This is the Pre-Published Version.
Wang, G., Choi, K. -., & Deng, Z. (2016). Noise-benefit FRSDE for speedup of density estimation on large data. Journal of Intelligent and Fuzzy Systems, 30(1), 443-450. The final publication is available at IOS Press through http://dx.doi.org/10.3233/IFS-151768.

Noise-benefit FRSDE for Speedup of Density Estimation on Large Data

Guanjin Wang[a], Kup-Sze Choi[a], Zhaohong Deng[b,*]
[a] *Centre for Smart Health, School of Nursing, Hong Kong Polytechnic University, Hong Kong*
[b] *School of Digital Media, Jiangnan University, Wuxi, Jiangsu, P.R. China*

Abstract. Fast Reduced set density estimator (FRSDE) is an important technique to realize the fast kernel density estimation based on the fast minimal enclosing ball (MEB) approximation technique. However, its performance on the running time is severely affected by the approximation parameter ε used in this algorithm, where a smaller value will lead to more accurate approximation but heavy learning burden. In this study, we reveal that the random Gaussian white noise manually added to the data will speed up the learning and accordingly propose a speedup version of FRSDE, i.e., the noise-benefit FRSDE (NB-FRSDE). NB-FRSDE can realize such a speedup because a larger value of ε can be used on the noisy version of the original data to obtain the equivalent approximation performance, which only can be obtained by FRSDE on the original data with a smaller value of ε. The distinctive characteristics of NB-FRSDE exist in the following aspects: (1) its implementation is very simple because NB-FRSDE is the same as FRSDE except that there are Gaussian noises manually added to the original data in NB-FRSDE. (2) While most of the existing machine learning methods always try to remove the noise in order to overcome the influence of noise, NB-FRSDE benefits from the manually added noise in the sense of the average running time. The experimental studies on density estimation and its application to image segmentation demonstrate the above advantages.

Keywords: Fast reduced set density estimator, Minimal enclosing ball, Core set, Noise-benefit, NB-FRSDE

1. Introduction

Probability density estimation is very important in many fields, such as pattern recognition and machine learning [1, 2, 14-16]. Parzen window kernel density estimator is one of the most attractive nonparametric density estimation methods [3-7, 17-19], which uses a full reference set for computing the estimate of a given datum. On the other hand, it is prohibitively expensive for online testing purposes in the practical applications for large data environments [8]. The reduced set density estimator (RSDE) [9] is another quite effective density estimation method due to its distinctive features, that it only needs both time and space complexities $O\left(N^2\right)$ to estimate the weighting coefficients. Although RSDE shows a nicer performance, a critical challenge to RSDE is its long time and space complexities in learning the weighting coefficients of a model. Fast Reduced Set Density Estimator (FRSDE), therefore, has been proposed as a fast version of RSDE in order to overcome the shortcoming [10]. The FRSDE has the following distinctive features: 1) The upper bound of its time complexity is asymptotically linear with the size N of a data set and the space complexity is independent of N, and 2) FRSDE can usually obtain comparable density accuracy with much higher data condensation rates than RSDE. However, the performance of FRSDE is severely influenced by the approximation parameter ε used in its learning algorithm, where a smaller value leads to a more accurate approximation but takes expensive computational cost. In this study we will address this problem by using noise-benefit technique and present the improved method for the speedup of the density estimation on large datasets.

We reveal that the random Gaussian white noise manually added to the data will result in a further speedup for FRSDE due to an obtained observation that a larger value of ε can lead to the equivalent approximation performance based on the noisy data.

*Corresponding author. Zhaohong Deng. Tel.: +86 510 85915666; Fax: +86 510 85913570; E-mail: dzh666828@aliyun.com.

This only can be obtained by FRSDE with a smaller value of ε on the original data. Based on the above observation, a noise-benefit FRSDE (NB-FRSDE) algorithm is proposed. The NB-FRSDE can be implemented in a very simple way because it is the same as FRSDE except that the Gaussian noises are added to the original data. Moreover, NB-FRSDE can benefit from the noise added manually in the sense of the average running time, which is very different from the most of the existing machine learning methods which always try to suppress the influence of noise. The experimental studies on density estimation and the application to image segmentation will demonstrate the above advantages.

The rest of this paper is organized as follows. Section 2 describes RSDE and FRSDE in brief. In section 3, the noise-benefit FRSDE is proposed to obtain a speedup version of FRSDE. Experimental studies are reported in Section 4, and conclusions are given in the last section.

2. RSDE and FRSDE

2.1. RSDE

RSDE was first proposed in [9] for kernel density estimation. The general form of a kernel density estimator can be denoted as $\hat{p}(\mathbf{x};h,\boldsymbol{\gamma})=\sum_{i=1}^{N}\gamma_i K_h(\mathbf{x},\mathbf{x}_i)$ based on a given dataset $S=\{\mathbf{x}_1,\cdots,\mathbf{x}_N\}\in R^d$, with $K_h(\mathbf{x},\mathbf{x}_i)$ as the kernel function. The solution of RSDE can be described as a QP problem with the following form:

$$\begin{aligned}&\underset{\boldsymbol{\gamma}}{\operatorname{argmax}}\ \ 2\boldsymbol{\gamma}^T\mathbf{p}-\boldsymbol{\gamma}^T\mathbf{C}\boldsymbol{\gamma}\\ &s.t.\ \ \boldsymbol{\gamma}^T\mathbf{1}=1,\ \ \gamma_i\ge 0,\ \ \forall\ i\end{aligned} \tag{1}$$

In Eq.(1), $\mathbf{C}$ is a $N\times N$ matrix with $C(\mathbf{x}_i,\mathbf{x}_j)=\int_{R^d}K_h(\mathbf{x};\mathbf{x}_i)K_h(\mathbf{x};\mathbf{x}_j)d\mathbf{x}$; $\mathbf{p}$ is a $N\times 1$ vector of Parzen density estimates with $p(\mathbf{x}_i)=\frac{1}{N}\sum_{j=1}^{N}K_h(\mathbf{x}_i,\mathbf{x}_j)$, i.e., $\mathbf{p}=\mathbf{K}_h\mathbf{1}_N$, where $\mathbf{1}_N$ is the $N\times 1$ vector whose elements are all $1/N$; $\mathbf{1}$ is the $N\times 1$ vector whose elements are all 1. Although the exact shape of the kernel function does not affect the approximation greatly a lot, Gaussian density function $G_h(\mathbf{x},\mathbf{x}_i)$ is the most adopted kernel function. When $G_h(\mathbf{x},\mathbf{x}_i)$ is adopted, $C(\mathbf{x}_i,\mathbf{x}_j)=\int_{R^d}K_h(\mathbf{x};\mathbf{x}_i)K_h(\mathbf{x};\mathbf{x}_j)d\mathbf{x}$ can be directly denoted as $G_{2h}(\mathbf{x}_i,\mathbf{x}_j)$, which results in the easier computation for $\mathbf{C}$. By solving the above QP problem, the final density estimator can be represented as

$$\hat{p}(\mathbf{x};h,\boldsymbol{\gamma})=\sum_{m=1}^{M}\gamma_m K_h(\mathbf{x},\mathbf{x}_m) \tag{2}$$

with $\mathbf{x}_m\in Sr$, where $Sr=\{\mathbf{x}_i\mid\gamma_i>0,i=1,2,\cdots,N\}$ is the reduced set of RSDE and $M=|Sr|$ is the size of the reduced data.

2.2. FRSDE

RSDE is very time consuming on large datasets. In [10], the fast version on large datasets, i.e., FRSDE, is proposed by using the MEB approximation strategy and the core-set technique.

(a) *Core-set-based MEB approximation*

A ball can be denoted as $B(\mathbf{c},R)$ with center and radius as $\mathbf{c}$ and R, respectively. Given a data set $S=\{\mathbf{x}_1,\cdots,\mathbf{x}_N\}\in R^d$, the minimum enclosing ball of S can be denoted as MEB(S), which is the smallest ball that contains all the data in S. The MEB can be solved by the following constrained optimization problem:

$$\begin{aligned}&\underset{\mathbf{c},R}{\operatorname{argmin}}\ \ R^2\\ &s.t.\ \ (\mathbf{x}_i\text{-}\mathbf{c})^{\mathrm{T}}(\mathbf{x}_i\text{-}\mathbf{c})\le R^2,\ \ \forall\ \mathbf{x}_i\in S\ .\end{aligned} \tag{3}$$

Many machine learning methods are related to MEB. In order to investigate the relationship between the MEB problem and kernel methods, Tsang *et al* [11] extended the MEB to the Center-Constrained minimal enclosing ball (CCMEB). The CCMEB problem can be expressed as

$$\begin{aligned}&\underset{\mathbf{c},R}{\operatorname{argmin}}\ \ R^2\\ &s.t.\ \ (\varphi(\mathbf{x}_i)\text{-}\mathbf{c})^{\mathrm{T}}(\varphi(\mathbf{x}_i)\text{-}\mathbf{c})+\delta_i^2\le R^2,\ \ \forall\ i\end{aligned} \tag{4}$$

The dual of (6) is a QP problem as follows.

$$\begin{aligned}&\underset{\boldsymbol{\alpha}}{\operatorname{argmax}}\ \ \boldsymbol{\alpha}^T(diag(\mathbf{K})+\boldsymbol{\Delta})-\boldsymbol{\alpha}^T\mathbf{K}\boldsymbol{\alpha}\\ &s.t.\ \ \boldsymbol{\alpha}^T\mathbf{1}=1,\ \ \alpha_i\ge 0,\ \ \forall\ i\end{aligned} \tag{5}$$

with

$$\boldsymbol{\Delta}=\left[\delta_1^2,\cdots,\delta_N^2\right]^T\ge\mathbf{0} \tag{6}$$

Since arbitrary multiple of $\boldsymbol{\alpha}^T \mathbf{1}$ added to the objective function will not affect the optimal solution, for an arbitrary $\eta \in R$, (5) yields the same optimal $\boldsymbol{\alpha}$ as

$$\begin{aligned} &\underset{\boldsymbol{\alpha}}{\operatorname{argmax}}\ \boldsymbol{\alpha}^T \left(diag\left(\mathbf{K}\right)+\boldsymbol{\Delta}-\eta\mathbf{1}\right)-\boldsymbol{\alpha}^T\mathbf{K}\boldsymbol{\alpha} \\ &s.t.\quad \boldsymbol{\alpha}^T\mathbf{1}=1,\ \ \alpha_i \ge 0,\ \ \forall\ i \end{aligned} \tag{7}$$

For the MEB problem, an important advance is that it can be solved by using the fast core-set-based approximation algorithm [12-14], aiming at returning a good approximate solution for the MEB problem. Given a $\varepsilon > 0$, a ball $B\left(\mathbf{c},\left(1+\varepsilon\right)R\right)$ is called an $\left(1+\varepsilon\right)$–*approximation* of MEB(S) if $R \le R_{\mathrm{MEB}(S)}$ and $S \subset B\left(\mathbf{c},\left(1+\varepsilon\right)R\right)$. It has been found that solving the MEB on a subset, called the *core-set*, Q from S, can often return an accurate and efficient approximation solution [14]. A breakthrough on achieving such an $\left(1+\varepsilon\right)$–approximation was obtained by using the following iterative scheme [12]: At the t th iteration, the current estimate $B\left(\mathbf{c}_t, R_t\right)$ is expanded by including the farthest point outside the $\left(1+\varepsilon\right)$–ball $B\left(\mathbf{c}_t, R_t\right)$ and this operation is repeated until all the points in S are covered by $B\left(\mathbf{c}_t,\left(1+\varepsilon\right)R_t\right)$. Although this strategy is simple, a surprising property is revealed that the maximal number of iterations and the size of the final core-set depend only on ε, which are not related to the dimensional number or the size of a dataset [14].

(b) *MEB approximation based FRSDE*

According to the core-set based MEB fast approximation technique, the FRSDE is proposed based on the relationship between RSDE and CCMEB problems [10]. Let us observe the relationship between RSDE formulated in (1) and CCMEB problem formulated in (7). For (1), let

$$\boldsymbol{\Delta} = -diag\left(\mathbf{C}\right)+2\mathbf{p}+\eta\mathbf{1} \tag{8}$$

with $\eta \ge 0$ as a much larger constant such that $\boldsymbol{\Delta} \ge \mathbf{0}$. Then (1) is equivalent to the following formulation.

$$\begin{aligned} &\arg\max_{\boldsymbol{\gamma}}\ \ \boldsymbol{\gamma}^T(diag(\mathbf{C})+\boldsymbol{\Delta}-\eta\mathbf{1})-\boldsymbol{\gamma}^T\mathbf{C}\boldsymbol{\gamma} \\ &s.t.\quad \boldsymbol{\gamma}^T\mathbf{1}=1,\ \ \alpha_i \ge 0,\ \ \forall\ i \end{aligned} \tag{9}$$

By comparing (9) with (7), it is found that they have the same expressions when $\mathbf{K}=\mathbf{C}$. Thus, a significant conclusion on RSDE is that it can be taken as a special MEB problem, i.e. the CCMEB problem. The important finding implies that the core-set based fast MEB approximation technique is highly useful to develop efficient algorithms for RSDE including the corresponding FRSDE. The algorithm FRSDE in [10] can be described in brief below.

Algorithm FRSDE: Fast reduced sets density estimator	
Inputs:	Data set S, the approximation parameter ε, a much larger positive constant η and the kernel width h.
Outputs:	Core set Q, reduced set Sr and the weight coefficients vector $\boldsymbol{\gamma}$ of density function $\widehat{p}\left(\mathbf{x}\right)$.
Training procedure:	
Step 1:	Initialize Q_0, $\mathbf{c}_0$, and R_0. Set the iteration number $t=1$.
Step 2:	If there is no training point $\mathbf{x}$ in the extended feature space such that $\mathbf{x}$ falls outside the $\left(1+\varepsilon\right)$-ball $B\left(\mathbf{c}_t,\left(1+\varepsilon\right)R_t\right)$, go to Step 6.
Step 3:	Find $\mathbf{x}$ such that it is farthest away from $\mathbf{c}_t$ in the extended feature space. Set $Q_{t+1}=Q_t \cup \left\{\mathbf{x}\right\}$.
Step 4:	Find the new CCMEB, i.e., MEB (Q_{t+1}) in the extended feature space and then set $\mathbf{c}_{t+1} = \mathbf{c}_{\mathrm{MEC}\left(Q_{t+1}\right)}$ and $R_{t+1} = R_{\mathrm{MEC}\left(Q_{t+1}\right)}$.
Step 5:	Let $t=t+1$ and go to Step 2
Step 6:	Terminate the training procedure and return the obtained outputs.

3. Noise-Benefit FRSDE for the Speedup

In this study, we will reveal that the random Gaussian white noise manually added to the data can result in a further speedup for FRSDE. The details are presented below.

Let $\tilde{\mathbf{x}}=\mathbf{x}+n(\mathbf{x})$ be the corresponding noisy data point to the original data point $\mathbf{x}$. When the common Gaussian kernel function is adopted for FRSDE, its objective function based on the noisy dataset can be expressed as follows.

$$\begin{aligned} &\underset{\boldsymbol{\gamma}}{\operatorname{argmax}}\ \ 2\boldsymbol{\gamma}^T\mathbf{p}-\boldsymbol{\gamma}^T\mathbf{C}\boldsymbol{\gamma} \\ &=\ \underset{\boldsymbol{\gamma}}{\operatorname{argmax}}\ \ 2\sum_{i=1}^{N}\left(\frac{1}{N}\sum_{j=1}^{N}\frac{1}{\sqrt{2\pi}h^d}\exp\left(\frac{\|\tilde{\mathbf{x}}_i-\tilde{\mathbf{x}}_j\|^2}{2h^2}\right)\right)\cdot\gamma_i \\ &\qquad -\sum_{i=1}^{N}\frac{1}{\sqrt{2\pi}(\sqrt{2}h)^d}\exp\left(\frac{\|\tilde{\mathbf{x}}_i-\tilde{\mathbf{x}}_j\|^2}{4h^2}\right)\gamma_i\gamma_j \end{aligned} \tag{10}$$

Let the Gaussian white noise as the random variable with mean and standard deviation as 0 and σ, respectively. Then the expectation $\mathrm{E}\left(\|\tilde{\mathbf{x}}_i-\tilde{\mathbf{x}}_j\|^2\right)$ can be computed as follows.

If $i \neq j$

$$
\begin{aligned}
&\mathrm{E}\left(\|\tilde{\mathbf{x}}_i-\tilde{\mathbf{x}}_j\|^2\right)\\
&=\mathrm{E}\left(\|\mathbf{x}_i+n(\mathbf{x}_i)-\mathbf{x}_j-n(\mathbf{x}_j)\|^2\right)\\
&=\mathrm{E}\left(\|\mathbf{x}_i-\mathbf{x}_j\|^2+2(\mathbf{x}_i-\mathbf{x}_j)^T\left(n(\mathbf{x}_i)-n(\mathbf{x}_j)\right)+\|n(\mathbf{x}_i)-n(\mathbf{x}_j)\|^2\right)\\
&=\mathrm{E}\left(\|\mathbf{x}_i-\mathbf{x}_j\|^2\right)+2\mathrm{E}\left((\mathbf{x}_i-\mathbf{x}_j)^T\left(n(\mathbf{x}_i)-n(\mathbf{x}_j)\right)\right)+\mathrm{E}\left(\|n(\mathbf{x}_i)-n(\mathbf{x}_j)\|^2\right)\\
&=\|\mathbf{x}_i-\mathbf{x}_j\|^2+\sum_{k=1}^{d}E(n(x_{ik})-n(x_{jk}))^2\\
&=\|\mathbf{x}_i-\mathbf{x}_j\|^2+\sum_{k=1}^{d}E\left(n(x_{ik})^2\right)+\sum_{k=1}^{d}\left(n(x_{jk})^2\right)\text{-}2\sum_{k=1}^{d}E\left(n(x_{ik})n(x_{jk})\right)\\
&=\|\mathbf{x}_i-\mathbf{x}_j\|^2+2\|\boldsymbol{\sigma}\|^2
\end{aligned}
\tag{11}
$$

where $\boldsymbol{\sigma}=[\sigma_1,\ldots,\sigma_d]$. Otherwise, $\mathrm{E}\left(\|\tilde{\mathbf{x}}_i-\tilde{\mathbf{x}}_j\|^2\right)=\mathrm{E}\left(\|\tilde{\mathbf{x}}_i-\tilde{\mathbf{x}}_i\|^2\right)=\mathbf{0}$.

Based on the noisy data with the random Gaussian noise, the expectation of the objective can be expressed as follows.

$$
\begin{aligned}
&\underset{\boldsymbol{\gamma}}{\operatorname{argmax}}\ \mathrm{E}\left(2\boldsymbol{\gamma}^T\mathbf{p}-\boldsymbol{\gamma}^T\mathbf{C}\boldsymbol{\gamma}\right)\\
&s.t.\quad \boldsymbol{\gamma}^T\mathbf{1}=1,\ \ \gamma_i\geq 0,\ \ \forall\ i
\end{aligned}
\tag{12}
$$

Since $\sum_{i=1}^{N}\frac{1}{\sqrt{2\pi}h^d}\cdot\gamma_i=\frac{1}{\sqrt{2\pi}h^d}$, which is a constant, we can have

$$
\begin{aligned}
&\underset{\boldsymbol{\gamma}}{\operatorname{argmax}}\ \mathrm{E}\left(2\boldsymbol{\gamma}^T\mathbf{p}-\boldsymbol{\gamma}^T\mathbf{C}\boldsymbol{\gamma}\right)\\
&=\underset{\boldsymbol{\gamma}}{\operatorname{argmax}}\ \mathrm{E}\begin{pmatrix}2\sum_{i=1}^{N}\left(\frac{1}{N}\sum_{j=1}^{N}\frac{1}{\sqrt{2\pi}h^d}\exp\left(-\frac{\|\tilde{\mathbf{x}}_i-\tilde{\mathbf{x}}_j\|^2}{2h^2}\right)\right)\cdot\gamma_i\\
-\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{1}{\sqrt{2\pi}(\sqrt{2}h)^d}\exp\left(-\frac{\|\tilde{\mathbf{x}}_i-\tilde{\mathbf{x}}_j\|^2}{4h^2}\right)\gamma_i\gamma_j\end{pmatrix}\\
&=\underset{\boldsymbol{\gamma}}{\operatorname{argmax}}\ \exp\left(-\frac{\|\boldsymbol{\sigma}\|^2}{2h^2}\right)\begin{bmatrix}\exp\left(-\frac{\|\boldsymbol{\sigma}\|^2}{2h^2}\right)2\sum_{i=1}^{N}\left(\frac{1}{N}\sum_{j=1}^{N}\frac{1}{\sqrt{2\pi}h^d}\exp\left(\frac{\|\|\mathbf{x}_i-\mathbf{x}_j\|^2}{2h^2}\right)\right)\cdot\gamma_i\\
-\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{1}{\sqrt{2\pi}(\sqrt{2}h)^d}\exp\left(\frac{\|\tilde{\mathbf{x}}_i-\tilde{\mathbf{x}}_j\|^2}{4h^2}\right)\gamma_i\gamma_j\end{bmatrix}\\
&\quad+\sum_{i=1}^{N}\frac{1}{\sqrt{2\pi}(\sqrt{2}h)^d}\exp\left(-\frac{2\|\boldsymbol{\sigma}\|^2}{4h^2}\right)\gamma_i^2\\
&=\underset{\boldsymbol{\gamma}}{\operatorname{argmax}}\left[\exp\left(-\frac{\|\boldsymbol{\sigma}\|^2}{2h^2}\right)2\boldsymbol{\gamma}^T\mathbf{p}-\boldsymbol{\gamma}^T\mathbf{C}\boldsymbol{\gamma}\right]+\sum_{i=1}^{N}\frac{1}{\sqrt{2\pi}(\sqrt{2}h)^d}\gamma_i^2
\end{aligned}
\tag{13}
$$

Eq.(13) can be equivalently expressed as the following compact form.

$$
\begin{aligned}
&\underset{\boldsymbol{\gamma}}{\operatorname{argmax}}\ 2\boldsymbol{\gamma}^T\cdot\exp\left(-\frac{\|\boldsymbol{\sigma}\|^2}{2h^2}\right)\mathbf{p}-\boldsymbol{\gamma}^T\mathbf{C}\boldsymbol{\gamma}+R(\boldsymbol{\gamma})\\
&s.t.\quad \boldsymbol{\gamma}^T\mathbf{1}=1,\ \ \gamma_i\geq 0,\ \ \forall\ i
\end{aligned}
\tag{14}
$$

where $R(\boldsymbol{\gamma})=\sum_{i=1}^{N}\frac{1}{\sqrt{2\pi}(\sqrt{2}h)^d}\gamma_i^2$ is a regularization term. The regularization term usually can improve the generalization ability of the model. So let us observe other terms in (14) in a qualitative way.

Let

$$
\tilde{\boldsymbol{\Delta}}=-diag\left(\mathbf{C}\right)+\exp\left(-\frac{\|\boldsymbol{\sigma}\|^2}{2h^2}\right)2\mathbf{p}+\eta\mathbf{1}
\tag{15}
$$

where $\eta\geq 0$ is a much larger constant such that $\boldsymbol{\Delta}\geq\mathbf{0}$. Then (14) can be written as

$$
\begin{aligned}
&\underset{\boldsymbol{\alpha}}{\operatorname{argmax}}\ \boldsymbol{\gamma}^T\left(diag\left(\mathbf{C}\right)+\tilde{\boldsymbol{\Delta}}-\eta\mathbf{1}\right)-\boldsymbol{\gamma}^T\mathbf{C}\boldsymbol{\gamma}+R(\boldsymbol{\gamma})\\
&s.t.\quad \boldsymbol{\gamma}^T\mathbf{1}=1,\ \ \alpha_i\geq 0,\ \ \forall\ i
\end{aligned}
\tag{16}
$$

Now, we have a comparison on the MEB obtained by using the core-set based fast approximation technique based on the original dataset and the noisy dataset. In Fig. 1, the MEB and CCMEB obtained on the original dataset and the CCMEB obtained on the noisy dataset (called CCMEB_noise for simplicity) by FRSDE are shown. The extended part of the CCMEB and CCMEB_noise are as follows.

CCMEB:

$$
\boldsymbol{\Delta}=-diag\left(\mathbf{C}\right)+2\mathbf{p}+\eta\mathbf{1}=[\delta_1^2,\cdots,\delta_N^2]
\tag{17.a}
$$

CCMEB_noise:

$$
\begin{aligned}
\tilde{\boldsymbol{\Delta}}&=-diag\left(\mathbf{C}\right)+\exp\left(-\frac{\|\boldsymbol{\sigma}\|^2}{2h^2}\right)2\mathbf{p}+\eta\mathbf{1}\\
&=[\delta_1'^2,\cdots,\delta_N'^2]
\end{aligned}
\tag{17.b}
$$

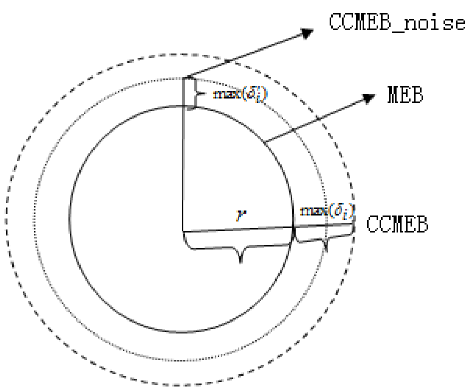


Fig. 1 Comparison of the MEB obtained on the original dataset and its noisy version

Comparing (17.a) and (17.b), we can find $\delta_1'^2<\delta_N'^2$. It indicates that FRSDE on the noisy dataset is able to get a more compact ball than that obtained on the original dataset with the same values of the approximation parameter ε. Thus if a larger value is set for the approximation parameter ε on the noisy dataset, the more slack ball can be learned which is equivalent to the ball obtained by FRSDE on the original dataset. We reveal that the random Gaussian white noise added to the data will result in a fast learning since a much large value of ε can be adopted to obtain the equivalent approximation performance, which only can be obtained based on the original data with a smaller value of approximation parameter. More precisely, the smaller value of the approximation parameter ε will result in the longer learning time of FRSDE, so that a bigger value of this parameter can lead to a speedup of the solution to the MEB associated with the kernel density estimation. Thus, the noisy data based FRSDE will be faster than the original data based FRSDE in order to obtain the equivalent density estimation accuracy. Here, the noisy-benefit FRSDE is called NB-FRSDE in this study. In general, the proposed NB-FRSDE has the following distinctive characteristics.

(1) It is a simple implementation of FRSDE on the dataset with the Gaussian white noise added.

(2) The NB-FRSDE can benefit from the noise in the sense of the average running time, which is very different from the most of the existing machine learning methods that always try to remove the noise from the data in order to overcome the influence of noise.

4. Experimental studies

In this section, in order to investigate the performance of NB-FRSDE, the experimental results on density estimation using two density estimators, i.e., FRSDE and NB-FRSDE, are compared and reported. In our experiments, two kinds of experiments are conducted. Firstly, the accuracy of density estimation is evaluated on different size of datasets. Secondly, the adopted algorithms are applied to image segmentation.

We use the following index, i.e., the accuracy of density estimation, denoted as *J*, to measure the density estimation accuracy, which is computed by the L2 error between the obtained density function and the real density function. Meanwhile, the running time, denoted as T, is also used to further evaluate the performance of the adopted algorithms.

In our experiments, the best Gaussian kernel width parameter is determined by 10 folds cross-validation strategy based on the log-likelihood criterion as in [9].

4.1. Performance on Density Estimation

The 1-D multi-modal benchmarking Gaussian density function in (18) is used in this experiment [9, 10]. Training sets of 1e3 to 1e6 data points are drawn from this distribution and two density estimators FRSDE and NB-FRSDE are fit to the data. A testing set of 1000 points is then drawn and the L_2 testing errors of two estimators are computed. For the noisy data, 10 different datasets with the same extent of noise are generated based on the original noise-free dataset. In order to observe and compare the performances of FRSDE and NB-FRSDE, we take the approximation parameter ε as 2e-6 for FRSDE, and 3e-6 and 4e-6 for NB-FRSDE respectively. In other words, in order to verify the idea in the last section, we want to observe whether NB-FRSDE with a comparatively bigger ε on the noisy version of the original dataset is comparable in the sense of the average approximation accuracy of density estimation but faster in the sense of the average running time, when comparing with FRSDE with a comparatively smaller ε on the original dataset.

$$p(x) = \frac{1}{8}\sum\nolimits_{i=0}^{7} G_{h_i}(u_i, x), \tag{18}$$

where $h_i = \left(\frac{2}{3}\right)^i$ and $u_i = 3(h_i - 1)$

In Tab.1 and 2, the comparison of the accuracy and training time are reported. From these results we can have the following observations and analysis.

(1) In Table 1, the accuracy obtained by FRSDE on the original dataset and the means and standard deviations (s.d for simplicity) of accuracy obtained by NB-FRSDE on the noisy datasets are compared. The results show that the obtained accuracies of both methods are comparative.

(2) In Table 2, the running time of FRSDE on the original dataset and the means and standard deviations of accuracies of NB-FRSDE on the noisy datasets is compared. The results on the average running time show that the NB-FRSDE is more efficient than FRSDE.

In general, while FRSDE and NB-FRSDE have the comparative performances on the accuracy of density estimation, the latter has an obvious advantage on the running time.

Table 1

Performance comparison of between Accuracy (J) between FRSDE and NB-FRSDE

Size	Accuracy (J)						
	FRSDE (ε =2e-6)	NB-FRSDE (ε =3e-6)					
		(σ =0.05)		(σ =0.1)		(σ =0.2)	
		mean	s.d	mean	s.d	mean	s.d
1e3	0.0028	0.0022	0.0006	0.0014	0.0006	0.0013	0.0004
5e3	0.0029	0.0019	0.0011	0.0019	0.0011	0.0015	0.0004
1e4	0.0025	0.0015	0.0010	0.0021	0.0010	0.0014	0.0002
5e4	0.0016	0.0017	0.0014	0.0019	0.0014	0.0013	0.0005
1e5	0.0017	0.0012	0.0009	0.0018	0.0009	0.0018	0.0007
5e5	0.0013	0.0017	0.0003	0.0015	0.0003	0.0013	0.0005
1e6	0.0015	0.0014	0.0010	0.0017	0.0010	0.0011	0.0002
Size	Accuracy (J)						
	FRSDE (ε =2e-6)	NB-FRSDE (ε =4e-6)					
		(σ =0.05)		(σ =0.1)		(σ =0.2)	
		mean	s.d	mean	s.d	mean	s.d
1e3	0.0028	0.0022	0.0012	0.0020	0.0014	0.0026	0.0012
5e3	0.0029	0.0019	0.0011	0.0021	0.0008	0.0019	0.0007
1e4	0.0025	0.0015	0.0008	0.0015	0.0009	0.0015	0.0007
5e4	0.0016	0.0017	0.0002	0.0017	0.0004	0.0017	0.0007
1e5	0.0017	0.0012	0.0007	0.0014	0.0005	0.0012	0.0006
5e5	0.0013	0.0017	0.0003	0.0015	0.0007	0.0017	0.0007
1e6	0.0015	0.0013	0.0007	0.0014	0.0004	0.0014	0.0004

Table 2

Performance comparison of running time between FRSDE and NB-FRSDE

Size	Running time (T)						
	FRSDE (ε =2e-6)	NB-FRSDE (ε =3e-6)					
		(σ =0.05)		(σ =0.1)		(σ =0.2)	
		mean	s.d	mean	s.d	mean	s.d
1e3	180.3684	0.9766	0.9739	0.3705	0.0390	0.4134	0.0772
5e3	255.4984	0.8775	0.2751	1.2074	0.5938	0.9984	0.0220
1e4	360.0659	1.3416	0.5872	2.0436	0.4899	1.2870	0.6508
5e4	486.5983	1.7706	0.7086	2.2744	0.4504	1.2792	0.5956
1e5	1506.657	1.6887	0.0234	1.4391	0.4103	1.6302	0.0772
5e5	2588.274	1.9851	0.4140	1.8135	0.0702	2.0124	0.0882
1e6	2421.759	2.3244	0.8255	3.0732	0.7317	2.2230	0.0772
Size	Running time (T)						
	FRSDE (ε =2e-6)	NB-FRSDE (ε =4e-6)					
		(σ =0.05)		(σ =0.1)		(σ =0.2)	
		mean	s.d	mean	s.d	mean	s.d
1e3	180.3684	0.3432	0.1141	0.2925	0.0603	0.3744	0.1416
5e3	255.4984	0.7831	0.2232	0.8229	0.3515	0.8642	0.1367
1e4	360.0659	1.6505	0.1100	1.6068	0.0422	1.3026	0.5405
5e4	486.5983	1.7940	0.3315	1.4157	0.3978	1.7253	0.5436
1e5	1506.657	1.3603	0.4775	1.8915	0.4343	1.8595	0.6592
5e5	2588.274	3.1044	0.9153	1.8798	0.4949	2.0280	0.3885
1e6	2421.759	2.7424	0.5569	1.7355	0.5149	2.6707	0.5089

4.2. Application to large image segmentation

In this subsection, we apply FRSDE and NB-FRSDE to image segmentation. The proposed NB-FRSDE and FRSDE based image segmentation method can be stated as follows [10]:

1) Generate the data set S_{im} by extracting the features of each pixel in an image;

2) Estimate the density function of the data set S_{im} or noisy S_{im}, and then obtain the reduced set S_{re} of S_{im} by using FRSDE or NB-FRSDE;

3) Use the famous fuzzy clustering algorithm FCM in [22] to cluster the reduced set S_{re} into C clusters and label the data points in S_{re};

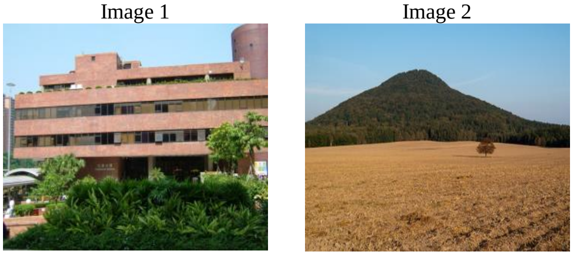
4) Take S_{im} as the testing set and the labeled S_{re} as the training set and then utilize KNN classifier [23] to classify S_{im} into C clusters;

5) Obtain the segmented image using the obtained classification result in step 4).

In our experiment, HSV features of a color image are extracted to generate the corresponding data set. In the above step 4), we set $k=1$ for KNN classifier. Two color images, as shown in Fig. 2, are adopted for this experiment. The sizes of Fig.2 (a) and (b) are 426*527*3 and 223*350*3, respectively. For each image, it is partitioned into three sections.

Fig. 3 and Table 3 show the segmented results. From Fig. 3, we can see that these two methods can obtain comparable segmentation results on the adopted images. However, from Table 3, it can be easily found that for these images, the NB-FRSDE based method has an obvious advantage over the FRSDE based method in running time.

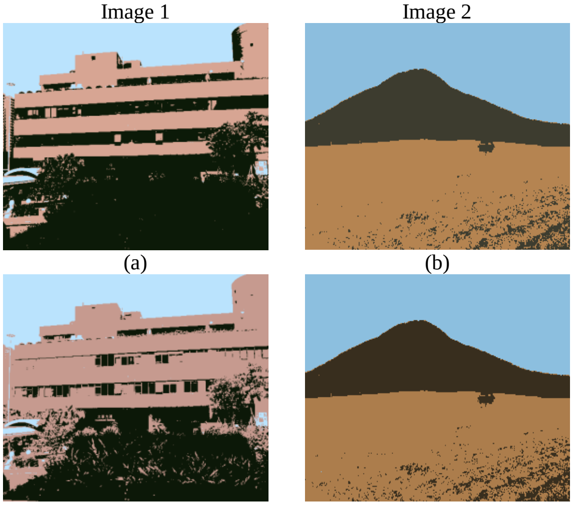
Image 1 Image 2



(a) (b)

Fig. 2 The adopted images for segmentation.

Image 1 Image 2



(a) (b) (c) (d)

Fig. 3 The segmentation results by two methods: (a) and (b) obtained by FRSDE based method; (c) and (d) obtained by NB-FRSDE based method.

Table 3

Performance comparison between FRSDE and NB-FRSDE on running time.

	Running time (T)		
	FRSDE	NB-FRSDE (σ =0.1)	
		mean	s.d
Image 1	308.1955	154.1393	7.4803
Image 2	301.5655	160.0414	1.6748

5. Conclusions

In this study, it is revealed that the random Gaussian white noise manually added to the original data will result in a speedup of FRSDE density estimator where a much larger value of ε can be used to obtain the equivalent approximation performance that only can be obtained by FRSDE based on the original data with a smaller value of ε. The experimental studies on density estimation and the application on to image segmentation also demonstrate this advantage.

Although the proposed NB-FRSDE has demonstrated the promising performance, many further works can be done in depth. For example, the following several aspects deserve the further study.

(1) When a larger value of ε is adopted for NB-FRSDE, how to determine the most appropriate value is still an open problem.

(2) Large scale datasets are becoming main data sources for many modeling tasks [20, 21]. How to use the proposed method for different application scenarios with large scale data is still a challenge work.

(3) For complicated modeling scenarios, transfer learning and multi-view learning are attracting more and more attentions. Thus, it is also a very valuable work to integrate the proposed method with transfer learning and multi-view learning techniques and develop the corresponding new methods.

Acknowledgements

This work was supported in part by the General Research Fund of the Hong Kong Research Grants Council (PolyU5134/12E), the National Natural Science Foundation of China (61170122, 61272210), the Ministry of Education Program for New Century Excellent Talents (NCET-120882), the Fundamental Research Funds for the Central Universities (JUSRP51321B), and the Outstanding Youth Fund of Jiangsu Province (BK20140001).

References

[1] B.W. Silverman, *Density Estimation for Statistics and Data Analysis Chapman and Hall*, CRC press, 1986.

[2] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford Univ. Press, 1995.

[3] E. Parzen, On Estimation of a Probability Density Function and Mode, Annals of Math. Statistics, **33** (1962), 1065-1076.

[4] S. Mukherjee and V. Vapnik, *Support Vector Method for Multivariate Density Estimation*, Center for Biological and Computational Learning. Department of Brain and Cognitive Sciences, MIT, 1999, 170.

[5] D.W. Scott and S.J. Sheather, Kernel Density Estimation with Binned Data, *Comm. Statistics-Theory and Methods*, **14** (1985), 1353-1359.

[6] V. Vapnik and S. Mukherjee, *Support Vector Method for Multivariate Density Estimation*, Advances in Neural Information Processing Systems, MIT Press, 2000, 659-665.

[7] L. Holnström , The Error and the Computational Complexity of a Multivariate Binned Kernel Density Estimator, *J. Multivariate Analysis*, **72** (2) (2000), 264-309.

[8] B. Jeon and D.A. Landgrebe, Fast Parzen Density Estimation Using Clustering-Based Branch and Bound, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **16** (9) (1994), 950-954.

[9] M. Girolami, C. He, Probability density estimation from optimally condensed data samples, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **25** (10) (2003), 1253 – 1264.

[10] Z.H Deng, F.L. Chung, S.T. Wang, FRSDE: Fast reduced set density estimator using minimal enclosing ball approximation, *Pattern Recognition*, **41** (4) (2008), 1363-1372.

[11] I.W. Tsang, J.T. Kwok, J.M. Zurada, Generalized core vector machines, *IEEE Transactions on Neural Networks*, **17** (5) (2006), 1126-1140.

[12] M. Bādoiu and K. L. Clarkson, *Optimal core sets for balls*, In DIMACS Workshop on Computational Geometry, 2002.

[13] M. Bādoiu , S. Har-Peled, and P. Indyk, *Approximate clustering via core sets*, In Proceedings of 34th Annual ACM Symposium on Theory of Computing, 2002, 250–257.

[14] M. Xu, H. Ishibuchi, X., Gu, and S. Wang, Dm-KDE: dynamical kernel density estimation by sequences of KDE estimators with fixed number of components over data streams, **Frontiers of Computer Science**, **8** (4) (2014), 563-580.

[15] R. Abdenebi, S. Chitroub, and A. Bouridane, Density estimation of high dimensional data using ICA and Bayesian networks, *Intelligent Data Analysis*, **18**(2) (2014) 157-179.

[16] N. Kanta, and S. Eguchi, Density estimation with minimization of U-divergence, *Machine learning*, **90**(1) (2013), 29-57.

[17] S.T. Wang, J. Wang, and F.L. Chung, Kernel density estimation, kernel methods, and fast learning in large data sets, *IEEE Transactions on Cybernetics*, **44**(1) (2014), 1-20.

[18] K. Matej, and A. Leonardis, Online Discriminative Kernel Density Estimator With Gaussian Kernels, *IEEE Transactions on Cybernetics*, **44**(3) (2014), 355-365.

[19] X. Honga, S. Chenb, A. Qatawnehc, K. Daqrouqc, M. Sheikhc and A. Morfeqc, Sparse probability density function estimation using the minimum integrated square error, *Neurocomputing*, **115** (2013),122-129.

[20] Z. Deng, K.S. Choi, F.L. Chung, and S. Wang, S, Scalable TSK fuzzy modeling for very large datasets using minimal-enclosing-ball approximation, *IEEE Transactions on Fuzzy Systems*,**19** (2) (2011), 210-226.

[21] F.L. Chung, Z.H. Deng, and S. Wang, From minimum enclosing ball to fast fuzzy inference system training on large datasets, *IEEE Transactions on Fuzzy Systems*, **17** (1) (2009), 173-184.

[22] P.Yan, C.Zhang, *Pattern Recognition*, Beijing: Tsinghua University Press, 2000.

[23] A.W.C. Liew, H. Yan and N.F. Law, Image Segmentation Based on Adaptive Cluster Prototype Estimation, *IEEE Transactions on Fuzzy Systems*, **13** (4) (2005), 444-453.