African Research Review

An International Multi-Disciplinary Journal, Ethiopia
Vol. 5 (1), Serial No. 18, January, 2011
ISSN 1994-9057 (Print) ISSN 2070-0083 (Online)

Adaptive Kernel in Meshsize Boosting Algorithm in KDE

(Pp. 438-446)

Ishiekwene, C. C. **-** Department of Mathematics**,** University of Benin, Nigeria
E-mail: cycigar@yahoo.co.uk

Nwelih, E. **-** Department of Computer Science**,** University of Benin, Nigeria

Abstract
This paper proposes the use of adaptive kernel in a meshsize boosting algorithm in kernel density estimation. The algorithm is a bias reduction scheme like other existing schemes but uses adaptive kernel instead of the regular fixed kernels. An empirical study for this scheme is conducted and the findings are comparatively attractive.

Key Words: Boosting, kernel density estimates, bias reduction, adaptive kernel, meshsize, noises

Introduction

Boosting in kernel density estimation was first proposed by Schapire (1990). Other authors like Freund (1995), Schapire and Singer (1999) but to mention a few have also made contributions. Boosting is a means of improving the performance of a 'weak learner'. It is applied in this context using the adaptive kernel. Boosting does not only guarantee an error rate that is better than random guessing but also deals with the correction of 'noises' at the tails of the distribution or where we have sparse cluster of data within a given region.

In 2004, Mazio and Taylor proposed an algorithm in which a kernel density classifier is boosted by suitably re-weighting the data. This weight placed on the kernel estimator, is a ratio of a log function in which the denominator is a leave-one-out estimate of the density function. A theoretical explanation is also given to show how boosting is a bias reduction technique i.e a reduction of the bias term in the expression for the asymptotic mean integrated squared error (AMISE).

Methods

Algorithm on Boosting Kernel Density Estimates and Bias Reduction

Throughout this paper, we shall assume our data to be univariate. The algorithm of Mazio and Taylor (2004) is briefly summarized in algorithm 1.

Algorithm 1

Step 1: Given $\{x_i, i = 1,2,\ldots,n\}$, initialize $W_1(i) = 1/n$

Step 2: Select h (the smoothing parameter).

Step 3: For m =1, 2, … M, obtain a weighted kernel estimate

$$\hat{f}_m(x) = \sum_{i=1}^{n} \frac{W_m(i)}{h} k\left(\frac{x - x_i}{h}\right) \qquad (2.1)$$

where x can be any value within the range of the x_i 's, k is the kernel function and w is a weight function and then update the weights according to

$$W_{m+1}(i) = W_m(i) + \log\left\{\frac{\hat{f}_m(x_i)}{\hat{f}_m^{(-1)}(x_i)}\right\} \qquad (2.2)$$

Step 4: Provide output as

$$\prod_{m=1}^{M} \hat{f}_m(x) \text{ renormalized to integrate to unity}$$

For the full implementation of this algorithm see Marzio and Taylor (2004).

Boosting as a Bias Reduction in Kernel Density Estimation

Suppose we want to estimate f(x) by a multiplicative estimate. We also suppose that we use only "weak" estimates which are such that h does not tend to zero as n→ ∞. Let us use a population version instead of sample in which our weak learner, for h >0 is given by

$$\hat{f}_m(x) = \int \frac{1}{h} W_m(y) K\left(\frac{x-y}{h}\right) f(y)dy \qquad (2.3)$$

Where $W_1(y)$ is taken to be 1. We shall take our kernel function to be Gaussian (since all distributions tend to be normal as n, the sample size, becomes large through central limit theory (Towers, 2002). The first approximation in the Taylor's series, valid for h < 1 provided that the derivatives of f(x) are properly behaved, is

$\hat{f}_{(1)}(x) = f(x) + \frac{h^2 f''(x)}{2}$ and so we observe the usual bias of order $0(h^2)$ of Wand and Jones (1995). If we now let $W_2(x) = \hat{f}_{(1)}(x)^{-1}$, the boosted estimator at the second step is

$$\hat{f}_2(x) = \int k(z)\left\{f(x+zh) + h^2 \frac{f''(x+zh)}{2} + 0(h^4)\right\}^{-1} f(x+zh)dz$$

$$= 1 - \frac{h^2 f''(x)}{2f(x)} + 0(h^4) \qquad (2.4)$$

This gives an overall estimator at the second step as

$$\hat{f}_{(1)}(x) \cdot \hat{f}_{(2)}(x) = f(x)\left\{1 + h^2 \frac{f''(x)}{2f(x)} + 0(h^4)\right\}\left\{1 - \frac{h^2 f''(x)}{2f(x)} + 0(h^4)\right\}$$

$$f(x) + 0(h^4) \qquad (2.5)$$

This is clearly of order four and so we can see a bias reduction from order two to order four.

Meshsize Algorithm in Boosting

We shall see how the leave-one-out estimator of the (2.2) in the weight function can be replaced by a meshsize estimator due to the time complexity involved in the leave-one-out estimator. In the leave-one-out estimator, we require (n+(n-1)).n function evaluations of the density for each boosting step. Thus, we are using a meshsize in its place. The only limitation on this meshsize algorithm is that we must first determine the quantity $1/nh$ so as to know what the meshsize that would be placed on the weight function of (2.2) would be (Ishiekwene et.al, 2008). The need to use a meshsize in place of the leave-one-out lies on the fact that boosting is like the steepest-descent algorithm in unconstrained optimization and thus a good substitute that approximates the leave-one-out estimate of the function (Duffy and Helmbold, 2000; Taha, 1971; Ratsch *et al.*, 2000; Mannor *et al.*, 2001: Hazelton & Turlach,2007).

The new meshsize algorithm is stated as:

Algorithm 2

Step 1: Given $\{x_i, i = 1,2,\ldots,n\}$, initialize $W_1(i) = 1/n$

Step 2: Select *h* (the smoothing parameter).

Step 3: For m =1, 2 … M,

(i) Get

$$\hat{f}_m(x) = \sum_{i=1}^{n} \frac{W_m(i)}{h} k_A\left(\frac{x - x_i}{h}\right)$$

where x can be any value within the range of the x_i 's, k_A is the adaptive kernel function and w is a weight function

(ii) Update

$$W_{m+1}(i) = W_m(i) + mesh$$

Step 4: Provide output

$\prod_{m=1}^{M} \hat{f}_m(x)$ and normalize to integrate to unity

We can see that the weight function uses a meshsize instead of the leave-one-out log ratio function of Mazio and Taylor (2004). The kernel function used is the adaptive kernel unlike the fixed used in Ishiekwene et.al 2008. The numerical verification of this algorithm would be seen in the next section (numerical results and discussion).

Results & Discussion

In this section, we shall use three sets of data to illustrate our algorithm and BASIC programming language is used. Data 1 is a sample of size forty and is the lifespan of car batteries in years. Data 2 is a sample of size sixty-four and is the number of written words without mistakes in every 100 words by a set of students in a written essay. Data 3 is the scar length of patients randomly selected in millimeters (Ishiekwene and Afere, 2001; Ishiekwene and Osemwenkhae, 2006).

The results are shown in figures 1 – 3. Figure 1 is the graph for Data 1, Figure 2 for Data 2 while Figure 3 is for Data 3. In all three charts shown in figures 1 – 3, the three kernel methods are plotted on the same sheet for easy comparison at a glance (ie the classical fixed kernel method ,the adaptive kernel method and the boosted kernel method). The boosted version is obtained using the Meshsize Boosting algorithm of Ishiekwene et.al (2008). That is Algorithm 2.

The results as shown in figures 1 - 3 reveal that the classical fixed kernel density estimation method oversmooths the curves by obscuring some important features in the data. The adaptive kernel method showed a clearer picture of the nature of the data around the tails. The boosted kernel method was close to the adaptive kernel method in all three data used thus showing that this method is clearer than the classical fixed kernel method in terms of revealing data features. It does not only reveal features at the tails but is a bias reduction scheme as shown theoretically above and in Table 1(Birke,2009 and See Appendix).

Conclusion

We have shown that the adaptive kernel can be used in place of the classical fixed kernel in boosting in kernel density estimation. The charts- figs. 3.1 – 3.3 and table 3.1 clearly reveals that the adaptive kernel method does better than the classical fixed kernel method in kernel density estimation. It is therefore recommended for use in place of the classical fixed kernel method in boosting in KDE having exhibited the qualities of bias reduction and revealing the data features at the tails.

References

Birke, Melanie (2009). "Shape constrained KDE." *Journal of Statistical Planning & Inference, vol **139**, issue 8* , August 2009, pg 2851 – 2862.

Duffy, N. and Hemlbold, D. (2000). "Potential bosters? Advances in Neural info." *Proc. Sys. **12***, 258 – 264.

Freund, Y. (1995). "Boosting a Weak Learning Algorithm by Majority." *Info. Comp. **121**,*pg 256 - 285.

Freund, Y. and Schapire, R. (1996). "Experiments with a New Boosting Algorithm. In *Machine Learning*: proceedings of the **12th** Intl. conf. Ed. L. Saite 148 - 156.

Hazelton, M.L & Turlach, B. A. (2007). Reweighted KDE. *Computational Statistics & Data Analysis,Vol **51**,issue 6*,March 2007,pg 3057 – 3069.

Ishiekwene, C.C and Afere, B.A.E. (2001). Higher Order Window Width Selectors for Empirical Data. *Journal of Nigerian Statistical Association (J.N.S.A), **14**,* 69 – 82.

Ishiekwene, C. C. and Osemwenkhae, J. E. (2006). "A comparison of fourth order window width selectors in Kernel Density Estimation (A Univarate case)", *ABACUS, 33;* 14 - 20.

Ishiekwene, C.C; Ogbonmwan, S.M and Osemwenkhae, J.E. (2008) – A Meshsize Boosting Algorithm in Kernel Density Estimation. *Journal of Science & Technology, Ghana. Vol **28*** # 2,pg 69 – 74.

Jones, M.C. and Signorini, D.F. (1997). "A comparison of Higher-order Bias Kernel Density Estimators." *JASA 92***,**pg 1063 – 1072.

Mannor, S.; Meir, R. and Mendelson, S. (2001). "On Consistency of Boosting Algorithms; submitted to Adv. In *Neural info. Proc. Sys.*

Mazio, D. M. and Taylor, C.C. ( ) Boosting Kernel Density Estimates: A bias Reduction Technique? *Biometrika 91*, pg 226 - 233.

Ratsch, G. Mika, S., Scholkopf, B. and Muller, K. K. (2000). Construction boosting algorithms from SVM's: An application to-one-class classification. *GMD tech report no. 1.*

Rohatgi, V.J (1984). *Statistical Inference*. New York: John Wiley and sons,

Schapire, R. (1990). "The Strength of Weak Learnability." *Machine learn. 5*,pg 313 – 321.

Schapire, R. and Singer, Y. (1999). "Improved Boosting Algorithm Using Confidence Rated Prediction." *Machine learn. 37,*pg 297 – 336.

Taha, H. A. (1971). *Operations Research: An Introduction*. Prentice Hall, New Jersey.

Towers, S. (2002). *Kernel Probability Density Estimation Methods*. New York: State Univ. of New York at Brook.

Wand, M.P. and Jones, M.C. (1995). *Kernel Smoothing*. London: Chapman and Hall/CRC.

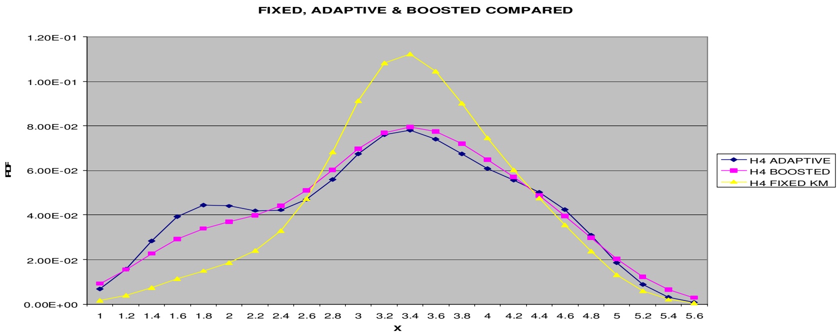

Fig 1: Chart showing the three techniques Using Data 1

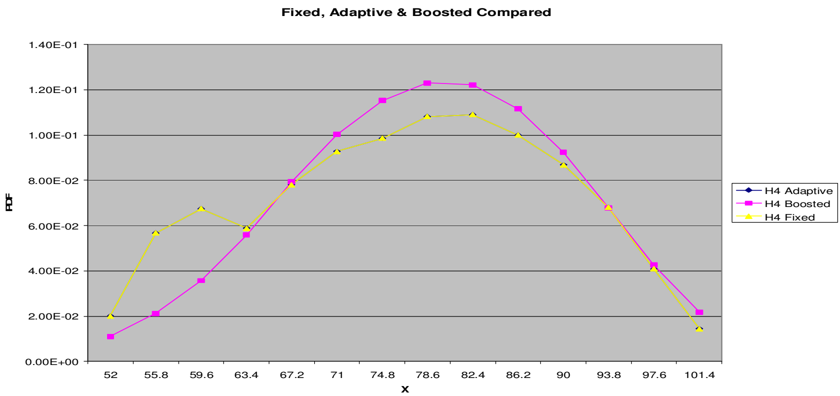

Fig 2: Chart showing the three techniques Using Data 2

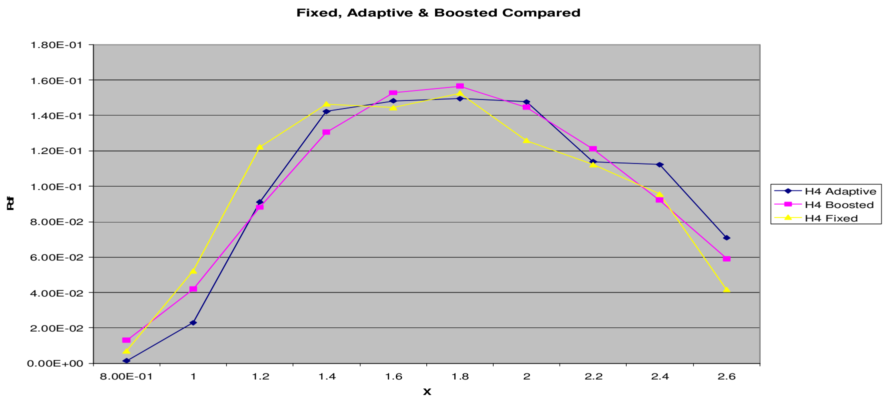

Fig 3: Chart showing the three techniques Using Data 3

Table 1 SHOWING BIAS REDUCTION

	FIXED KDE Method			VARIABLE KDE Method			BOOSTED Method		
	Bias2	Var	AMISE	Bias2	Var	AMISE	Bias2	Var	AMISE
Data 1	0.005276637	0.019811685	0.250883225	0.002071803	0.014789245	0.016861048	0.002078009	0.015168456	0.017246655
Data 2	0.000293946	0.001130402	0.001424348	0.000108591	0.000809307	0.000917898	0.000108715	0.000822153	0.000930868
Data 3	0.004697604	0.016623515	0.021321119	0.001767617	0.011342162	0.013109779	0.001768218	0.011446219	0.013214437