

Software Reliability Using SPRT: Burr Type III Process Model

CH. Smitha¹, R. Satya Prasad², R. Kiran Kumar³

¹Department of Computer Science, Krishna University, Andhra Pradesh, India

^{2,3}CSE, Acharya Nagarjuna University, Andhra Pradesh, India

Article Info

Article history:

Received Jun 11, 2016

Revised Aug 18, 2016

Accepted Sep 1, 2016

Keyword:

Burr type III model

ML estimation

Software failure data

Software reliability

Time domain data

ABSTRACT

Increased dependence on software systems elicited the assessment of their reliability, a crucial task in software development. Effective tools and mechanisms are required to facilitate the assessment of software reliability. Classical approaches like hypothesis testing are significantly time consuming as the conclusion can only be drawn after collecting huge amounts of data. Statistical method such as Sequential Analysis can be applied to arrive at a decision quickly. This paper implemented Sequential Probability Ratio Test (SPRT) for Burr Type III model based on time domain data. For this, parameters were estimated using Maximum Likelihood Estimation to apply SPRT on five real time software failure datasets borrowed from different software projects. The results exemplify that the adopted model has given an acceptance decision for the used datasets.

Copyright © 2016 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

CH. Smitha Chowdary,
Department of Computer Science,
P.B.Siddhartha College of Arts & Science PG.Centre,
Moghalrajpuram, Vijayawada,
Andhra Pradesh-520010, India.
Email: smithacsc@gmail.com

1. INTRODUCTION

Software reliability is probability of fault free operations provided by the software product under consideration over a specified period of time in a specified operational environment [1]. A good reliability model can predict software reliability action accurately, and this is of great importance to software resource allocation and software market decision making [2]. Since 1970s, research on software reliability model domain greatly improved, many models have been put into practice, and software reliability has stepped into engineering stage from conceptual stage [3]. Considering testing effort in reliability modeling process may further improve the fitting and prediction results of software reliability growth models (SRGMs) [4]. However, facing increasing complexity of software and their development course, reliability models still appear to have inherent shortcomings [5].

Software reliability assessment needs effective tools/mechanisms. In classical Hypothesis Testing, once the entire data has been collected, the analysis is done and conclusions are drawn. If classical testing strategies are used (no usage testing), the application of software reliability growth models may be difficult and reliability predictions can be misleading. However statistical methods can be successfully applied to the failure data [6]. Sequential analysis is a method of statistical inference where number of observations required by the procedure is not determined in advance of the experiment. The decision to terminate the experiment depends, at each stage, on the results of the observation previously made. A merit of sequential method, as applied to testing statistically a hypothesis, is that a test procedure can be constructed which requires on average a small number of observations that equally test the reliability of the procedure based on a predetermined number of observations [7-8]. Stieber's observations are demonstrated by applying the well-known Sequential Probability Ratio Test (SPRT) of Wald [9] for a software failure data to detect unreliable software components and compare the reliability of different software versions.

Software reliability analysis needs software failure data. There are two types of failure data: time-domain data and interval-domain data. The time-domain data records the individual times at which the failures have occurred. The interval-domain data counts the number of failures occurring during a fixed time period. With existing software reliability models, time-domain data provides better accuracy in the estimation of parameters, but involves more data collection efforts [10]. The probability equation of the stochastic process representing the failure occurrences is given by a homogeneous Poisson process with the expression.

$$P[N(t) = n] = \frac{[\lambda t]^n}{n!} e^{-\lambda t} \quad (1)$$

A number of methods are extant for describing the software reliability based on the SPRT [11]-[13]. This paper describes a method for detecting reliable software based on the SPRT, using Maximum Likelihood Estimation (MLE) of parameter estimation. The Wald's SPRT procedure can be used to distinguish the software under test into one of the two categories like reliable/unreliable, pass/fail and certified/uncertified [14]. SPRT is the optimal statistical test that makes the correct decision in the shortest time among all tests that are subject to the same level of decision errors [15]. SPRT is used to detect the fault based on the calculated likelihood of the hypotheses. We consider one of the popular software reliability growth model Burr Type III and adopted the principle of Stieber [6] in detecting whether the software is reliable or unreliable in order to accept or reject the developed software.

The theory proposed by Stieber is described in Section 2. Implementation of SPRT for the proposed Burr type III Software Reliability Growth Model is illustrated in Section 3. Maximum Likelihood estimation method is used to estimate the parameters is presented in Section 4. Application of the decision rule to detect the unreliable software with reference to the Software Reliability Growth Model Burr Type III is depicted in Section 5.

2. WALD'S SEQUENTIAL TEST FOR A POISSON PROCESS

The Sequential Probability Ratio Test (SPRT) was developed by Abraham Wald at Columbia University in 1943 [9]. The SPRT procedure is used for quality control studies during the manufacturing of software products. The tests can be performed on fixed sample size sets with fewer observations. The SPRT methodology for Homogeneous Poisson Process is described below.

Let $\{N(t), t \geq 0\}$ be a homogeneous Poisson process with rate ' λ '. In this case, $N(t)$ = number of failures up to time ' t ' and ' λ ' is the failure rate (failures per unit time). If the system is put on test and that if we want to estimate its failure rate ' λ '. We cannot expect to estimate ' λ ' precisely. But we want to reject the system with a high probability if the data suggest that the failure rate is larger than λ_1 and accept it with a high probability, if it is smaller than λ_0 . Here we have to specify two (small) numbers ' α ' and ' β ', where ' α ' is the probability of falsely rejecting the system. That is rejecting the system even if $\lambda \leq \lambda_0$. This is the "producer's" risk. ' β ' is the probability of falsely accepting the system. That is accepting the system even if $\lambda \leq \lambda_1$. This is the "consumer's" risk. Wald's classical SPRT is very sensitive to the choice of relative risk required in the specification of the alternative hypothesis. With the classical SPRT, tests are performed continuously at every time point as $t > 0$ additional data are collected. With specified choices of λ_0 and λ_1 such that $0 < \lambda_0 < \lambda_1$, the probability of finding $N(t)$ failures in the time span $(0, t)$ with λ_1, λ_0 as the failure rates are respectively given by

$$P_1 = \frac{e^{-\lambda_1 t} [\lambda_1 t]^{N(t)}}{N(t)!} \quad (2)$$

$$P_0 = \frac{e^{-\lambda_0 t} [\lambda_0 t]^{N(t)}}{N(t)!} \quad (3)$$

The ratio $\frac{P_1}{P_0}$ at any time ' t ' is considered as a measure of deciding the truth towards λ_0 or λ_1 , given a sequence of time instants say $t_1 < t_2 < t_3 < \dots < t_K$ and the corresponding realizations $N(t_1), N(t_2), \dots, N(t_K)$ of $N(t)$. Simplification of $\frac{P_1}{P_0}$ gives

$$\frac{P_1}{P_0} = \exp(\lambda_0 - \lambda_1)t + \left(\frac{\lambda_1}{\lambda_0}\right)^{N(t)}$$

The decision rule of SPRT is to decide in favor of λ_1 , in favor of λ_0 or to continue by observing the number of failures at a later time than 't' according as $\frac{P_1}{P_0}$ is greater than or equal to a constant say A, less than or equal to a constant say B or in between the constants A and B. That is, we decide the given software product as unreliable, reliable or continue [16] the test process with one more observation in failure data, according to

$$\frac{P_1}{P_0} \geq A \quad (4)$$

$$\frac{P_1}{P_0} \leq B \quad (5)$$

$$B < \frac{P_1}{P_0} < A \quad (6)$$

The approximate values of the constants A and B are taken as

$$A \cong \frac{1 - \beta}{\alpha}, \quad B \cong \frac{\beta}{1 - \alpha}$$

where ' α ', and ' β ', are the risk probabilities as defined earlier. A simplified version of the above decision processes is to reject the system as unreliable if $N(t)$ falls for the first time above the line

$$N_U(t) = at + b_2 \quad (7)$$

To accept the system to be reliable if $N(t)$ falls for the first time below the line

$$N_L(t) = at - b_1 \quad (8)$$

To continue the test with one more observation on $(t, N(t))$ as the random graph of $[t, N(t)]$ is between the two linear boundaries given by equations (7) and (8) where

$$a = \frac{\lambda_1 - \lambda_0}{\log\left(\frac{\lambda_1}{\lambda_0}\right)} \quad (9)$$

$$b_1 = \frac{\log\left[\frac{1-\alpha}{\beta}\right]}{\log\left(\frac{\lambda_1}{\lambda_0}\right)} \quad (10)$$

$$b_2 = \frac{\log\left[\frac{1-\beta}{\alpha}\right]}{\log\left(\frac{\lambda_1}{\lambda_0}\right)} \quad (11)$$

The parameters α, β, λ_0 and λ_1 can be chosen in several ways. One way suggested by Stieber is

$$\lambda_0 = \frac{\lambda \cdot \log(q)}{q-1}, \quad \lambda_1 = q \frac{\lambda \cdot \log(q)}{q-1} \quad \text{where } q = \frac{\lambda_1}{\lambda_0}$$

If λ_0 and λ_1 are chosen in this way, the slope of NU (t) and NL (t) equals λ . The other two ways of choosing λ_0 and λ_1 are from past projects (for a comparison of the projects) and from part of the data to compare the reliability of different functional areas (components).

3. SEQUENTIAL TEST FOR SOFTWARE RELIABILITY GROWTH MODELS

We know that for any Poisson process, the expected value of $N(t) = \lambda(t)$ called the average number of failures experienced in time 't'. Which is also called the mean value function of the Poisson process. On the other hand if we consider a Poisson process with a general function (not necessarily linear) $m(t)$ as its mean value function the probability equation of a such a process is

$$P[N(t) = Y] = \frac{[m(t)]^y}{y!} \cdot e^{-m(t)}, \quad y = 0, 1, 2, \dots$$

Depending on the forms of $m(t)$ we get various Poisson processes called NHPP, for the Burr Type III model. The mean value function is given as

$$m(t) = a[1 + t^{-c}]^{-b}$$

It can also be written as

$$P_1 = \frac{e^{-m_1(t)} \cdot [m_1(t)]^{N(t)}}{N(t)!}$$

$$P_0 = \frac{e^{-m_0(t)} \cdot [m_0(t)]^{N(t)}}{N(t)!}$$

where $m_1(t)$, $m_0(t)$ represents the mean value function of stated parameters indicating reliable software and unreliable software respectively. The mean value function $m(t)$ comprises the parameters 'a', 'b' and 'c'. The two specifications of NHPP for b are considered as b_0 , b_1 where ($b_0 < b_1$) and two specifications of c say c_0 , c_1 where ($c_0 < c_1$). For our proposed model, $m(t)$ at b_1 is said to be greater than b_0 and $m(t)$ at c_1 is said to be greater than c_0 . The same can be denoted symbolically as $m_0(t) < m_1(t)$. The implementation of SPRT procedure is illustrated below.

System is said to be reliable and can be accepted if

$$\frac{P_1}{P_0} \leq B$$

$$\text{i.e., } \frac{e^{-m_1(t)} \cdot [m_1(t)]^{N(t)}}{e^{-m_0(t)} \cdot [m_0(t)]^{N(t)}} \leq B$$

$$\text{i.e., } N(t) \leq \frac{\log\left(\frac{\beta}{1-\alpha}\right) + m_1(t) - m_0(t)}{\log m_1(t) - \log m_0(t)} \quad (12)$$

System is said to be unreliable and rejected if

$$\frac{P_1}{P_0} \geq A$$

$$\text{i.e., } N(t) \geq \frac{\log\left(\frac{1-\beta}{\alpha}\right) + m_1(t) - m_0(t)}{\log m_1(t) - \log m_0(t)} \quad (13)$$

Continue the test procedure as long as

$$\frac{\log\left(\frac{\beta}{1-\alpha}\right) + m_1(t) - m_0(t)}{\log m_1(t) - \log m_0(t)} < N(t) < \frac{\log\left(\frac{1-\beta}{\alpha}\right) + m_1(t) - m_0(t)}{\log m_1(t) - \log m_0(t)} \quad (14)$$

Substituting the appropriate expressions of the respective mean value function, we get the respective decision rules and are given in followings lines.

Acceptance Region

$$N(t) \leq \frac{\log\left(\frac{\beta}{1-\alpha}\right) + a((1+t^{-c_1})^{-b_1} - (1+t^{-c_0})^{-b_0})}{\log\left[\frac{(1+t^{-c_1})^{-b_1}}{(1+t^{-c_0})^{-b_0}}\right]} \quad (15)$$

Rejection Region:

$$N(t) \geq \frac{\log\left(\frac{1-\beta}{\alpha}\right) + a((1+t^{-c_1})^{-b_1} - (1+t^{-c_0})^{-b_0})}{\log\left[\frac{(1+t^{-c_1})^{-b_1}}{(1+t^{-c_0})^{-b_0}}\right]} \quad (16)$$

Continuation Region:

$$\frac{\log\left(\frac{\beta}{1-\alpha}\right) + a((1+t^{-c_1})^{-b_1} - (1+t^{-c_0})^{-b_0})}{\log\left[\frac{(1+t^{-c_1})^{-b_1}}{(1+t^{-c_0})^{-b_0}}\right]} N(t) < \frac{\log\left(\frac{1-\beta}{\alpha}\right) + a((1+t^{-c_1})^{-b_1} - (1+t^{-c_0})^{-b_0})}{\log\left[\frac{(1+t^{-c_1})^{-b_1}}{(1+t^{-c_0})^{-b_0}}\right]} \quad (17)$$

For the specified model, it may be observed that the decision rules are exclusively based on the strength of the sequential procedure (α, β) and the value of the mean value functions namely $m_0(t)$ $m_1(t)$. As described by Stieber, these decision rules become decision lines if the mean value function is linear in passing through origin, that is $m(t) = \lambda t$. The equations (12) and (13) are considered as generalizations for the decision procedure of Stieber. SPRT procedure is applied on live software failure data sets and the results that were analyzed are illustrated in Section 5.

4. PARAMETER ESTIMATION

We present expressions for the parameter estimates of the Burr type III model. Parameter estimation is very significant in software reliability prediction. Once the analytical solution form is known for a given model, parameter estimation is achieved by applying a well-known estimation, Maximum Likelihood Estimation (MLE). The main idea behind Maximum Likelihood parameter assessment is to decide the parameters that maximize the probability (likelihood) of the specimen data. In the other words, MLE methods are versatile and applicable to most models and for different types of data. Here parameters are estimated from the time domain data [17].

The mean value function of Burr type III model is given by

$$m(t) = a[1 + t^{-c}]^{-b} \quad t > 0, a, b, c > 0 \quad (18)$$

The parameters a, b, c are estimated with Maximum Likelihood (ML) estimation. The likelihood function for time domain data is given by

$$LLF = \sum_{i=1}^n \log[\lambda(t_i)] - m(t_n) \quad (19)$$

Substituting Equation (18) in equation (19) we get

$$\begin{aligned} & \frac{1 + t_i^{-c} \quad b+1}{abc} \\ & \frac{t_i^{c+1} - \frac{a}{[1 + t_n^{-c}]^b}}{\log} \\ \text{LogL} &= \sum_{i=1}^n \\ \text{LogL} &= \frac{-a}{(1+t_n^{-c})^b} + \sum_{i=1}^n [\log a + \log b + \log c - (c+1)\log t_i - (b+1)\log(1+t_i^{-c})] \end{aligned} \quad (20)$$

Taking the Partial derivative with respect to 'a' and equating to '0'.

$$\begin{aligned} & \frac{\partial \text{LogL}}{\partial a} = 0 \\ \Rightarrow a &= n(1 + t_n^{-c})^b \end{aligned} \quad (21)$$

Taking the Partial derivative with respect to 'b' and equating to '0'.

$$\begin{aligned} & \frac{\partial \text{LogL}}{\partial b} = 0 \\ \Rightarrow b &= \frac{n}{\sum_{i=1}^n \log(1+t_i^{-1}) - n \log(1+t_n^{-1})} \end{aligned} \quad (22)$$

The parameter 'c' is estimated by iterative Newton-Raphson Method using $c_{i+1} = c_i - \frac{g(c_i)}{g'(c_i)}$ where $g(c)$ and $g'(c)$ are expressed as follows.

$$\begin{aligned} & \frac{\partial \text{LogL}}{\partial c} = 0 \\ \Rightarrow g(c) &= \frac{-n \log(t_n)}{1+t_n^c} + \frac{n}{c} + \sum_{i=1}^n \log t_i \left[-1 + \frac{2}{1+t_i^c} \right] \end{aligned} \quad (23)$$

$$\begin{aligned} & \frac{\partial^2 \text{LogL}}{\partial c^2} = 0 \\ \Rightarrow g'(c) &= \frac{n(\log t_n)^2 t_n^c}{(t_n^c + 1)^2} - \frac{n}{c^2} - \sum_{i=1}^n \frac{2t_i^c (\log t_i)^2}{(t_i^c + 1)^2} \end{aligned} \quad (24)$$

5. SPRT ANALYSIS OF LIVE DATASETS

SPRT methodology is applied on five different data sets that are borrowed from pham [18], Iyu [19] and SONATA [20] software services. The decisions are evaluated based on the considered mean value function (18). Based on the estimates of the parameters 'b' and 'c' in each mean value function, we have chosen the specifications of $b_0 = b - \delta$, $b_1 = b + \delta$ and $c_0 = c - \delta$, $c_1 = c + \delta$, and apply SPRT such that $b_0 < b < b_1$ and $c_0 < c < c_1$. Assuming the δ value of 0.6 the choices are given in Table 1.

Using the specification b_0 , b_1 , and c_0 , c_1 the mean value functions $m_0(t)$ and $m_1(t)$ are computed for each 't'. Later the decisions are made based on the decision rules specified by the equations (15), (16), (17) for the data sets. At each 't' of the data set, the strengths (α , β) are considered as (0.3, 0.3). SPRT procedure is applied on five different data sets and the necessary calculations are given in Table 2.

Table 1. Estimates of a, b, c & Specifications of b_0 , b_1 , c_0 , c_1

Data sets	Estimate of 'a'	Estimate of 'b'	b_0	b_1	Estimate of 'c'	c_0	c_1
NTDS	34.465706	1.763647	1.163647	2.363647	1.810222	1.210222	2.410222
AT&T	26.839829	1.658692	1.058692	2.258692	1	0.4	1.6
SONATA	79.831359	6.74281	6.14281	7.34281	0.60244	0.00244	1.20244
XIE	33.310426	2.270095	1.670095	2.870095	1.371974	0.771974	1.971974
IBM	20.624785	1.71163	1.11163	2.31163	1.447815	0.847815	2.047815

Table 2. SPRT Analysis for 5 Data Sets

Data Set	T	N(t)	R.H.S. of equation (15) Acceptance region (\leq)	R.H.S. of equation (16) Rejection region (\geq)	Decision
NTDS	9	1	22.16983832	2.790902447	ACCEPT
AT&T	5.5	1	3.798845246	2.84300646	ACCEPT
SONATA	52.5	1	16.8099181	2.238720666	ACCEPT
XIE	30.02	1	3.488345958	2.274061395	ACCEPT
IBM	10	1	4.061265728	1.670408722	ACCEPT

It may be noted that the decision is obtained in significantly lesser number of iterations N(t) in the proposed model when drawn in comparison with other models [11] based on SPRT.

6. CONCLUSION

The SPRT methodology for the proposed software reliability growth model Burr type III is applied for the software failure data sets. From the observation we are able to come to a conclusion in a very less time regarding the reliability or unreliability of a software product. The results obtained from the datasets exemplify that the model has given a decision of acceptance for all the data sets at very first time instance of the data. Hence, we may conclude that, by applying SPRT on data sets we can come to an early decision of reliable/unreliable of software.

REFERENCES

- [1] S. S. Marinos, *et al.*, "Important Milestones in Software Reliability Modeling," in *Proceedings of Software Engineering and Knowledge Engineering (SEKE' 96), Lake Tahoe, NV*, pp. 345-352, 1996.
- [2] Rebello, *et al.*, "Software system reliability and safety assessment: An extended FMEA approach," *International Journal of Reliability and Safety*, vol/issue: 20(4), pp. 366-380, 2010.
- [3] Y. Zhang and H. Cheng, "Improved Genetic Programming Algorithm Applied to Symbolic Regression and Software Reliability Modeling," *Software Engineering & Applications*, vol. 2, pp. 354-360, 2009.
- [4] Z. Qian, *et al.*, "Software Reliability Modeling with Testing-Effort Function and Imperfect Debugging," *Indonesian Journal of Electrical Engineering and Computer Science*, vol/issue: 10(8), pp. 1992-1998, 2012.
- [5] S. J. Sun and J. Xiao, "A Software Reliability GEP Model Based on Usage Profile," *TELKOMNIKA*, vol/issue: 10(7), pp. 1756~1764, 2012. e-ISSN: 2087-278X.
- [6] H. A. Stieber, "Statistical Quality Control: How To Detect Unreliable Software Components," *Proceedings the 8th International Symposium on Software Reliability Engineering*, pp. 8-12, 1997.
- [7] Card D., "Statistical Process Control for Software," *IEEE Software*, pp. 95-97, 1994.
- [8] J. D. Musa, "Software Quality and Reliability Basics," *AT&T Bell Laboratories*, 1994. CH 2468-7/87/0000/014,1987 IEEE.
- [9] Wald A., "Sequential Analysis," New Impression edition, New York, John Wiley and Son, Inc, 1947.
- [10] K. B. Misra, "Handbook of Performability Engineering," Springer, 2008.
- [11] R. S. Prasad, *et al.*, "Software Reliability using SPRT: Log Power Model," *International Journal of Current Research and Academic Review*, vol/issue: 2(9), pp. 25-32, 2014. ISSN: 2347-3215.
- [12] K. V. M. Mohan, *et al.*, "Detection of Burr Type XII Reliable Software using Sequential Process Ratio Test," *Indian Journal of Science and Technology*, vol/issue: 8(16), 2015. ISSN: 0974 -5645.
- [13] S. D. Haritha and R. S. Prasad, "A Sequential Probability Ratio Test in Assessing Software Quality Using LPETM," *International Journal of Innovative Technology and Exploring Engineering*, vol/issue: 2(5), pp. 345-348, 2013. ISSN: 2278-3075.
- [14] Reckase M. D., "A Procedure for decision making using tailored testing," in Weiss DJ, editor, "New horizons in testing: Latent trait theory and computerized adaptive testing," New York, Academic Press, pp. 237-54, 1983.
- [15] V. Goutham and R. S. Prasad, "An SPRT Procedure for an Ungrouped Data using MMLE Approach," *IOSR Journal of Computer Engineering (IOSR-JCE)*, vol/issue: 14(6), pp. 37-42, 2013.

- [16] R. S. Prasad, "Half Logistic Software Reliability Growth Model," [PhD Thesis], India, ANU, 2007.
- [17] C. S. Chowdary, *et al.*, "Burr Type III Software Reliability Growth Model," *IOSR-JCE*, vol/issue: 17(1), 2015.
- [18] Pham H., "System Software Reliability," Springer, 2006.
- [19] Lyu M. R., "The Hand book of software Reliability engineering," McGrawHill and IEEE Computer Society Press, 1995. ISBN: 9-07-039400-8.
- [20] Ashoka M., "Data set," Bangalore, Sonata Software Limited, 2010.

BIOGRAPHIES OF AUTHORS



Mrs. Ch. Smitha Chowdary received MCA from Kakatiya University in 2003 and M. Tech., (Computer Science & Engineering) from Acharya Nagarjuna University in 2010. Now she is pursuing Ph.D., in Computer Science & Engineering from Krishna University as Part-Time Research Scholar under the guidance of Dr. R.Satya Prasad and co guide Dr.R.Kiran Kumar. Her research interest lies in Software Reliability Engineering, Data Warehousing and Data Mining



Dr. R Satya Prasad received Ph.D. degree in Computer Science in the Faculty of Engineering in 2007 from Acharya Nagarjuna University, Andhra Pradesh, India. He received gold medal from Acharya Nagarjuna University for his outstanding performance in master's degree. He is currently working as Associate Professor in the department of Computer Science & Engineering, Acharya Nagarjuna University. He performed various academic roles like practical examiner, project adjudicator, external member of board of examiners for various universities and colleges in and around Andhra Pradesh. He received Dr. Abdul Kalam Life Time Achievement Award for his remarkable achievements in the field of Teaching, Research and Publications. His current research is focused on Software engineering, Image processing & Database Management system. He has published 70 research papers in National & International Journals



Dr. K Kiran kumar is a Assistant professor in Department of Computer Science, Krishna University. He is having 12+ experience in Teaching and college Administration. He guided and developed many tools for smooth administration of the college. He is having more than 25 publications in different International journals. His areas of interest are bioinformatics, data mining, software engineering, embedded systems, and network security. He is guiding the PhD scholars, who are doing research from reputed universities.