The Korean Communications
in Statistics Vol. 11 No. 3, 2004
pp. 609-617

Selecting the Number and Location of Knots for Presenting Densities

Jeong Yong Ahn[1], Gill Sung Moon[2], Kyung Soo Han[3], and Beom Soo Han[4]

Abstract

To present graph of probability densities, many softwares and graphical tools use methods that link points or straight lines. However, the methods can't display exactly and smoothly the graph and are not efficient from the viewpoint of process time. One method to overcome these shortcomings is utilizing interpolation methods. In these methods, selecting the number and location of knots is an important factor. This article proposes an algorithm to select knots for graphically presenting densities and implements graph components based on the algorithm.

Keywords : Presenting Densities, Knots, Optimization Technique

1. Introduction

Probability distribution models are used widely in many disciplines such as natural science, engineering, social science and finance(Lee *et al.*, 2000; Jung, 2000, Hong, 2000; Hyun *et al.*, 2000; Lee and Peak, 2003). To represent the distribution of random variables, they are based on probability density functions(PDF). However, PDFs are defined usually with a complex numerical formula - for example, see equation (1) - and it is not easy to presume shape of distribution through the functions. To help intuitive comprehension on the shape of intricate functions such as equation (1), computer graphics are often used. As mentioned in Hyun *et al.*(2000), computer graphics are the most versatile and powerful means of communication between a computer and a human being.

$$f(x)=\frac{1}{\Gamma(r/2)2^{r/2}}x^{r/2-1}e^{-x/2}, \qquad x<\infty \tag{1}$$

To present graphs of probability densities, many statistical softwares and graphical tools use

1) Assistant Professor, Division of Mathematics and Statistical Informatics, Chonbuk National University
E-mail: jyahn@chonbuk.ac.kr
2) Graduate Student, Division of Mathematics and Statistical Informatics, Chonbuk National University
3) Professor, Division of Mathematics and Statistical Informatics, Chonbuk National University
4) Graduate Student, Division of Mathematics and Statistical Informatics, Chonbuk National University

the methods that link points or straight lines. These methods assume that the functions are approximated linearly in a fixed interval. However, the methods based on the assumption can't display exactly and smoothly the graphs and are not efficient from the viewpoint of process time.

An alternative to overcome these shortcomings is to utilize the approximation techniques using curves such as cubic spline, B-spline and the bezier curve, as they are known as efficient methods to draw graphs of probability densities(Doh and Chwa, 1993; Wegman and Carr, 1993; Ruppert, 2002). The techniques, however, have a weakness which is the precision of approximation is affected by the number and location of knots(Kim and Lee, 2001). Selecting the number and location of knots, therefore, is an important factor to present graphs of probability densities.

In this article, we propose an algorithm to select knots for graphically presenting densities and develop graph components based on the algorithm. To find the knots, the algorithm uses the optimization techniques to minimize the difference of the value of the original function and the approximated function. As a criterion to select the number and location of knots, a threshold based on the computer resolution is used.

2. The methods for presenting densities

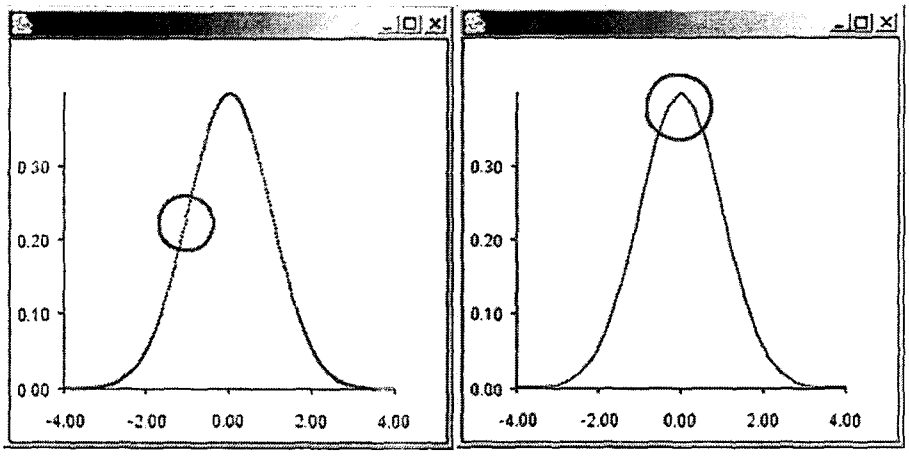
There are methods to present graphs of probability densities. In this section, we introduce the characteristics of the methods and explore matters that should be considered in the method using curves.

2.1 Traditional methods

Generally and traditionally, many statistical softwares for statistics education have used methods that link points or straight lines together to present graphs of probability densities. These methods are appropriate to roughly express the graphs. However, they have some shortcomings as follows: first, the methods can't display exactly and smoothly the graphs, especially, in the interval sections that have a high variation of curvature. Figure 1 shows the graphs using the methods. Graph (a) is an example that places points on all pixels. In the circle part, the graph is broken off. Graph (b) is an example that link straight lines together (piecewise linear fitting). In the circle part of the graph (b), the graph isn't smooth.

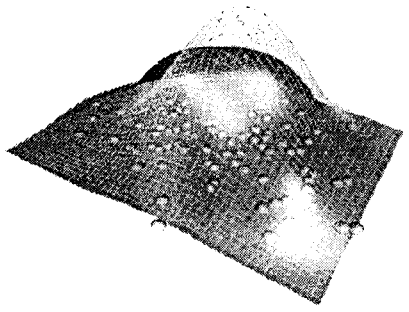
Second, the methods are not efficient from the viewpoint of process time in case that PDFs are defined with a complex numerical formula. To draw a chi-square distribution graph, for example, we ought to calculate the value of equation (1). In the case of using points and lines, the running time is approximately $O(nr/2)$ and $O(nr/2k)$ (a line per k pixels) respectively. In addition, the methods are not very efficient when a graph is updated frequently or when we draw a graph as in Figure 2.

Third, the graphs using these methods have aliasing as in Figure 3. Aliasing is the well-known effect on computer screens, in fact on all pixel devices, where diagonal and curved lines are displayed as a series of little zigzag horizontal and vertical lines. When the pixels are large, like on computer screens, some kind of remedy is highly desirable.

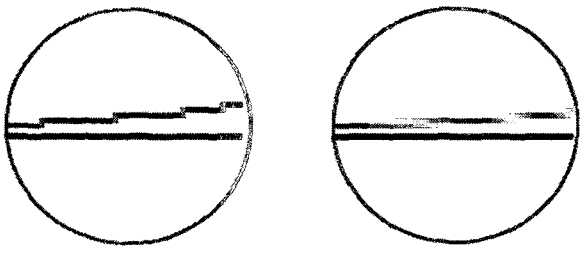


(a) Using the points (b) Using the lines

<Figure 1> Graphs of standard normal density



<Figure 2> Graph of bivariate standard normal density

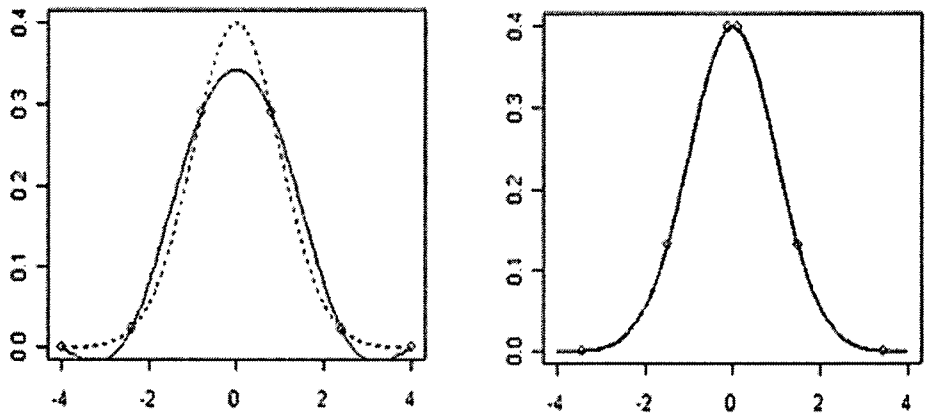


<Figure 3> Aliasing and anti-aliasing image

2.2 The method using curves

The shortcomings of traditional methods mentioned in section 2.1 can be solved by using curves, and the most common method is one that uses cubic spline interpolation. A cubic spline fits a smooth curve to sub-intervals. This device is a flexible rod, bent to conform to the intervals. The precision of the method, however, is affected by the number and location of knots.

Figure 4 shows a graph of standard normal density using 6 knots. The dotted line is the original function and the solid line is approximated. Graph (a) is an example that arranges the knots at sub-intervals with equal width. In the graph (b), knots are arranged at sub-intervals with unequal width. Approximate precision of the graph (a) is not good, especially, in the neighborhood of $x=0$. Although graph (a) and (b) of Figure 4 use an equal number of knots, approximate precision of the latter is very exact. The knots in graph (b) are arranged in the neighborhood of $x=0$ and $x=\pm 1$ where they have a high variation of curvature.



(a) sub-intervals with equal width (b) sub-intervals with unequal width

<Figure 4> Graphs of standard normal density using knots

Figure 4 gives an intimation in which it's approximate precision is very different according to the arrangement of knots though we use knots of the equal number. When we use interpolation methods for graphically presenting densities, it is important not only to select the number of knots but also to arrange the location of knots.

3. Selecting the number and location of knots

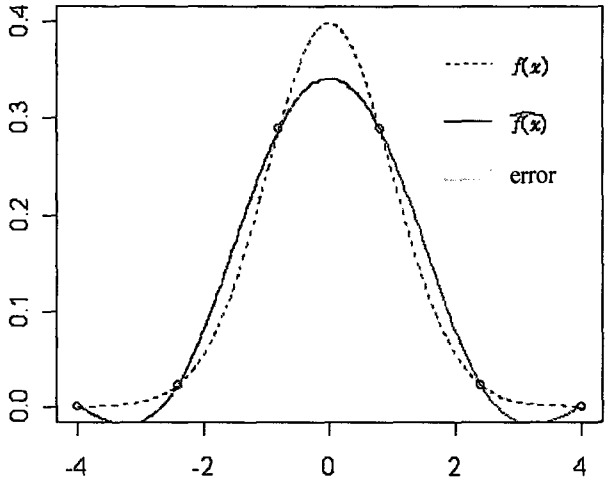
3.1 An algorithm for selecting knots

The problem of selecting appropriate knots is equal to finding knots to minimize the difference of the value and area of the original function and the approximated function as in Figure 5. This

problem can be solved by optimization techniques.

Table 1 represents the algorithm proposed in this study for selecting knots based on an optimization technique. In step ① of the algorithm, we define an object function for optimization as equation (2), where Ω is the interval of a density and knots(interpolation points) x_i satisfy $x_1 < x_2 < \ldots < x_n$ and $x_i \in \Omega$ for all $i = 1, 2, \ldots, n$. The $f(x)$ and $\hat{f}_n(x)$ are original density and cubic spline function respectively.

$$g(f, \hat{f}) = \sup_{\Omega} | f(x) - \hat{f}_n(x) | \tag{2}$$



<Figure 5> Error of approximate function

<Table 1> An algorithm for selecting knots

① Defining the object function
② Set the distribution and approximate interval
③ Set the number of knots
④ Set the initial values of knots
⑤ Find the maximum error and location of knots(interpolation points)
- using downhill simplex method
⑥ Decide the threshold
- based on the computer resolution
⑦ If maximum error ≤ threshold then
select location of knots
Else
go to step ④

Steps ② ~ ④ set up a PDF and interval for approximation, the number and initial values of knots. In step ⑤, the algorithm find the maximum error and location of knots using an optimization technique. Optimization techniques have two forms that only need evaluations of the object function to be minimized and that also require evaluations of the derivative of that function. In this study, we use the downhill simplex method that doesn't require a derivative, because the object function doesn't have a derivative. The downhill simplex method may frequently be the best method to use if the figure of merit is "get something working quickly" for a problem whose computational burden is small (Press *et al.*, 1992).

In step ⑥, we decide the threshold based on the computer resolution and maximum value of the density. Let y_p be pixels of the vertical axis, y_m be the maximum value of the density, and y_s be y_p/y_m. If we want to present less than n pixels for the difference of the original function and the approximated function, the threshold of maximum error can be decided as equation (3) and (4).

$$y_s \times \varepsilon \leq n \tag{3}$$

$$\varepsilon \leq \frac{n}{y_s} \tag{4}$$

For example, let us consider standard normal density and let the resolution of the monitor be 1024×768. The maximum value(y_m) of standard normal function is approximately 0.3989, and $y_s \cong 1925$. If we want to present the graph in less than 2 pixels, $\varepsilon \leq 0.001038$. In step ⑦, we select the location of knots based on the threshold. This process to find the location of knots, however, requires many process times. To overcome this shortcoming, we use a technique that finds the preliminary knots and store them in a database table.

<Table 2> Maximum error and knots from standard normal distribution

No. of knots	maximum error	knots
4	0.100304	-3.0300, -0.8918, 0.8918, 3.0300
5	0.001217	-3.4136, -1.5158, 0, 1.5158, 3.4136
6	0.000959	-3.4269, -1.4961, -0.0880, 0.0880, 1.4961, 3.4269
7	0.000951	-3.4285, -1.4946, -0.04783, 0, 0.04783, 1.4946, 3.4285
8	0.000612	-3.3677, -1.5638, -0.9084, -0.2956, 0.2956, 0.9084, 1.5638, 3.3677
9	0.000607	-3.3684, -1.6341, -1.2534, -0.6278, 0, 0.6278, 1.2534, 1.6341, 3.3684
10	0.000607	-3.3685, -1.6347, -1.3714, -0.8654, -0.2268, 0.2268, 0.8654, 1.3714, 1.6347, 3.3685

Table 2 gives the maximum errors and the knots from standard normal distribution. When we use the threshold $\varepsilon \cong 0.001$, the maximum errors are less than the threshold when the number of

knots is more than 6. Therefore, we use 6 knots at least to present exactly and smoothly the graph of standard normal density. Table 3 and 4 give the maximum errors from the t and χ^2 distribution. When the degrees of freedom are 1 and 30, the thresholds for presenting the t density are approximately 0.0008 and 0.001 respectively, and the thresholds for presenting χ^2 density are approximately 0.01 and 0.0001 respectively.

<Table 3> Maximum error from t distribution(d.f.: 1, 30)

No. of knots	maximum error
4	0.145939
5	0.010788
6	0.009809
7	0.005706
8	0.002464
9	0.000698
10	0.000697
11	0.000650

No. of knots	maximum error
4	0.087661
5	0.001058
6	0.000931
7	0.000930
8	0.000444
9	0.000222
10	0.000222

<Table 4> Maximum error from chi-square distribution(d.f.: 1, 30)

No. of knots	maximum error
4	0.595387
5	0.092510
6	0.063133
7	0.043288
8	0.040701
9	0.040280
10	0.039463
11	0.039293
12	0.030140
13	0.011082
14	0.009824
15	0.008009
16	0.007560
17	0.005323
18	0.002839
19	0.002054
20	0.001595

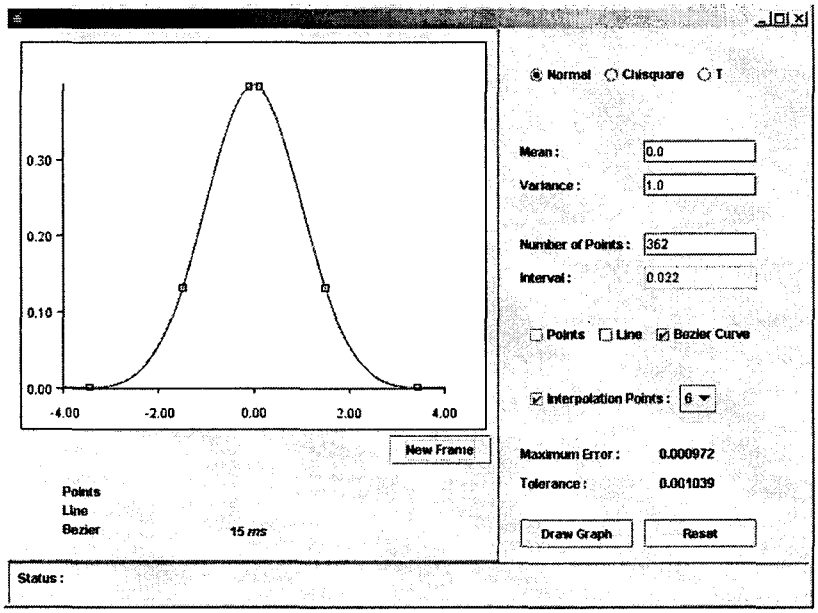
No. of knots	maximum error
4	0.011887
5	0.000226
6	0.000173
7	0.000092
8	0.000073
9	0.000042
10	0.000029
11	0.000028

3.2 Graph components and performance

To present the graphs of probability densities such as standard normal, t and χ^2 distribution, we use the bezier curve. The graph modules for the bezier curve are usually provided in the programming languages. Figure 6 represents the graph components implemented in this study.

The components are designed to be used easily in many applications through the modulization

of graph routines. In the components we can compare the methods that link points or straight lines with an interpolation method. In addition, we can shift the location of knots using the computer mouse and explore the change of the graphs.



<Figure 6> Graph component of probability densities

To assess the performance of the algorithm, we performed several experiments on an IBM PC(Pentium Ⅳ) with a CPU clock rate of 1.5GHz and 1.5GB of main memory. Table 5 shows the execution times for the three methods. The interpolation method proposed in this article did a little better than traditional methods. In particular, we expect that the method will be more efficient when a graph is updated frequently or when we draw a graph as in Figure 2. In addition, the method doesn't have aliasing because it is hardly affected by computer resolution.

<Table 5> Execution times (milisecond, d.f.: 20)

	Points	Lines	Interpolation
Normal	63.695	8.120	1.165
t	61.330	7.020	2.275
χ^2	59.340	6.485	2.585

4. Conclusions

In this paper, we propose an algorithm to select the number and location of knots for graphically presenting densities, and develop graph components based on the algorithm. The algorithm uses the optimization techniques and the threshold based on the computer resolution as a criterion to select the most appropriate knots. With the algorithm, we can present exactly and

smoothly the graphs of probability densities, and the method which uses curves is efficient from the viewpoint of process time.

To find maximum error and knots in this study, we use R, a language and environment for statistical computing and graphics, and graph modules are developed in JAVA, a programming language. The method proposed in this article, however, has a shortcoming in that it can not be applied for general functions such as the gamma function. Generalization on any functions and integration of the processes remain for further research.

References

[1] Doh, J. I. and Chwa, K. Y. (1993), An Algorithm for Determining the Internal Line Visibility of a Simple Polygon, *Journal of Algorithms*, Vol. 14, No. 1, pp. 139-168.

[2] Enderle, G., Kansy, K. and Pfaff, G. (1987), *Symbolic Computation: Computer Graphics Systems and applications*, Springer-Verlag, Berlin.

[3] Hong, S. K. (2000), Study on Enumerating the Degree of Similarity in Pairs of the Standardized Scores and Lower and Upper tail Probabilities using the Folded Normal Distribution, *The Mathematical Education*, Vol. 39, No. 2, pp. 167-177.

[4] Hyun, I. H., Kim, J. S., Lee, S. M. and Lee, I. J. (2000), The Characteristics of Probability Distribution for the Peak Day Demand Factors, *Proceedings of the Conference of the Korean Society of Water and Waste*, pp. 31-34.

[5] Jung, M. K. (2000), Estimation of Premium Rates using Poisson Probability Distribution for Livestock Insurance, *The Korean Journal of Agricultural Economics*, Vol. 41, No. 3, pp. 79-96.

[6] Kim, T. W. and Lee, K. W. (2001), Weight Control and Knot Placement for Rational B-spline Interpolation, *Korean Society of Mechanical Engineers International Journal*, Vol. 15, No. 2, 192-198.

[7] Lee, K. W., Kang, T. J. and Cho, H. J. (2000), Prediction of Laminate Composite Strength Using Probabilistic Approach, *Journal of Korean Society for Composite Materials*, Vol. 13, No. 1, pp. 33-39.

[8] Lee, S. H. and Paek, Y. T. (2003), Computer Science : Dynamic Adaptive Model Based On Probabilistic Distribution Functions And User's Profile For Web Media Systems, *The Journal of Korean Association of Computer Education*, Vol. 6, No. 1, pp. 29-39.

[9] Press, H. W., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P. (1992), *Numerical Recipes in C*, Cambridge University Press.

[10] Ruppert, D (2002), Selecting the Number of Knots for Penalized Splines, *Journal of Computational and Graphical Statistics*, Vol. 11, No. 4, 735-757.

[11] Wegman, E. J. and Carr, D. B. (1993), Statistical Graphics and Visualization, *Handbook of Statistics*, Vol. 9, 857-958.

[Received June 2004, Accepted October 2004]