Comparative Study of Some Numerical Iterations using Zero Truncated Poisson Distribution

Adegoke, T. M.[1]; Adegoke, G. K.[2]; Oduwole, H. K.[2]; Yahya, A. M.[3]

[1]Shopping Internet Services, Lagos State, Nigeria.
[2] Department of Mathematical Science,
Nasarawa State University, Keffi, Nigeria.
[3]Department of Mathematics and Statistics,
University of Maiduguri, Maiduguri, Nigeria
E-mail: [1]adegoketaiwom@gmail.com; [2]abdulstat@gmail.com; [3]adegokegkola@yahoo.com

Professional Statisticians

Abstract **— The study was aimed at comparing the rate of performance, viz-aviz, the rate of convergence of Bisection method, Newton-Raphson method and the Secant method of root-finding. The software, R programming language was used to find the root of the function, $f(x,\varphi)=\frac{\varphi^x e^{-\varphi}}{x!(1-e^{-\varphi})}$ using the Bisection method, the Newton's method and the Secant method and the result compared. The Bisection method converges at the 22 second iteration while Newton and Secant methods converge to the exact root of 0.739085 with error 0.0001 at the 5th and 6th iteration respectively. R programming language were developed to obtain the root of the distribution under study. It was then concluded that of the three methods considered, Secant method is the most effective scheme.**

Keywords- *Convergence, Roots, Algorithm, Iterations, Bisection method, Newton-Raphson method, Secant method and Truncated Poisson Distribution.*

I. Introduction

programming (GP) is a mathematical technique for solving Suppose $\pi(x,\varphi)$ is the original distribution. Then the zero-truncated version of $\pi(x,\varphi)$ is defined as

$$\pi(x;\varphi)=\frac{\pi(x;\varphi)}{1-\pi(0;\varphi)};x=1,2,3,\cdots \quad (1)$$

In probability theory, zero-truncated distribution is a certain class of discrete distribution whose support is the set of positive integers. When the data to be modelled originate from a mechanism which generates data that structurally excludes zero counts, zero-truncated distribution is the appropriate choice.

The probability mass function (PMF) of Poisson-Lindley distribution (PLD) given by

$$\pi^*(x;\varphi)=\frac{\varphi^2(x+\varphi+2)}{(\varphi+1)^{x+3}};x=0,1,2,3,\cdots \ \varphi>0 \quad (2)$$

has been introduced by [8] to model count data. It is a Poisson mixture of distribution having probability density function (PDF)

$$f(x,\varphi)=\frac{\varphi^2}{\varphi+1}(1+x)e^{-\varphi x};x>0,\varphi>0 \quad (3)$$

[5] has detailed study on various properties, estimation of parameter and application of Lindley distribution. [6] has discussed the estimation methods of PLD along with simulation study and application. [10] has detailed discussion on the applications of exponential and Lindley distributions for modelling lifetimes data from different fields of knowledge. [9] discussed the applications of Poisson-Lindley distribution in biological sciences.

Using (1) and (2), [5] obtained zero-truncated Poisson-Lindley distribution (ZTPLD) defined by its PMF

$$\pi^{**}(x;\varphi)=\frac{\varphi^2}{\varphi^2+3\varphi+1}\frac{x-\varphi+2}{(\varphi+1)^x};x=1,2,3,\cdots,\varphi>0 \quad (4)$$

[10] has done comparative study on applications of ZTPLD and zero-truncated Poisson distribution (ZTPD) on different real data sets from different fields of knowledge and showed that ZTPLD gives better fit than ZTPD in almost all data sets relating to demography, biological sciences and social sciences. The zero-truncated Poisson distribution (ZTPD) is defined by its PMF

$$f(x,\varphi)=\frac{\varphi^x e^{-\varphi}}{x!(1-e^{-\varphi})};x=1,2,3,\cdots,\varphi>0 \quad (5)$$

.

II. Research Methodology

Root finding is a root of the equation f(x)= 0, where f(x) is a function of a single variable, x. Let f(x) be a function, we are interested in finding $x=\gamma$ such that $f(\gamma)=0$. The number γ is called the root or zero of f(x). f(x) may be algebraic, trigonometric or transcendental function.

The root finding problem is one of the most relevant computational problems. It arises in a wide variety of practical applications in Physics, Chemistry, Biosciences,

© 2018, *A Publication of Professional Statisticians Society of Nigeria*

Engineering, etc. As a matter of fact, the determination of any unknown appearing implicitly in scientific or engineering formulas, gives rise to root finding problem [1]. Relevant situations in Physics where such problems are needed to be solved include finding the equilibrium position of an object, potential surface of a field and quantized energy level of confined structure [2]. The common root-finding methods include: Bisection, Newton-Raphson, False position, Secant methods etc. Different methods converge to the root at different rates. That is, some methods are faster in converging to the root than others. The rate of convergence could be linear, quadratic or otherwise. The higher the order, the faster the method converges [3]. The study is at comparing the rate of performance (convergence) of Bisection, Newton-Raphson and Secant as methods of root-finding.

Obviously, Newton-Raphson method may converge faster than any other method but when we compare performance, it is needful to consider both cost and speed of convergence. An algorithm that converges quickly but takes a few seconds per iteration may take more time overall than an algorithm that converges more slowly, but takes only a few milliseconds per iteration [11]. Secant method requires only one function evaluation per iteration, since the value of $f(x_{n+1})$ can be stored from the previous iteration [1,4]. Newton's method, on the other hand, requires one function and the derivative evaluation per iteration. It is often difficult to estimate the cost of evaluating the derivative in general (if it is possible) [1, 4-5]. It seem safe, to assume that in most cases, evaluating the derivative is at least as costly as evaluating the function [11]. Thus, we can estimate that the Newton iteration takes about two functions evaluation per iteration. This disparity in cost means that we can run two iterations of the secant method in the same time it will take to run one iteration of Newton method.

In comparing the rate of convergence of Bisection, Newton and Secant methods, [11] used C++programming language to calculate the cube roots of numbers from 1 to 25, using the three methods. They observed that the rate of convergence is in the following order: Bisection method < Newton method < Secant method. They concluded that Newton method is 7.678622465 times better than the Bisection method while Secant method is 1.389482397 times better than the Newton method

A. *Bisection Method:*

Given f(x) = 0, continuous on a closed interval [a,b], such that f(a)f(b)<0, then the function f(x) has at least a root or zero in the interval [a,b]. The method calls for a repeated halving of subintervals of [a,b] containing the root. The root always converges, though very slow in converging [12].

B. *Newton Raphson Method*

The Newton-Raphson method finds the slope (tangent line) of the function at the current point and uses the zero of the tangent line as the next reference point. The process is repeated until the root is found [5-7]. The method is probably the most popular technique for solving nonlinear equation because of its quadratic convergence rate. But it is sometimes damped if bad initial guesses are used [8-9].It was suggested however, that Newton's method should sometimes be started with Picard iteration to improve the initial guess [14]. Newton Raphson method is much more efficient than the Bisection method. However, it requires the calculation of the derivative of a function as the reference point which is not always easy or either the derivative does not exist at all or it cannot be expressed in terms of elementary function [6,7-8]. Furthermore, the tangent line often shoots wildly and might occasionally be trapped in a loop [13]. The function, f(x)=0 can be expanded in the neighbourhood of the root x_0 through the Taylor expansion

$$f(x_0) \approx f(x) + (x_0 - x)f'(x) + \frac{(x_0 - x)^2}{2!} f''(\gamma(x_0)) = 0$$

where x can be seen as a trial value for the root at the nth step and the approximate value of the next step x_{k+1}can be derived from $f(x_{k+1}) = f(x_k) + (x_{k+1} - x_k)f'(x_k) = 0$.

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}; k = 0,1,2,\cdots \quad (6)$$

called the Newton- Raphson method

C. *Secant Method*

As we have noticed, the main setback of the Newton-Raphson method is the requirement of finding the value of the derivative of f(x) at each iterations. There are some functions that are either extremely difficult (if not impossible) or time consuming. The way out of this, according to [1] is to approximate the derivative by knowing the values of the function at that and the previous approximation. Knowing f(x_{k-1}) we can then approximate f'(x) as

$$f'(x_k) \approx x_k - \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} \quad (7)$$

putting (7) into the Newton iteration we have: $x_{k+1} \approx x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}$ (8)

which is known as Secant iteration.

D. *Simulation Design*

A simulation study was carried out in order to be able to estimate the parameter on interest. Using R programming

Professional Statisticians

language was used to simulate a sample size of 50 from a Poison distribution with mean 1;

III. Results And Discussion

A. Results

The Bisection, Newton-Raphson and Secant methods were applied to a zero truncated Poisson distribution (5). Since this is an iterative process, it worthwhile to use a tool to implement the above methods in other to save time. Functions were developed for each of the iterative methods using R Programming Language. A random sample of 50 dataset were simulated from a Poisson distribution with mean (φ) 1. The R codes used in implementing Bisection method, Newton Raphson method and Secant method are shown blow and the results are presented in Tables 1 to 3.

```
set.seed(10000)
 yp <- rpois (50,1)
bisection=function(a,b,n){
   xa=a
   xb=b
   for(i in 1:n){ if(gd(xa)*gd((xa+xb)/2)<0)
xb=(xa+xb)/2
   else xa=(xa+xb)/2}
   list(left=xa,right=xb,
midpoint=(xa+xb)/2)
}
 ld=function(x,t){
   s=0
   n=length(x)
   for(j in 1:n)
     s = s+ (x[i]*log(t)-log(factorial(x)))
    l = -n*t-n*log(1-exp(-t))+s
   l
 }
 gd <- function(lam) {( lam -ybar *(1- exp(-
lam ) ) ) /
      (1- ybar * exp(-lam))}
 for (i in 1:30) {
   x<- bisection(0,1,i)
   bi[i] <- x$midpoint
   bl[i] <-x$left
   br[i]<-x$right
 }
 bi
 funcr <- f(br)
 funcl <- f(bl)
```

Table 1: Iteration Data for Bisection Method

Step	a	f(a)	b	F(b)
0	0.5	-0.399999	1.000	0.24517
1	0.5	-0.309599	0.75	0.04409
2	0.625	-0.087642	0.75	0.04409
3	0. 6750	-0.016636	0.75	0.04409
4	0.6750	-0.016636	0.71875	0.01469
5	0.6875	-7.0167e-4	0.71875	0.01469
6	0.703125	-7.0167e-4	0.7109375	0.00706
7	0.703125	-7.0167e-4	0.7070312	0.00319
8	0.703125	-7.0167e-4	0.7050781	0.00125
9	0.703125	-7.0167e-4	0.7041061	0.00028
10	0.703125	-7.0167e-4	0.7041016	0.00028
11	0.7036133	-2.1289e-4	0.7038574	3.12e-5
12	0.7036133	-2.1289e4	0.7038574	3.12e-5
13	0.7037354	-2.0779e-5	0.7038574	3.12e-5
14	0.7037964	-2.9736e-5	0.7038269	7.82e-7
15	0.7037964	-2.9736e-5	0.7038269	7.82e-7
16	0.7038000	-1.4477e-5	0.7038269	7.82e-7
17	0.7038193	-6.3873e-6	0.7038269	7.82e-7
18	0.7038231	-3.033e-6	0.7038269	7.82e-7
19	0.7003825	-1.1252e-6	0.7038269	7.82e-7
20	0.7038260	-1.7155e-7	0.7038264	3.05e-7
21	0.7038260	-1.7155e-7	0.7038262	6.69e-8
22	0.8038260	-1.7155e-7	0.7038262	6.69e-8
23	0.7038261	-5.2342e-8	0.7038261	7.26e-9
24	0.7038261	-5.2342e-8	0.7038261	7.26e-9

Table 2: Iteration Data for Newton- Raphson Method

Step	x_k	$F(x_{k+1})$
1	0.839536	0.1408867
2	0.723069	0.5310635
3	0.7042146	0.0003883108
4	0.7038263	0.0000001778265
5	0.7038261	-0.000000022173648
6	0.7038263	0.0000001778265

Table 3: Iteration Data for Secant Method

		f(x)
x_0	0.6665610	-0.03898724
x_1	0.7175561	-0.01352910
X_2	0.7044189	-0.0005923909
X_3	0.7038173	-0.000008792124
x_4	0.7038261	0.00000000577991
x_5	0.7038261	0.00000000577991

© 2018, *A Publication of Professional Statisticians Society of Nigeria*

```
y <- yp[yp >0]
 ybar <- mean(y) ; ybar
 lam <- ybar ; it <- 0 ; step <- 1
       while (abs(step) > 0.0001 && ( it <-
      it + 1) < 10) {
 step <- ( lam - ybar *(1 - exp(-lam ) ) ) /
      (1- ybar * exp(-lam ) )
 lam <- lam - step
      cat ( it , lam , "\n ")
             }
      ### Secant Method
      f=function(lam){( lam - ybar *(1 -
      exp(-lam ) ) ) /
     (1- ybar * exp(-lam ) )}
 g=function(x,y){y-(f(y)/(f(x)-f(y)))*(x-y)}
       h=function(x,y,n){ # Katherine
      Earles's code
      xa=x
      xb=y
      xc=0
   for(i in (1:n)){if
(identical(all.equal(xa, xb), TRUE)) break
     else # or {xc=g(xa,xb)}&{
     xa=xb}&{xb=xc}
       xc=g(xa,xb)
     xa=xb
     xb=xc
   }
   list("x(n)"=xa,"x(n+1)"=xb)}
      for (i in 1:30) {
   S<- h(1,2,i)
       sec1[i] <- S$`x(n)`
   sec2[i]<- S$`x(n+1)
      fsec1= f(sec1)
      fsec2= f(sec2)
```

Table 1 shows the iteration data obtained for Bisection method with the aid of R. It was observed in Table 1 that the Bisection method converges to 0.7038261 at the 24th iteration, Table 2 revealed that the function (2) converges to 0.7038261 at the 5th iteration and From Table 3, we noticed that the function converges to 0.7038261 after the 5th iteration.

B. *Discussion*

Comparing the results in Tables 1-3, we observed that the rates of convergence of the methods are in the following order: Secant method > Newton-Raphson method > Bisection method. This supports the claims of [11], Newton's method may converge faster than Secant method (order 2 as against α=1.6 for Secant). However, Newton's method requires the evaluation of both the function $f(x)$ and its derivative at every iteration while Secant method only requires the evaluation of $f(x)$. Hence, Secant method may occasionally be faster in practice as in the case of our study (see Tables 1 - 3) [10, 11]. So, on this premises we can claim that Secant method is faster than the Newton's method in terms of the rate of convergence.

IV. Conclusion

Based on our results and discussions, we now conclude that the Secant method is formally the most effective of the methods we have considered here in the study. This is sequel to the fact that it has a converging rate close to that of Newton-Raphson method, but requires only a single function evaluation per iteration.

It can also be concluded that though the convergence of Bisection is certain, its rate of convergence is too slow and as such it is quite difficult to extend its use for systems of equations.

Acknowledgment

The authors are grateful to anonymous reviewers for their valuable comments on the original draft of this manuscript.

References

[1] Biswa Nath Datta (2012), Lecture Notes on Numerical Solution of root Finding problems.www.math.niu.edu/dattab.

[2] Iwetan, C.N, Fuwape,I.A, Olajide, M.S and Adenodi, R.A(2012), Comparative Study of the Bisection and Newton Methods in solving for Zero and Extremes of a Single-Variable Function. J. of NAMP, 21, 173-176.

[3] Charles A and W.W Cooper (2008), Non-linear Power of Adjacent Extreme Points Methods in Linear Programming, Econometrica, 25(1), 132-153.

[4] Ehiwario, J.C. and Agimie S.O. (2014), comparative Study of Bisection, Newton-Raphson and Secant Methods of Root-Findings Problems. IOSR Journal of Engineering, 4, 01-07.

[5] Ghitany, M.E., Atieh, B. and Nadarajah, S. (2008a): Lindley distribution and Its Applications, Mathematics Computation and Simulation, 78, 493 -506.

[5] Ghitany, M.E., Al-Mutairi, D.K. and Nadarajah, S. (2008b): Zero-truncated Poisson-Lindley distribution and its Applications, Mathematics and Computers in Simulation, 79 (3), 279 – 287.

[6] Lindley, D.V. (1958): Fiducial distributions and Bayes' theorem, Journal of the Royal Statistical Society, Series B, 20, 102-107.

[7] [8] Sankaran, M (1970): The discrete Poisson-Lindley distribution, Biometrics, 26, 145 – 149.

[8] Shanker, R. and Hagos, F. (2015): On Poisson-Lindley distribution and Its Applications to Biological Sciences, Biometrics & Biostatistics International Journal, 2(4), 1– 5.

[9] Shanker, R., Hagos, F. and Sujatha, S. (2015): On modeling of Lifetimes data using exponential and Lindley distributions, Biometrics & Biostatistics International Journal, 2(5), 1-9.

Professional Statisticians

[10] Srivastava,R.B and Srivastava, S (2011), Comparison of Numerical Rate of Convergence of Bisection, Newton and Secant Methods. Journal of Chemical, Biological and Physical Sciences, 2(1) , 472-479.

[11] Ehiwario, J.C (2013), Lecture Notes on MTH213: Introduction to Numerical Analysis, College of Education Agbor. An Affiliate of Delta State University, Abraka.

[12] http://www.efunda.com/math/num_rootfinding-cfm February,2014

[13] Allen, M.B and Isaacson E.L (1998), Numerical Analysis for Applied Science. John Wiley and sons, 188-195

Professional Statisticians