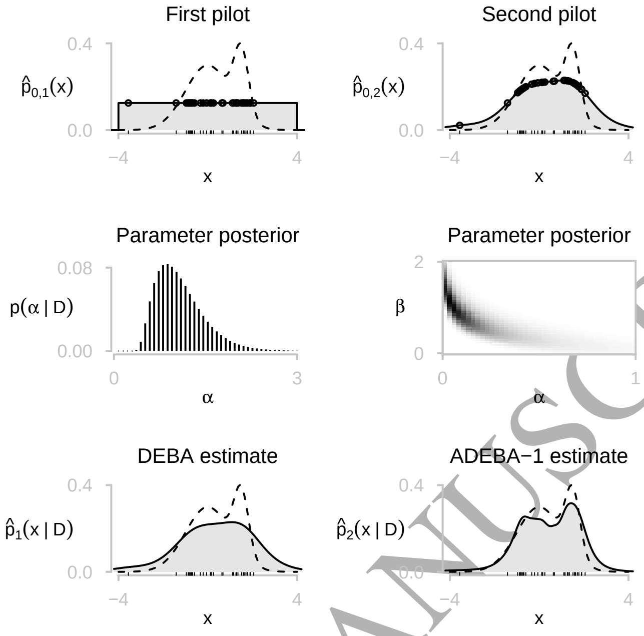

Figure 1: Graphical summary of the ADEBA pdf estimation procedure for two iterations. **Top left:** A data set D containing data points x_i (ticks above the x-axis) is sampled from an unknown distribution (dashed line). For the first iteration of the estimation, an uninformative uniform prior is used as a pilot estimate $\hat{p}_{0,1}(x)$. Since the value of the pilot function is constant at all points in the data set $\hat{p}_{0,1}(x_i) = C$ (circles). **Middle left:** It follows from (7) that with a constant pilot all kernels will get the same bandwith regardless of the choice of β. Thus there is no need to estimate the joint posterior of (α, β) for the uninformative prior, allowing the leave-one-out calculation (13) described in Section 2.3.1 to be simplified to $p(\alpha, \beta|D) = p(\alpha|D)$. **Bottom left:** Each value of α corresponds to a unique pdf estimate. All those estimates are added together weighted by their corresponding $p(\alpha|D)$ to produce a combined estimate $\hat{p}_1(x|D)$ referred to as the DEBA estimate (solid line). **Top right:** The DEBA estimate from the first iteration is used as a pilot pdf for the second iteration, this time producing different pilot function values $\hat{p}_{0,2}(x_i)$ at each data point than before (circles). **Middle right:** How much $\hat{p}_{0,2}(x_i)$ should influence the final estimate is controlled through the parameter β, whose joint posterior probability with α, $p(\alpha, \beta|D)$, is estimated with (13). The colour intensity is proportional to the posterior probability $p(\alpha, \beta|D)$. **Bottom right:** When all the pdf estimates corresponding to the different (α, β) are averaged an adaptive pdf is obtained, here denoted the ADEBA-1 estimate (bottom right). The ADEBA-1 estimate is able to amplify features of the data that was smoothed out by the non-adaptive DEBA estimator and more accurately capture the shape of the distribution's tails.

Algorithm 1 ADEBA procedure including data normalisation. Input: Raw data X and a list B containing equidistant β values. A_β refers to the column of A associated with β.

```
function EstimateDenisty($X, B, T$)
    $(m, R, D) \leftarrow$ Inititalize($X$)
    $\hat{p}_0 \leftarrow f(x|D) = 1$                                  ▷ Uniform pilot
    for $t = 1, \ldots, T$ do
        $(A, W) \leftarrow$ FindIntegrationGrid($D, B, \hat{p}_{t-1}$)
        $\hat{p}_t \leftarrow f(x|D) = \sum_{\beta \in B} \sum_{\substack{\alpha \in A_\beta \\ w \in W_\beta}} w\hat{p}(x|D, \{h_i(\alpha, \beta)\}_{i=1}^{n})$   predict function
    return $(m, R, \hat{p}_T)$
function Initialize($X$)
    $m \leftarrow$ Mean vector of the raw dataset $X$
    $\tilde{X} \leftarrow$ Center $X$ by subtracting $m$ from all observations
    $R \leftarrow$ Rotation and scaling matrix obtained by PCA
    $D \leftarrow \tilde{X}R$
    return (m, R, D)
```

Algorithm 2 Procedure for finding the integration grid used in (11). The search is controlled by a number of constants with the following default values: $\alpha_{\min} = 10^{-6}$ and $\alpha_{\max} = 10^{3}$ set the range of possible α values; $n_{\alpha 1} = 11$ and $n_{\alpha 2} = 41$ set the number of grid points in the initial rough search grids and the final fine grid; $\tau = 10^{-3}$ sets the threshold for which α values to keep when narrowing the search grid.

```
function FindIntegrationGrid($D, B, \hat{p}_0$)
    $A \leftarrow [\,]$
    $W \leftarrow [\,]$
    $w_\Sigma \leftarrow 0$
    for $\beta \in B$ do
        $A_\beta \leftarrow$ FindAlpha($D, \beta$)
        $\Delta_\alpha \leftarrow$ Distance between values in $A_\beta$
        $W_\beta \leftarrow \{\Delta_\alpha p(D|\alpha,\beta) p(\alpha,\beta|D)\}_{\alpha \in A_\beta}$   ▷ Uses pilot $\hat{p}_0$ for obtaining $h$
        $w_\Sigma \leftarrow w_\Sigma + \sum_{w_i \in W_\beta} w_i$
        $A \leftarrow [A, A_\beta]$   ▷ Add column to matrix
        $W \leftarrow [W, W_\beta]$
    $W \leftarrow W / w_\Sigma$
    return $(A, W)$
function FindAlpha($D, \beta$)
    repeat   ▷ Rough grid search
        $A_\beta \leftarrow \{\alpha_a\}_{a=1}^{n_{\alpha 1}}$ logarithmically spaced from $\alpha_{\min}$ to $\alpha_{\max}$
likelihood  $w \leftarrow \{p(D|\alpha,\beta) p(\alpha,\beta|D)\}_{\alpha \in A_\beta}$ → uniform
        $\alpha_{\min} \leftarrow \{\min(\alpha_a) \mid \alpha_a < \tau \max(w)$ and $\tau \max(w) \leq \alpha_{a+1}\}$
        $\alpha_{\max} \leftarrow \{\min(\alpha_a) \mid \alpha_{a-1} \geq \tau \max(w)$ and $\tau \max(w) > \alpha_a\}$
    until $(\alpha_{\min}, \alpha_{\max})$ updated twice
    $A_\beta \leftarrow \{\alpha_a\}_{a=1}^{n_{\alpha 2}}$ linearly spaced from $\alpha_{\min}$ to $\alpha_{\max}$   ▷ Fine grid search
    return $A_\beta$
```

3. Simulation results

The ADEBA estimators were compared to three other types of density estimators in a series of simulation studies (summarised below). All univariate estimators compared are summarised in Table 1.

1. DEBA and ADEBA-1 vs. traditional univariate AKDE, here referred to as RoT-SRL and BMA-SRL (Section 3.1). RoT is short for Silverman's *rule of thumb*, selecting α as (6). SRL is short for Abramson's *square root law*, selecting $\beta = 0.5$. Thus RoT-SRL is a completely non-Bayesian AKDE and BMA-SRL only uses BMA to select α. BMA-SRL performed better or equal to RoT-SRL on all distributions. BMA-SRL had similar performance to DEBA on most studied distributions, but was better on moderately skewed or kurtotic distributions, and worse on very complex distributions. ADEBA-1 outperformed all the other methods on distributions containing both sharp modes and long tails, but performed worse on Gaussian-like distributions.
2. DEBA and ADEBA-α vs. state-of-the-art univariate non-adative KDE, using the bandwidth selection methods of SJ and Liao (Section 3.2). ADEBA-α was an adaptive estimator that selected α with BMA and used a fixed $\beta = 1$, in contrast to the ADEBA-1 that selects both α and β with BMA. The simplification was used to reduce computation time. ADEBA-α was found to outperform SJ and Liao on distributions containing both sharp modes and long tails, but not on Gaussian-like distributions. On Gaussian-like distributions DEBA closely matched SJ and Liao in performance.
3. DEBA and ADEBA-α vs. multivariate (non-adaptive) Gaussian mixture modeling (GMM) (Section 3.3). DEBA was found to outperform ADEBA-α on all distributions. When using an optimal number of kernels and enough data GMM performed better than DEBA. However, the GMM models took considerably longer time to fit and performed worse when using a suboptimal number of kernels.

Method	Global component	Adaptive component
DEBA	α chosen with BMA	No adaptivity ($\beta = 0$)
ADEBA-1	α chosen with BMA	β chosen with BMA
ADEBA-α	α chosen with BMA	$\beta = 1$
RoT-SRL	α chosen as (6)	$\beta = 0.5$
BMA-SRL	α chosen with BMA	$\beta = 0.5$
SJ	α chosen according to [41]	No adaptivity ($\beta = 0$)
Liao	α chosen according to [26]	No adaptivity ($\beta = 0$)

Table 1: All univariate KDE methods studied in Section 3. To reduce the computational burden of ADEBA-1 only $\beta \in \{0, 0.5, 1, 1.5, 2\}$ was considered. DEBA used uniform pilots and the other methods used the DEBA estimates as pilots. The ADEBA* method mentioned in Section 3.2 was the same as ADEBA-1, but used a pilot function customised for the particular problem it was demonstrated on.

Performance was measured in terms of integrated square error (ISE) because it is intuitive and popular. In multivariate form it is defined as

$$\text{ISE}(\hat{p}(\mathbf{x}|D)) = \int (\hat{p}(\mathbf{x}|D) - p(\mathbf{x}))^2 d\mathbf{x} \tag{19}$$

Each of the simulation studies consisted of a set of distributions and sample sizes. 100 random samples were drawn for each combination of distribution and sample size, from which the different density estimation methods were used to produce 100 density estimates. An upper limit of one CPU hour was set for calculating the 100 estimates of each method, distribution, and sample size. The mean ISE was calculated from those estimates and is presented in figures 3, 4, and 7, with solid lines if at least 10 estimates were completed and dashed lines if fewer were completed.

The simulations were performed in the R environment for statistical computing [33] with source code publicly available at `http://github.com/backlin/ADEBA/`. The SJ bandwidth estimator was included in the base distribution of R, the improved estimator by Liao et al. [26] was provided by the original authors and published with their approval, and the GMM implementation of the mixtools package was used [3].

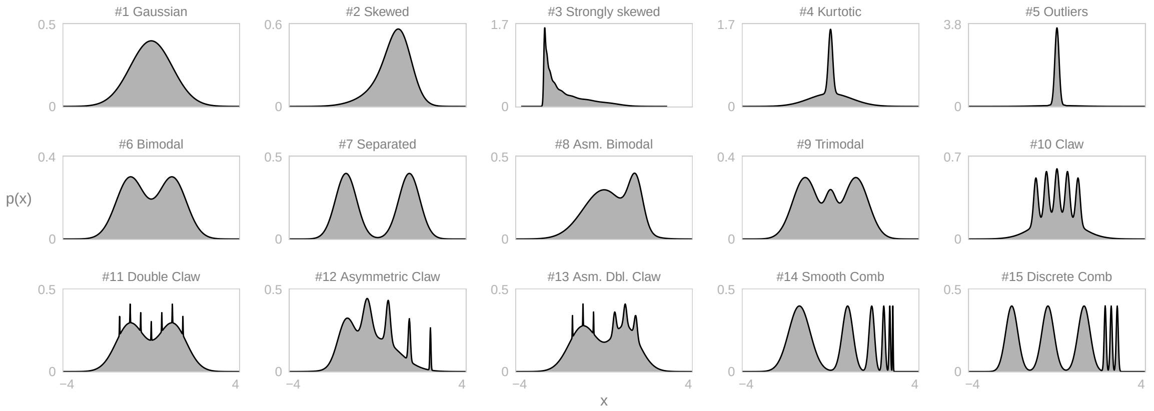

Figure 2: The Marron-Wand distributions, originally introduced in Marron and Wand [28], used here to generate samples in the numerical simulation study. Each distribution represents a situation that might appear in reality with different degree of skewness, kurtosis, outliers frequency and modality.

3.1. Comparison to traditional AKDE

A simulation study was set up to compare parameter selection with BMA to choosing α according to Silverman's rule-of-thumb (RoT) (6) and $\beta = 0.5$ according to Abramson's square root law (SRL). Random samples of size n ={50, 100, 200, 400} were generated from the 15 distributions presented in Figure 2, originally introduced in Marron and Wand [28]. Figure 3 contains four examples from the results that well illustrate the differences and similarities between the methods. The complete results for all 15 distributions are presented in Supplementary Figure 3.

BMA-SRL performed better or similar to RoT-SRL on all distributions and sample sizes. RoT-SRL did only work well on distributions whose bulk mass consisted of Gaussians of similar width and low skewness, such as #1, #6, and #11. On distributions very far from Gaussian its performance even remained unchanged with increased sample size (#3–5).

The nature of the true distribution affected which of BMA estimators had the best performance. Distributions consisting of multiple Gaussians with very different width were best estimated by ADEBA-1 (#4, #5, and #10). However, the posterior probability $p(\alpha, \beta|D)$ of ADEBA-1 was generally the highest for β values in the range of 1–1.5, i.e. considerably stronger adaptivity than SRL, which lead to reduced performance on less extreme distributions. DEBA and BMA-SRL had similar overall performance, but DEBA was slightly better on very complex densities (#12, #14, and #15) and BMA-SRL on skewed densities (#2 and #3).

3.2. Comparisons with state-of-the art non-adaptive estimators

In a larger simulation study, the ADEBA estimators were compared to non-adaptive KDE using the state-of-the-art bandwidth selectors of Sheather and Jones [41] (SJ) and Liao et al. [26] (Liao). Because the BMA-SRL estimator produced similar results as the DEBA estimator in the previous simulation (Section 3.1), we decided to instead use $\beta = 1$ and study the potential benefits of a stronger adaptive effect. This estimator is referred to as ADEBA-α to distinguish it from ADEBA-1 used above. Samples of sizes $n = \{25, 50, 100, 200, 400, 800, 1600\}$ were again drawn from the Marron-Wand distributions (Figure 2), each with 100 replicates. Selected highlights of the results are presented and discussed in Figure 4 and the remainder of this section. The complete results for all 15 distributions are presented in Supplementary figure 4.

The adaptive estimator ADEBA-α was only advantageous on certain distributions: Those for which there was a clear need for varying the bandwidth across the space to get the most of the data (Figure 4 middle two rows). On such distributions, DEBA compromised between modes and tails and selected intermediate bandwidths whereas SJ and Liao selected bandwidths only suitable for the mode. The strategy of SJ and Liao resulted in a lower mean ISE compared to the strategy of DEBA but it produced many spurious modes in the tails as an undesirable side-effect (Figure 4 second row).

On very complex distributions ADEBA-α performed worse than the non-adaptive estimators (Figure 4 bottom row), because there wasn't enough data

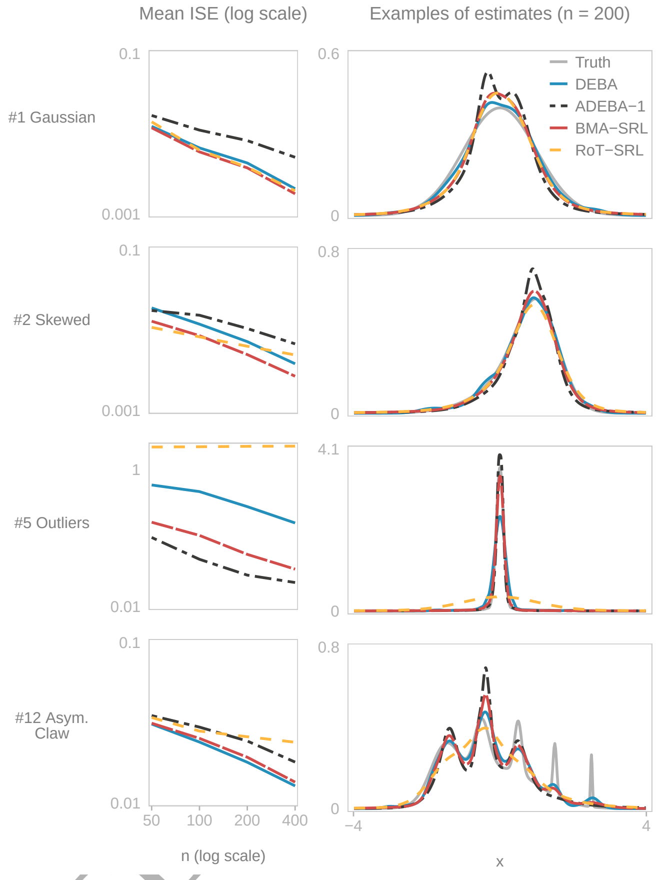

Figure 3: Four examples from the comparison of DEBA, ADEBA-1 and the traditional adaptive density estimators of Abramson and Silverman (see table 1). **#1:** DEBA, BMA-SRL, and RoT-SRL performed similarly on a purely Gaussian distribution. ADEBA-1 tended to produce overly adaptive estimates which caused it perform worse than the others. **#2:** BMA-SRL outperformed the other methods on this moderately skewed distribution. RoT-SRL improved slowly with increased sample size, DEBA couldn't simultaneously get both mode and tails right, and ADEBA-1 overshot the mode. **#5:** ADEBA-1 outperformed the other estimators on this highly skewed distribtuion. Data points in the tails had a large negative impact on the bandwidths selected by DEBA and BMA-SRL. RoT-SRL completely failed to even detect the rough shape of the distribution. **#12:** DEBA outperformed the other estimators on complex distributions. Although the true distributions have multiple modes of different width none of the adaptive estimators were able to efficiently capture that.

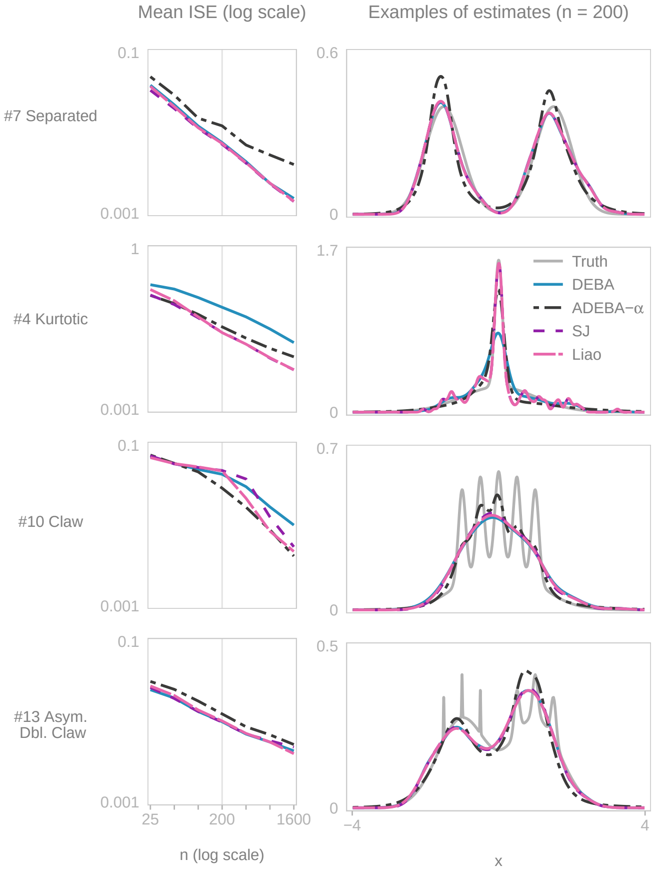

Figure 4: Highlights from the comparison of the ADEBA estimators to the state-of-the-art non-adaptive pdf estimators SJ and Liao. The Marron-Wand distributions (Figure 2) can broadly be grouped into three categories based on the characteristics of their bulk mass: Gaussian-like, unimodal (1–2) or multimodal (6–9); considerably skewed (3–5); and complex (10–15). **Top row:** On Gaussian-like distributions, like the separated distribution #7, the non-adaptive estimators (DEBA, SJ, and Liao) had very similar performance and outperformed the adaptive ADEBA-α estimator. This was not surprising since all the non-adaptive estimators are essentially designed for estimating Gaussians well. On many of these distributions ADEBA-α required about twice as many observations as the non-adaptive estimators for equal performance. **Second row:** On skewed distributions, like the kurtotic distribution #4, ADEBA-α demonstrated its ability to vary and adapt the bandwidth to the local data density. SJ and Liao selected bandwidths that were only suitable for the mode, resulting poorly estimated tails with many spurious modes. This qualitative drawback did not make a big difference in terms of ISE but persist even into large sample sizes. DEBA typically compromised on bandwidth size between mode and tails resulting in worse performance than SJ and Liao. **Bottom two rows:** On the complex distributions the adaptive property of ADEBA-α was the most advantageous on the claw distribution #10. The even more complex distributions, like the asymmetric double claw #13, could not be estimated in full detail by any method. This was because they contain modes of such small mass that not even samples of size $n = 1600$ allowed them to be detected.

for it to discern the underlying structure even with the largest samples sizes. Detection of fine details at reasonable sample sizes may however be possible by incorporating prior information in the estimator, a strong advantage of the Bayesian framework compared to SJ and Liao. For example, the comb distributions #14–15 have structures that are only likely to be encountered in particular applications. If it was known from the analysis context that the smooth comb distribution #14 contained many modes of decreasing width the standard bandwidth transformation function (7) can easily be replaced with a function h^* that takes this into consideration, for example

$$h^*(x_i, \alpha, \beta) = \alpha e^{-\beta x_i} \quad . \tag{20}$$

The exponential factor of (20) produces decreasing bandwidths with increasing x, with a rate of decrease controlled by β. Such a function can be directly plugged into the existing ADEBA framework, producing an estimator ADEBA*, which outperforms the all other estimators (Figure 5). For this example, ADEBA* achieved the lowest ISE and was the only estimator that correctly identified all six modes without introducing any spurious modes.

3.3. Multivariate density estimation

A detailed study of the performance of ADEBA for multivariate distributions is beyond the current scope. As a preliminary evaluation the ADEBA estimators were compared to conventional Gaussian mixture modeling (GMM) based on the classical expectation maximization. Samples of sizes $n = \{25, 50, 100, 200, 400, 800, 1600\}$ were drawn randomly from nine 2-d and 3-d distributions (Figure 6) in 100 replicates each. GMM estimates were calculated with a number of kernels $k \in \{2, 3, 5, 8\}$. Although k is the only hyperparameter of GMM that must be specified by the user (or tuned using some objective criterion) a total of $n_\theta = k\left(1 + d + \frac{d^2+d}{2}\right)$ internal parameters must be estimated from the data to produce a final estimate: 1 for each kernel's weight, d for each mean, and $\frac{d^2+d}{2}$ for the covariance matrix (where d is the dimension). A consequence of this is that no model can be fitted unless $n \geq n_\theta$, e.g. 12 examples are needed to fit a

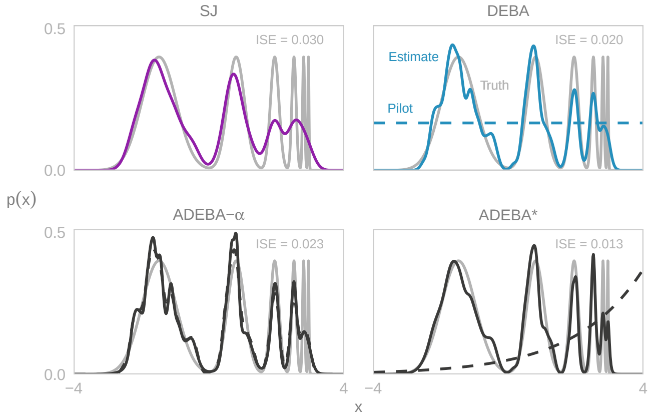

Figure 5: A comparison of the estimators used in the univariate simulation study and the improved estimator ADEBA*. Because ADEBA* incorporates prior knowledge about the pdf it outperformed all of the other estimators. Each panel shows a pdf estimate (solid non-gray lines) by a different estimator (panel titles) together with the smooth comb pdf that is to be estimated (gray). The dashed lines show the pilot function used for each estimator (mostly hidden behind the estimate for ADEBA-α). The scaling of the pilot is irrelevant as it will be tuned by the α parameter. The pilot of ADEBA* is $\hat{p}_0(x) = e^x$, which replaces the standard bandwidth transformation function (7) with (20). Both α and β of ADEBA* were tuned with BMA.

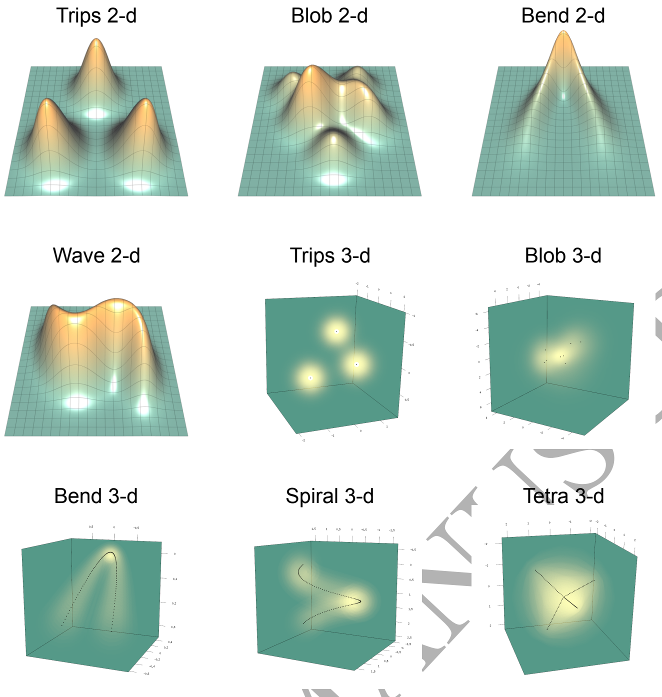

Figure 6: Visualizations of the 2-d and 3-d densities used in the simulations. The densities were chosen to pose different problems for the estimators in terms of shape, modality and mass of tails. All consisted of collections of Gaussian kernels of varying location, width and weight (whose centers are marked by black dots in 3-d). Trips and blob consists of 3 and 8 Gaussians and the remaining consist of 100. The mathematical definitions are provided in Section 2 of the supplementary material.

GMM in 2-d using $k = 3$ kernels. We only considered models with $n \geq 2n_\theta$ since estimates with less than 2 data points per parameter performed very poorly.

Because GMM uses randomly selected starting points for its kernels repeated estimates were performed for each sample and choice of k until 3 successful estimates had been obtained or 5 estimates had failed to converge. The ISE was then calculated for the estimate with the highest likelihood. GMM is a computationally demanding and this further increased the computation time.

None of the estimators emerged as winner on all distributions. GMM outperformed ADEBA on trips, for which GMM3 was the optimal estimator; ADEBA outperformed GMM on tetra, for which GMMs were particularly unsuitable; and the results were mixed on the remaining distributions. Overall, the ISE was higher compared to univariate estimation (Figure 7) and in most situations the simpler methods performed better than the more complex (DEBA outperformed ADEBA-α and GMM2 or 3 outperformed GMM5 and 8). Aside from the performance in terms of ISE, ADEBA demonstrated a few practical advantages over GMM: the ability to produce estimates on all sample sizes within shorter computation time (Supplementary Figure 5), without the need for parameter selection. The parameter selection issue is particularly visible on the blob distributions, where GMM2 had the best performance on small samples, GMM3 took over at $n = 200$, GMM5 took over at $n = 400$, and GMM8 never managed to catch up even though it was the theoretically ideal estimator for the problem.

4. Discussion

The ADEBA family of estimators unifies many desirable properties for pdf estimators while at the same time performing well. Compared to current state-of-the-art univariate estimators, the adaptive version of ADEBA performed better on distribution with long or heavy tails and the non-adaptive version of ADEBA performed similarly. However, the state-of-the-art estimators tended to select bandwidths only suitable to the mode of the distribution producing many spurious modes in the tails. This was not the case for the ADEBA estimators.

A preliminary multivariate evaluation showed that ADEBA performed better than GMM on complex multivariate distributions, but worse on simple distributions. It thereby makes an important contribution to multivariate pdf estimation, a topic that has received far less attention than for the case of univariate distributions. The perhaps most important reasons for this is the dramatic increase in complexity for most estimators and performance measures. As the dimensionality increases, it also becomes difficult to suggest sensible or

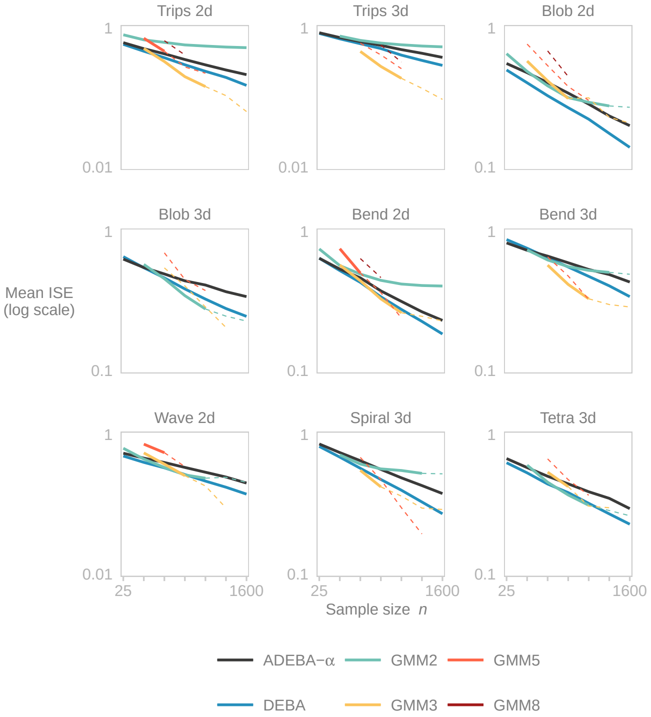

Figure 7: Multivariate density estimation performance for the 6 estimators presented in Section 3.3. Solid lines indicate that a method managed to produce estimates for at least 10 samples within one hour. Dashed lines indicate that fewer estimates were produced.

even stable ways to calculate the distance between a query point x and the kernels of a KDE [2, 21], even for the most basic estimator (1). Beyer et al. [5] provided an illustrative example of this where it is demonstrated that the distance between a query point and its nearest neighbor approaches the distance to its furthest neighbor as the dimensionality increases, rendering all attempts to estimate high dimensional densities futile. The impact of this problem can be reduced by compressing the data into a smaller subspace before pdf estimation, e.g. by methods like NMF [46] or NSSRD [39].

ADEBA is easy to use since it computes more quickly than GMM and provides the possibility to automatically tune all its parameters from the data. It is still possible to specify the parameters manually for increased control over adaptivity and for incorporating prior knowledge about the distribution to be computed.

4.1. Limitations

Due to the complexity of ADEBA it seems intractable to obtain a theoretical convergence analysis. Moreover, the computational complexity might become a limiting factor in certain applications.

It should be noted that several improvements of the conventional GMM method used in our comparison of multivariate estimatorshave have been suggested. To name a few, [48] reviewed various way for automatic selection of the number of kernels, and [29] reported performance improvements by introducting regularization through conjugate priors and averaging multiple estimates on each sample of data. These multiple estimates may also be fitted with different subsets of data, producing an ensemble estimator. Since our multivariate comparison is mainly intended as a proof-of-principle of multivariate extension these many flavours of GMM have not been considered.

4.2. Promising alternatives to ADEBA

ADEBA is not the only alternative that demonstrates great potential of replacing standard KDE. For example, Katkovnik and Scmulevich [25] demonstrate outstanding potential of using the so-called intersection of confidence intervals rule to determine the kernel size $h(x)$ for each evaluation point x. This method belongs to the family of balloon estimators discussed in the introduction and was evaluated only for a single simulated problem, a double peaked density. Results from that study show that it not only outperforms the Sheather-Jones estimator but in fact any constant-bandwidth method including an idealized method which always is able to select the optimal constant bandwidth which minimizes the expected mean squared error. Since it has only been evaluated for one problem, its practical potential needs to be confirmed across a wider range of problems and a generalization from univariate densities to multivariate densities should be developed. Anyway, these results already supports our conclusion that the potential of using adaptive kernels deserves future attention. One should also note the AKDE based on linear diffusion developed by Botev et al. [7], which comes with many attractive properties.

5. Conclusions

The results of our study show that the ADEBA method introduced here offers attractive features compared to the best density estimators of today. On Gaussian distributions it matches the performance of current state-of-the-art methods, and on non-Gaussians it excels beyond them. It is simple in construction, robust, not relying on any user defined parameters, and offers the posiblity to integrate prior knowledge into the model. Moreover, it is not restricted to univariate distributions but allows competitive estimation for multivariate distributions within reasonable time, which is a quite unusual feature. More generally the results provided shows that ADEBA offers a promising new alternative for the fundamental problem of pdf estimation.

Conflict of interest

None declared.

We thank Jason Liao for kindly providing his R-function for selecting bandwidths. This work was supported by the foundation for strategic research

(RBc08-008), the EU FP-7 project PROACTIVE, and the Swedish Research Council (621-2008-5854). In the early stages of this work, computational power was in part kindly provided by Ulf Hammerling and Daniel Soeria-Atmadja at the Swedish National Food Agency. Computational resources were also provided by Uppmax (Uppsala Multidisciplinary Center for Advanced Computational Science) at Uppsala University, which is also greatly acknowledged.

References

[1] Abramson, I. S., Dec. 1982. On bandwidth variation in kernel estimates-a square root law. The Annals of Statistics 10 (4), 1217–1223.
URL `http://projecteuclid.org/euclid.aos/1176345986`

[2] Aggarwal, C. C., Mar. 2001. Re-designing distance functions and distance-based applications for high dimensional data. ACM SIGMOD Record 30 (1), 13–18.
URL `http://portal.acm.org/citation.cfm?doid=373626.373638`

[3] Benaglia, T., Chauveau, D., Hunter, D. R., Young, D., 2009. mixtools: An r package for analyzing finite mixture models. Journal of Statistical Software 32 (6), 1–29.
URL `http://www.jstatsoft.org/v32/i06/`

[4] Berjón, D., Cuevas, C., Morán, F., García, N., 2018. Real-time non-parametric background subtraction with tracking-based foreground update. Pattern Recognition 74, 156–170.

[5] Beyer, K., Goldstein, J., Ramakrishnan, R., Shaft, U., 1999. When is "nearest neighbor" meningful? In: In Int. Conf. on Database Theory. pp. 217—-235.

[6] Bordes, A., Weston, J., Collobert, R., Bengio, Y., 2011. Learning structured embeddings of knowledge bases antoine. In: Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence Learning. pp. 301–306.

[7] Botev, Z., Grotowski, J., , Kroese, D., 2010. Kernel density estimation via diffusion. Ann. Statist. 38, 2916–2957.

[8] Breiman, L., Meisel, W., Purcell, E., 1977. Variable kernel estimates of multivariate densities. Technometrics 19 (2), 135–144.
URL `http://www.jstor.org/stable/1268623`

[9] Brewer, M. J., 2000. A bayesian model for local smoothing in kernel density estimation. Statistics and Computing 10, 299–309.
URL `http://www.springerlink.com/content/p4183478210v142x/`

[10] Chacon, J., 2013. Data-driven density derivative estimation with applications to nonparametric clustering and bump hunting. Electronic Journal of Statistics 7, 499–532.

[11] Chen, T., Lu, H., Lee, Y., Lan, H., Dec. 2008. Segmentation of cdna microarray images by kernel density estimation. J Biomed Inform. 41, 1021–7.

[12] De Marco, T., Cazzato, D., Leo, M., Distante, C., 2015. Randomized circle detection with isophotes curvature analysis. Pattern Recognition 48 (2), 411–421.
URL `http://dx.doi.org/10.1016/j.patcog.2014.08.007`

[13] Duin, R. P., 1976. On the choice of smoothing parameters for parzen estimators of probability density functions. IEEE Trans Comp C-25, 1175–9.

[14] Fix, E., Hodges, J., Dec. 1989. Nonparametric discrimination consistency properties. International Statistical Review 57, 238–247.

[15] Gao, W., Kannan, S., Oh, S., Viswanath, P., 2017. Estimating mutual information for discrete-continuous mixtures. In: Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 30. Curran Associates, Inc., pp. 5988–5999.
URL `http://papers.nips.cc/paper/7180-estimating-mutual-information-for-discrete-continu`
`pdf`

[16] Golub, G. H., Van Loan, C. F., 1996. Matrix Computations (3rd Ed.). Johns Hopkins University Press, Baltimore, MD, USA.

[17] Habbema, J., Hermans, J., van den Broek, K., 1974. A stepwise discriminant analysis program using density estimation. In: Compstat 1974, Proceedingsin Computational Statistic. Physica Verlag, Vienna, pp. 101–110.

[18] Hall, P., 1982. Cross-validation in density estimation. Biometrika 69, 383–90.

[19] Hastie, T., Tibshirani, R., Friedman, J., 2009. The elements of statistical learning, 2nd Edition. Springer series in statistics. Springer. URL `http://www-stat.stanford.edu/~tibs/ElemStatLearn/`

[20] Hoeting, J. A., Madigan, D., Raftery, A. E., Volinsky, C. T., 1999. Bayesian model averaging: A tutorial. Statistical Science 14, 382–417.

[21] Hsu, C.-M., Chen, M.-S., Apr. 2009. On the design and applicability of distance functions in high-dimensional data space. IEEE Transactions on Knowledge and Data Engineering 21 (4), 523–536. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4609382`

[22] Hu, S., Poskitt, D., Zhan, X., Mar. 2012. Bayesian adaptive bandwidth kernel density estimation of irregular multivariate distributions. Computational Statistics & Data Analysis 56, 732–740.

[23] Ji, P., Zhao, N., Hao, S., Jiang, J., 2014. Automatic image annotation by semi-supervised manifold kernel density estimation. Information Sciences 281, 648–660. URL `http://dx.doi.org/10.1016/j.ins.2013.09.016`

[24] Juban, J., Fugon, L., Kariniotakis, G., May 2007. Probabilistic short-term wind power forecasting based on kernel density estimators. In: European Wind Energy Conference and exhibition, EWEC. pp. 1–11.

URL https://hal-mines-paristech.archives-ouvertes.fr/hal-00526011

[25] Katkovnik, V., Shmulevich, I., Dec. 2002. Kernel density estimation with adaptive varying window size. Pattern recognition letters 23 (14), 1641–1648.
URL http://linkinghub.elsevier.com/retrieve/pii/S0167865502001277

[26] Liao, J. G., Wu, Y., Lin, Y., Jan. 2010. Improving sheather and jones' bandwidth selector for difficult densities in kernel density estimation. Journal of Nonparametric Statistics 22 (1), 105–114.
URL http://www.informaworld.com/openurl?genre=article&doi=10.1080/10485250903194003&magic=crossref||D404A21C5BB053405B1A640AFFD44AE3

[27] Loftsgaarden, D., Quesenberry, C., 1965. A nonparametric estimate of a multivariate density function. The Annals of Mathematical Statistics 36 (3), 1049–1051.
URL http://www.jstor.org/stable/2238216

[28] Marron, J., Wand, M., 1992. Exact mean integrated squared error. The Annals of Statistics 20 (2), 712–736.
URL http://www.jstor.org/stable/2241980

[29] Ormoneit, D., Tresp, V., 1996. Improved gaussian mixture density estimates using bayesian penalty terms and network averaging. In: Touretzky, D. S., Mozer, M. C., Hasselmo, M. E. (Eds.), Advances in Neural Information Processing Systems 8. MIT Press, pp. 542–548.
URL http://papers.nips.cc/paper/1036-improved-gaussian-mixture-density-estimates-using-pdf

[30] Parzen, E., 1962. On estimation of a probability function and mode. Annals of Mathematical Statistics 33 (3), 1065–1076.

[31] Pereira, O., Torres, E., Garcés, Y., Rodríguez, R., 2017. Edge detection based on kernel density estimation. In: Arabnia, H. R., Deligiannidis, L., Tinetti, F. G. (Eds.), Int'l Conf. on Image Processing, Computer Vision, and Pattern Recognition. CSREA Press, Las Vegas, Nevada, pp. 123–128.
URL https://csce.ucmss.com/cr/books/2017/LFS/CSREA2017/IPC3334.pdf

[32] Pérez, A., Larrañaga, P., Inza, I., 2009. Bayesian classifiers based on kernel density estimation: Flexible classifiers. International Journal of Approximate Reasoning 50, 341–362.
URL http://www.sciencedirect.com/science/article/pii/S0888613X08001400

[33] R Core Team, 2015. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.
URL http://www.R-project.org/

[34] Rhodes, A. D., Quinn, M. H., Mitchell, M., 2016. Fast on-line kernel density estimation for active object localization. CoRR abs/1611.05369.
URL http://arxiv.org/abs/1611.05369

[35] Rosenblatt, M., 1956. Remarks on some nonparametric estimates of a density function. Annals of Mathematical Statistics 27 (3), 832–837.
URL http://projecteuclid.org/DPubS?service=UI&version=1.0&verb=Display&handle=euclid.aoms/1177728190

[36] Sain, S. R., Scott, D. W., Dec. 1996. On locally adaptive density estimation. Journal of the American Statistical Association 91 (436), 1525.
URL http://www.jstor.org/stable/2291578?origin=crossref

[37] Santhanam, V., Morariu, V. I., Harwood, D., Davis, L. S., 2016. A non-parametric approach to extending generic binary classifiers for multi-classification. Pattern Recognition 58, 149–158.

[38] Sebestyen, G., 1962. Pattern recognition by an adaptive process of sample set construction. IRE Transactions on Information Theory 8, 82–91.

[39] Shang, R., Zhang, Z., Jiao, L., Wang, W., Yang, S., Jul. 2016. Global discriminative-based nonnegative spectral clustering. Pattern Recognition 55, 172–182.

[40] Sheather, S. J., Nov. 2004. Density estimation. Statistical Science 19 (4), 588–597.
URL `http://projecteuclid.org/Dienst/getRecord?id=euclid.ss/1113832723/`

[41] Sheather, S. J., Jones, M. C., 1991. A reliable data-based bandwidth selection method for kernel density estimation. Journal of the Royal Statistical Society. Series B (Methodological) 53 (3), 683–690.
URL `http://www.jstor.org/stable/2345597`

[42] Silverman, B., Jones, M., Dec. 1989. E. fix and j.l. hodges (1951): An important contribution to nonparametric discriminant analysis and density estimation. International Statistical Review 57, 233–238.

[43] Silverman, B. W., 1986. Density Estimation for Statistics and Data Analysis., 1st Edition. Monographs on statistics and applied probability. Chapman and Hall.
URL `http://www.worldcat.org/isbn/0412246201`

[44] Sousa de Lima, M., Saravia Atuncarb, G., 2011. A bayesian method to estimate the optimal bandwidth for multivariate kernel estimator. Journal of Nonparametric Statistics 23, 137–148.

[45] Spampinato, C., Palazzo, S., Kavasidis, I., 2014. A texton-based kernel density estimation approach for background modeling under extreme conditions. Computer Vision and Image Understanding 122, 74–83.
URL `http://dx.doi.org/10.1016/j.cviu.2013.12.003`

[46] Sra, S., Dhillon, I. S., 2006. Generalized nonnegative matrix approximations with bregman divergences. In: Weiss, Y., Schölkopf, B., Platt, J. C. (Eds.), Advances in Neural Information Processing Systems 18. MIT Press, pp. 283–290.
URL http://papers.nips.cc/paper/2757-generalized-nonnegative-matrix-approximations-with
pdf

[47] Terrell, G., Scott, D., 1992. Variable kernel density estimation. The Annals of Statistics 20 (3), 1236–1265.
URL http://www.jstor.org/stable/2242011

[48] the number of components in a Gaussian mixture model, O., Sep. 2014. Mclachlan, geoffrey j. and rathnayake, suren. Data Mining and Knowledge Discovery archive 4, 341–355.

[49] Vergara, J. R., Estévez, P. A., 2014. A review of feature selection methods based on mutual information. Neural Computing and Applications 24 (1), 175–186.

[50] Wang, L., Lu, J., Li, X., Huan, Z., Liang, J., Chen, S., 2017. Learning arbitrary-shape object detector from bounding-box annotation by searching region-graph. Pattern Recognition Letters 87, 171–176.
URL http://dx.doi.org/10.1016/j.patrec.2016.06.022

[51] Wu, T.-J., Chen, C.-F., Chen, H.-Y., Feb. 2007. A variable bandwidth selector in multivariate kernel density estimation. Statistics & Probability Letters 77 (4), 462–467.
URL http://linkinghub.elsevier.com/retrieve/pii/S0167715206002665

[52] Zambom, A. Z., Dias, R., 2013. A review of kernel density estimation with applications to econometrics. International Econometric Review 5, 20–42.

[53] Zeinali, Y., Story, B. A., 2017. Competitive probabilistic neural network. Integrated Computer-Aided Engineering 24, 105–118.

[54] Zhang, X., King, M. L., Hyndman, R. J., Jul. 2006. A bayesian approach to bandwidth selection for multivariate kernel density estimation. Computational Statistics & Data Analysis 50 (11), 3009–3031.
URL http://linkinghub.elsevier.com/retrieve/pii/S0167947305001362

[55] Zivkovic, Z., Van Der Heijden, F., 2006. Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recognition Letters 27 (7), 773–780.

[56] Zougab, N., Adjabi, S., Kokonendji, C. C., 2013. Adaptive smoothing in associated kernel discrete functions estimation using bayesian approach. Journal of Statistical Computation and Simulation 83, 2219–2231.

[57] Zougab, N., Adjabi, S., Kokonendji, C. C., 2013. A bayesian approach to bandwidth selection in univariate associate kernel estimation. Journal of Statistical Theory and Practice 7, 8–23.

[58] Zougab, N., Adjabi, S., Kokonendji, C. C., 2014. Bayesian estimation of adaptive bandwidth matrices in multivariate kernel density estimation. Computational Statistics and Data Analysis 75, 28–38.

Biography

Christofer Bäcklin have a BSc and MSc in bioinformatics and defended a PhD thesis in medical bioinformatincs in 2015. Since then he has worked as a data scientist, first at at Ocado Technology in the United Kingdom and then at Formulate AB in Stockholm, specializing in machine learning for retail applications.

Claes Andersson received the Ph.D degree in bioinformatics from Uppsala University, Sweden, in 2008. His current research interests include using pattern recognition and statistical modelling for personalized and precision medicine as well as algorithms for accelerated drug development.

Mats Gustafsson holds the first Swedish professorship in Medical Bioinformatics (2008). Having a strong engineering background as a former professor of signal processing, he is currently working on new integrated experimental-computational approaches for accelerated drug development and disease understanding including biomarker identification and multivariate data analysis using machine learning.