編號 FA-33

以數值法求取或然率密度函數的局部估算之研究

Numerical Study upon Local Estimates of Probability Density Functions

Yih Nen Jeng (鄭育能)
Department of Aeronautics and Astronautics
National Cheng Kung University
Tainan, Taiwan, R.O.C.
Email: ynjeng@mail.iaa.ncku.edu.tw
You-Chi Cheng (鄭又齊)
Department of Electrical Engineering
National Taiwan University

ABSTRACT

A local estimating algorithm of evaluating local probability density function is proposed by embedding the Gaussian kernel to the counting of occurrence frequencies. The random number generator of the Micro-Soft FORTRAN package (F-77), say the RANDOM _NUMBER(x) subroutine, is employed to produce pseudo- random number. Numerical tests show that, if the Gaussian kernel factor is large enough, local estimates converge to fixed density function distributions. Moreover, the local probability density function's variation of a random data string composed of two different distributions is captured by the present method. It is believed that, if a true random number generator is employed or a smoothing procedure upon the probability density function is employed, the convergent Gaussian kernel factor will be significantly reduced.

Keywords: local probability density function estimation, Gaussian kernel function, composite random variables.

INTRODUCTION

Because of the development of computer technologies, it was predicted that, about 60 years later, a single computer's physical and hard disk memories might be larger than the total memories of the whole living beings of the earth. In other words, the computing resource might increase to a practical level that many compli- cated problems can be analyzed in future. Simu-ltaneously, the related technologies will produce significantly large amount of data in which one part of data might require sophisticated analysis. To treat them accurately, reasonable methods should be fully automatic. Unfortunately, most of present available methods of data analysis heavily rely on artificial techniques. Therefore, it is reasonable to search fundamental techniques for automatic data analysis.

An automatic data analysis method might be divided into 4 stages: separates data into the deterministic and random parts via robust estimation [1-5]; decomposes the deterministic part into sinusoidal and non-sinusoidal part and the former is further decomposed into single waves [6-9]; examines the physical meaning of each single wave by searching its governing equation; and investigates the statistical proper- ties of the random part. The present study tries to develop fundamental techniques for automatic probability statistical analyses.

In the statistical analysis, many probability density functions had been developed and listed in mathematical handbooks [10-12]. People may calculate the estimate of probability density function of a data string and compares it to a existing one. However, if a long data string is composed of more than one independent informations, the classical probability density function analysis may be inadequate to represent the random part. In other words, the resulting function can not correctly reflect different data sources. Today, there are two available semi-artificial wave decomposition methods [6-9] for deterministic data. To the author's knowledge, however, there is no such decomposition method for the random data.

It is believed that, to search a decomposition method for the random data, it is an important issue to develop the localized probability density function analysis.

ANALYSIS

Consider a normalized distributed random data string, the corresponding probability den- sity and distribution functions are, respectively, [10-12]

$$p(z) = \frac{1}{\sqrt{2\pi}} \exp[-\frac{z^2}{2}] \tag{1}$$

$$P(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} \exp[-\frac{\xi^2}{2}] d\xi \tag{2}$$

where $z = (x - \mu_x)/\sigma$, with μ_x and σ as the mean and standard deviation of the random data. If the data is of the chi-squared distribution, the corresponding probability density and distri-
bution functions are, respectively

$$p(\chi^2) = \frac{1}{2^{n/2}\Gamma(n/2)} \left(\chi^2\right)^{(n/2)-1} \exp[-\frac{\chi^2}{2}]$$

$$\chi^2 \geq 0 \tag{3}$$

$$P(\chi^2) = \int_{-\infty}^{\chi^2} p(\chi^2) d\chi^2 \tag{4}$$

For two independent random variables x and y (where $a \leq x \leq b$ and $\alpha \leq y \leq \beta$), whose probability density functions are $p_1(x)$ and $p_2(y)$, a linear combination is

$$z = c_1 x + c_2 y \tag{5}$$

The corresponding probability density function can be evaluated as

$$p(z) = \sum_{z = c_1 x + c_2 y} p_1(x) p_2(y) \tag{6}$$

such that $c_1 a + c_2 \alpha \leq z \leq c_1 b + c_2 \beta$. Note that the expression of Eq.(6) can be expressed as

$$p(z) = p(c_1 x + c_2 y) = \int_{-\infty}^{\infty} p_1(x) p_2(\frac{z - c_1 x}{c_2}) dx \tag{7}$$

With known probability density or distribution function, the corresponding mean value, mean square value, variance and other statistical parameters can be evaluated. However, although all the existing probability density functions can be obtained from published literatures, they are developed under completely specified conditions. For a physical process whose conditions of generating the random variables are not completely known, it is very hard to determine the exact density function to be tested. Therefore, the following local test algorithm is proposed.

Random Number Generation

From the Micro-Soft IMSL Liberary in the F-77 FORTRAN power station, the pseudo- random number generator written in the form

$r(i)$ = RANDOM_NUMBER(x)

can give a pseudo random data set within the range between 0 and 1. The following **classical algorithm** gives the estimated probability density function [10- 11]:

1. Divide the range of $r(i)$, $0 \leq i \leq n$, into m equal spacing intervals $r_0 = r(i)_{\min}, r_1, r_{2,...,}$ $r_m = r(i)_{\max}$, $\Delta r = r_{j+1} - r_j =$ constant.
2. Assign each interval a counting number $a_j = 0, \quad j = 1, m$
3. Comparing $r(i)$ to all the intervals, if $r_j \leq r(i) \leq r_{j+1}$, add 1 to a_j.
4. After checking all i, normalize a_j to be $a_j/(n+1)$. These a_j's constitute the approximated probability density function of the random data..

By employing this procedure, a typical result is shown in Figs.1a and 1b. Since the random number generator gives a pseudo-random number, there is significant deviation (about 11% maximum error) of the probability density function from the exact density value of 0.5. If the interval number increases, the scattered error becomes more significant. Note that, there are many kernel density functions shown in Ref.[13] which can smooth these deviations. Because of the first study, it seems reasonably to omit the smoothing procedure and to restrict the number of intervals to be 20.

To generate random number of a known distribution, a convenient method is to set the pseudo-random number data as the value of probability density function and inversely obtain the corresponding random variable. For example, for the normal distribution, let

$$r(i) = P(z_i) \tag{8}$$

The inversion formulas for a normal distributed random variable is

$$z_i = P^{-1}(r(i)) \tag{9}$$

which can be approximately evaluated via a searching procedure upon the tabulated

table for the normal distribution together with a interpolation formulas.

Local Estimate of Probability Density Function

From a random data string, one can estimate the probability density function via the above mentioned procedure. However, if the random data string is a linear combination of two random variables each with a different distribution, the above estimation procedure certainly leads to a ultimately different estimate of the density function. In order to introduce the localization effect, the Gaussian kernel function is embedded to the procedure, so that it becomes the following **localized algorithm** for the $k-$th point.

1. Divide the range of $r(i),\ 0 \le i \le n$, into m equal spacing intervals $r_0 = r(i)_{\min}, r_1, r_2, ..., r_m = r(i)_{\max}$.
2. Assign each interval a counting number $a_{jk} = 0,\ j = 1, m$ for the fixed k. Meanwhile, assign 0 to T.
3. Comparing $r(i)$ to the intervals, if $r_j \le r(i) \le r_{j+1}$, add $\exp[-(i-k)^2 / 2\sigma^2]$ to a_{jk} and T respectively.
4. After checking all i, normalize a_{jk} to be a_{jk}/T. These a_{jk}'s are the approximated local probability density function of the $k-$ point.

The first step is not changed and the kernel factor σ roughly reflects the averaging range. Since the employed pseudo-random data's probability density function has about 10% error, the value of σ should be large enough as will be discussed below.

RESULTS AND DISCUSSIONS

Figures 2a and 2b are the normal distributed random number and the probability density function generated from the pseudo-random number. The estimate of the random number coincides to that of the normal distribution except some wiggles which is caused by the scattering of Figs.1b. The corresponding chi-squared distributed random number and probability density function are shown in Figs.3a and 3b, respectively. Again, small deviations from the exact density function are found. Nevertheless, the main characters of two distributions are captured by these two approximated random variables.

That shown in Fig.4a compares the local probability density function estimates of the normal distributed random data at the central point (total points = 2000) with different Gaussian kernel factor σ (100, 300, 500, and 1000). The long dashed line is the exact probability density function. It is seen that, as $\sigma \ge 500$, the approximated local density function estimates approach to the uniform estimate (the same as that of Fig.2b) with insig- nificant error. Figure 4b compares the local probability density function estimates of the chi- squared distributed random data at the same central location (total points = 2000 too) with similar σ's. Again, the estimates converge to the uniform estimate whenever $\sigma \ge 500$. It is believed that, if the error of the random number's probability density function (Fig.1b) can be reduced, the convergent criterion of σ can be further decreased.

In order to demonstrate the overall effect, that local probability density function estimates over the range of $i = 500$ to 1500 are shown in Fig.5a and 5b for the two different distributed random number. For those point within $0 \le i$ <500 and $1500 < i \le 2000$, the left and right handed data is not long enough and are not shown in these figures. For the normal distributed random number, the uniformity is very well. On the other hand, that of the chi-squared distributed random data does not achieve the uniformly distributed estimation. Figure 5c shows the result of using $\sigma = 700$ and $i = 600$ to 1400 where the uniformity is approximately achieved.

As to the random data composed of two different distributed random data, the following random data is constructed.

$$\begin{aligned} z(i) &= x(i), && 0 \le i < 500 \\ z(i) &= (1-w)x(i) + w \times y(i), && 500 \le i \le 1500 \\ z(i) &= y(i), && 1500 < i < 2000 \\ w &= (i-500)/1000 \end{aligned} \tag{10}$$

where $x(i)$ and $y(i)$ are corresponding to the normal and chi-squared distributed random numbers, respectively. Note that both $x(i)$ and $y(i)$ are constructed via the same random number $r(i)$. Here, the central part is composed of two different distributed

random number. Within this range, distributions of the random number are different from point to point. Obviously, if the classical algorithm of estimating the probability density function is employed, the result must lead to a wrong conclusion. Figure.6a shows the local estimates over the whole range of the composed random data. The local probability density function distribution does roughly exhibit a weighted averaged distribution in the central part. That shown in Fig.6b is the same distribution but with the random data using different data string: $x(i) = r(i)$, for $i = 0, 2000$, but $y(i) = r(i + 2000)$. Although Fig.6a is slightly different from that of Fig.6b, their overall characters are quite similar. That shown in Fig.6c is the exact probability density function distributions evaluated via the following calculation which is corresponding to the that of Eq.(10) without considering the true random data.

$$
\begin{aligned}
p_w(i,j) &= p_x(i,j), && 0 \le i < 500 \\
p_w(i,j) &= \sum_{k=0}^{m} p_x(k,j)\, p_y((x_k - w \times y_k)/(1-w)), \\
& && 500 \le i \le 1500 \\
p_w(i,j) &= p_y(i,j), && 1500 < i < 2000 \\
p(i,j) &= \sum_{k=0}^{m} p_w(k,j) \exp\left[-(i-k)^2/2\sigma^2\right] \div \\
& \quad \sum_{k=0}^{m} \exp\left[-(i-k)^2/2\sigma^2\right]
\end{aligned}
\qquad (11)
$$

where j denotes the magnitude of the composed random variable, $p_w(i,j)$'s and $p(i,j)$'s are normalized in j direction for every i and are not shown here, and w takes exactly the same formula of Eq.(10). Although the detail distribution are different, its overall character is similar to that of Figs.6a and 6c. In other words, the estimates of Fig.6a and 6b approximately reflect the local probability density.

From the above discussions, although there are some small discrepancies which are believed to be caused by the pseudo-random number generator, the proposed local estimation upon the local probability density function can reflect some important properties of the random number. Therefore, it seems that further studies about this issue are of worthwhile effort.

CONCLUSIONS

The local estimation on the local probability density function is proposed. Numerical tests show that this new definition can demonstrate the probability density function's local property which cannot be reflected by the classical definition.

ACKNOWLEDGEMENT

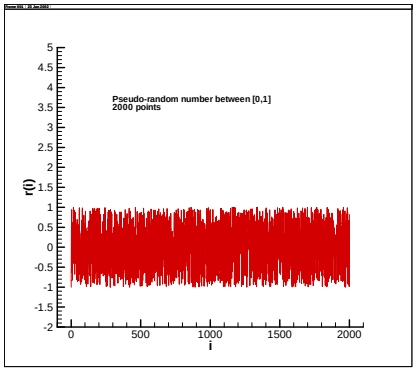
This work is supported by the Taiwan National Science Council grant no. NSC 90-2212-E-006-140.

REFERENCE

1. Press, W. H.; Flannery, B. R.; Teukolsky, S. A.; and Vetterling, W. T.,"Numerical Recipes in C, the Art of Scientific Computing," Chapter 14, Cambridge University Press, Cambridge, New York, 1988.
2. Jeng , Y. N. and Chen, C. T., "An Iterative Least ℓ_p Error Method," 1999 AASRC/CITOC/ CSCA Aerospace Joint Conference Paper no. AA-99-Fa-15, pp.155-162, 1999.
3. Jeng, Y. N. and Chen, C. T., "A Successive Error Eliminating Scheme for Simultaneous Equations of An Over-Constrained System via the Least ℓ_p Error Method, " 1999 AASRC/ CITOC/CSCA Aerospace Joint Conference Paper no. AA-99-Fa-15, pp.177-184, 1999.
4. Jeng, Y. N., "The Moving Least Squares and Least p-Power Methods for Random Data," The 7-th National Computational Fluid Dynamics Conference, P-9 to P-14, Aug. 2000.
5. Jeng, Y. N., "An Adaptive Robust Estimation Using the Least ℓ_p method," The First Taiwan-Japan Workshop on Mechanical and Aerospace Engineering, National Cheng Kung University, Tainan, Taiwan, R.O.C., Dec. 19, pp.524-534, 2001.
6. Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., Yen, N. C., Tung, C. C. and Liu, H. H., "The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-stationary Time Series Analysis,"

Proc. R. Soc. Lond. A., vol. 454, pp.903-995, 1998.
7. 鄭育能，'The High and Low Pass Filters of Non-Stationary Data via Modified Hilbert Transforms,' 中華民國力學學會第 24 屆全國力學會議論文集，H105-112, Dec. 2000.
8. Jeng, Y. N and Kuo, C. W., "Modifications upon the Huang Empirical Mode Decompo- sition," 中華民國力學學會第 25 屆全國力學會議論文集，pp.2739-2749, Dec. 2001.
9. Jeng, Y. N, "An Approximate Wave Decom- position Method," 中華民國力學學會第 25 屆全國力學會議論文集，pp.2753-2761, Dec. 2001.
10. Devore, J. L. "Probability and Statistics for Engineering and the Sciences," 5th ed. Duxbury Thomson Learning, Austrlia, 2000.
11. Bendat, J. S. and Piersol, A. G., "Random Data Analysis and Measurement Procedures," 3rd ed. John Wiley & Sons, New York, 2000.
12. Abramowitz, M., and Stegun, eds, "Handbook of Mathematical Functions," National Bureau of Standards, 1970.
13. Silverman, B. W., "Density Estimation for Statistics and Data Analysis," Chapman and Hall, London, 1986.

(1a)



(1b)

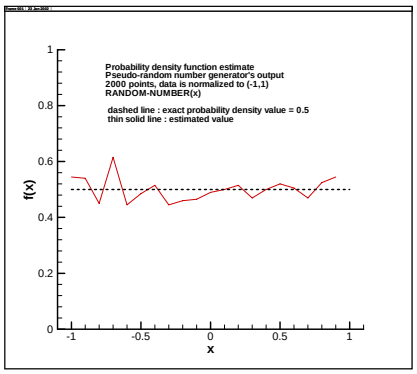
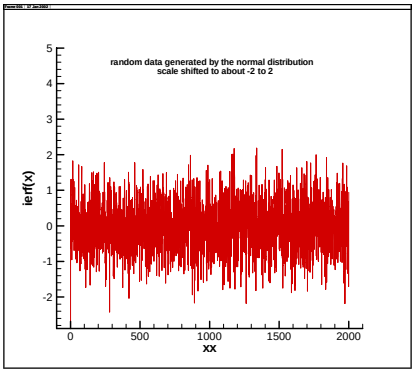


Fig.1 (a) The pseudo-random number, (b) the probability density function estimation for the result of RANDOM_NUMBER(x).

(2a)



(2b)

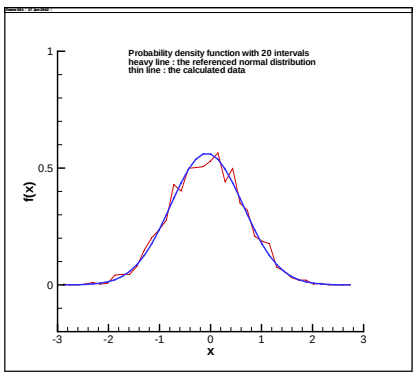
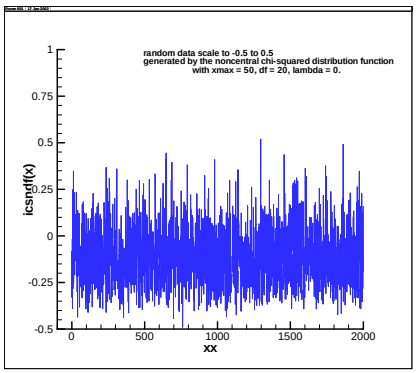


Fig.2 (a) The normal distributed random number of normal distribution; (b) the comparison between the estimated probability density function and the exact value.

(3a)



(3b)

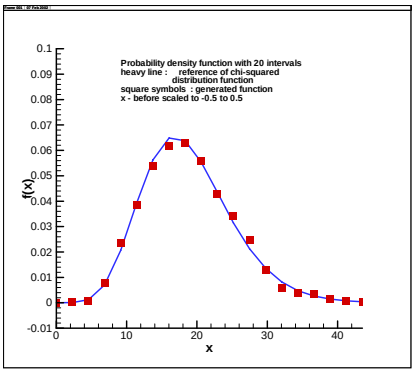
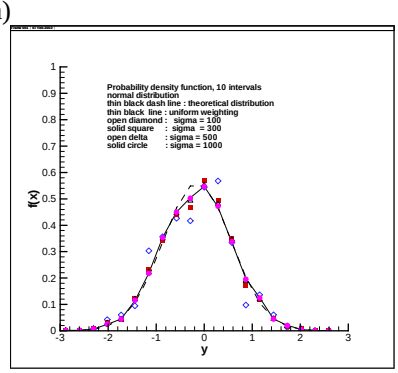


Fig.3 (a) The normal distributed random number of chi-squared distribution; (b) the comparison between the estimated probability density function and the exact value.

(4a)



(4b)

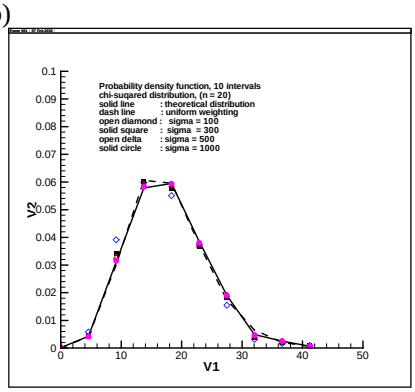
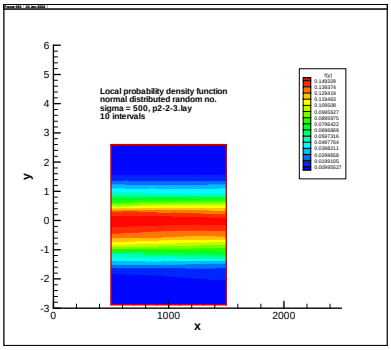
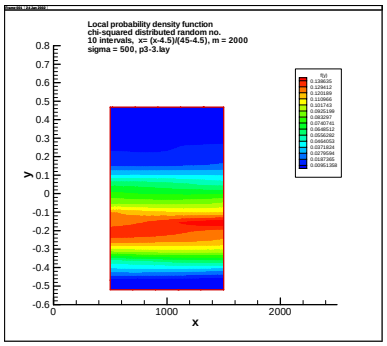


Fig.4 (a) The convergence of the local probability density function of normal distributed random number at the central point; (b) that of chi-squared distributed random number.

(5a)



(5b)



(5c)

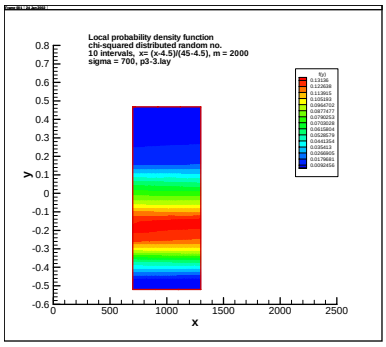
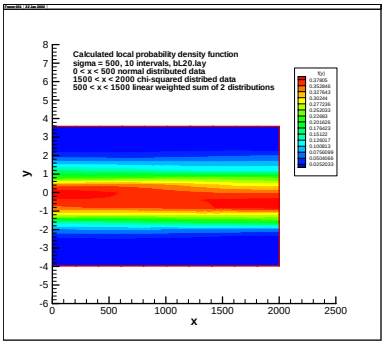
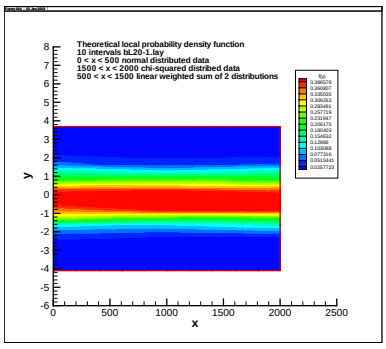


Fig.5 (a) The local probability density function distribution of the normal distributed random number from $i = 500$ to 1500, total point number = 2000, $\sigma = 500$; (b) that of the chi-squared distributed random number with $\sigma = 500$; and (c) that of the chi-squared distributed random number with $\sigma = 700$.

(6a)



(6b)



(6c)

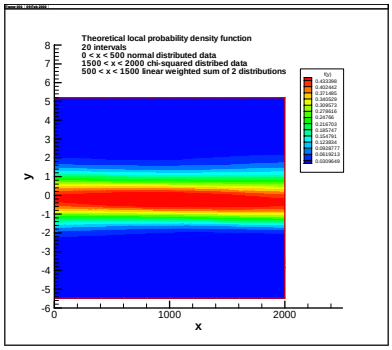


Fig.6 (a) The local probability density function of a composed random variable with $x(i)$ and $y(i)$ constructed from the same random data; (b) the same function with $x(i)$ and $y(i)$ constructed from different random data; (c) the reference exact probability density function distribution.

以數值法求取或然率密度函數的局部估算之研究

鄭育能
成功大學航太系教授
鄭又齊
台大電機系二年級學生

摘　要

本文發展一種新的局部或然率密度函數的估算法，在計算隨機變數的次數時加入高斯核函數。隨機數據是應用 FORTRAN 語言中的 RANDOM_NUMBDR(x)副程式所產生，對每一點而言，求取其密度函數時，將其它點與此點的距離 Δi 當做參數以求高斯核函數 $\exp(-(\Delta i)^2/(2\sigma^2))$，累加時不是加 1 而是加該核函數的值。數值驗證顯示核函數平滑參數 σ 足夠大時，新的估算法可反映或然率密度函數隨自變數 t 變化的情形。

關鍵詞：局部或然率密度函數，高斯核函數，不同或然率密度之隨機變數組合