A Bootstrap Approach to Error-Reduction of Nonlinear Regression Parameters Estimation

[1]Obiora-Ilouno H. O. And   [2]Mbegbu J. I
[1]Department of Statistics Nnamdi Azikiwe University, Awka, Nigeria
[2]Department of Mathematics University of Benin, Benin City Edo State, Nigeria.

Abstract: In this paper we proposed bootstrap algorithm for the estimation of parameters of the non-linear regression analysis by cooperating the Gauss- Newton method. We used bootstrapping to provide estimates of exponential regression coefficients. The computational difficulties that would have encountered in using the proposed method have been resolved by developing a computer program in R which was used to implement the algorithm. From the result obtained the bootstrap algorithm yielded a better reduced error sum of squares $SS\varepsilon$ than the analytical method. With these results, we have a greater confidence in the result obtained.
Keywords: bootstrap, algorithm, exponential, regression, parameters, non-linear, Gauss-Newton.

I.   Introduction

Bootstrap resampling as computer based methods is a tools for constructing inferential procedures in statistical data analysis. The bootstrap approach introduced by Efron (1979) was used initially for estimation of certain statistics such as the mean and the variability of estimated characteristics of the probability distribution of a set of observations by providing confidence intervals for the population parameters (Efron, 2003; Efron and Tibshirani (1998). It is a technique in which sample of size n are obtained randomly with replacement from the original sample each with n^{-1} chances of being selected. This is basically to analyze the population by replacing the unknown distribution function F by the empirical distribution function $\hat{F}$ obtained from the sample. (See Davison and Hinkley, 1997, 2003; Efron, 2003; Bickel and Freedman, 2007; Hall, 2003; Casella, 2003).

Exponential regression is a non-linear regression model often used to measure the growth of a variable, such as population, GDP etc. An exponential relationship between X and Y exists whenever the dependent variable Y changes by a constant percentage, as the variable X also changes.

II.   Materials And Method

Let $Y = f\left(X_1, X_2, \cdots, X_k, \theta_1, \theta_2, \cdots, \theta_j\right) + \varepsilon$ (2.1)

be a non linear regression model with $\theta's$ being the parameters, $X's$ are the predictor variables and the error term $\varepsilon \sim N\left(0, \sigma^2\right)$ independently identically distributed and are uncorrelated.

Equation (2.1) is assumed to be intrinsically nonlinear. Suppose n is number of observations on the Y and $X's$, then

$$Y_i = f\left(X_{i1}, X_{i2}, \cdots, X_{ik}, \theta_1, \theta_2, \cdots, \theta_j\right) + \varepsilon_i \; ; \; i = 1, 2, \cdots, n \qquad (2.2)$$

The n-equation can be written compactly in a matrix notation as

$$Y = f\left(X, \theta\right) + \varepsilon \qquad (2.3)$$

where

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad X = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{k1} \\ X_{12} & X_{22} & \cdots & X_{k2} \\ \vdots & \vdots & \cdots & \vdots \\ X_{1n} & X_{2n} & \cdots & X_{kn} \end{bmatrix}, \quad \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_j \end{bmatrix}, \quad \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

and $E\left(\varepsilon\right) = 0$

The error sum of squares for the nonlinear model is defined as

$$Q = S\left(\varepsilon\right) = \sum_{i=1}^{n}\left\{Y_i - f\left(X_i, \theta\right)\right\}^2 \qquad (2.4)$$

Denoting the least square estimates of θ by $\hat{\theta}$, these minimize the $S(\varepsilon)$. The least square estimates of θ are obtained by differentiating (2.4) with respect to θ, equate to zero and solve for $\hat{\theta}$, this results in J normal equations:

$$\sum_{i=1}^{n}\left\{Y_i - f\left(X_i, \hat{\theta}\right)\right\}\left[\frac{\partial}{\partial\theta_p}f\left(X_i, \theta\right)\right]_{\theta=\hat{\theta}} = 0 \; ; \; i = 1, 2, \cdots, n \, , \; p = 1, 2, \cdots, J \qquad (2.5)$$

In estimating the parameters of nonlinear regression model, we use the Gauss-Newton method based on Taylor's series to approximate equation (2.3). Now, considering the function $f(X, \theta)$ which is the deterministic component of

$$Y_i = (X_i, \theta) + \varepsilon_i \quad i = 1, 2, \cdots, n \qquad (2.6)$$

Let θ^0 be the initial approximate value of θ. Adopting Taylor's series expansion of $f(X_i, \theta)$ about θ^0, we have the linear approximation

$$f(X_i, \theta) = f(X_i, \theta^0) + (\theta - \theta^0)\frac{\partial}{\partial\theta}f(X_i, \theta)\bigg|_{\theta=\theta^0} \qquad (2.7)$$

Substituting expressions (2.7) in (2.6) we obtain

$$Y_i = f(X_i, \theta^0) + \sum_{p=1}^{J}\left[\frac{\partial}{\partial\theta_i}f(X_i, \theta)\right]_{\theta=\theta^0}(\theta_p - \theta_p^0) + \varepsilon_i \quad i = 1, 2, \cdots, n \, , \; p = 1, 2, \cdots, J$$

Equation (2.8) may be viewed as a linear approximation in a neighborhood of the starting value θ^0

Let $\quad f_i^0 = f(X_i, \theta^0)$

$$\beta_p^0 = \theta_p - \theta_p^0$$

$$Z_{pi}^0 = \left[\frac{\partial}{\partial\theta_p}f(X_i, \theta)\right]_{\theta=\theta^0} \, , \; for \; i = 1, 2, \cdots, n \; \; and \; \; p = 1, 2, \cdots, J \qquad (2.8)$$

Hence, equation (2.8) becomes

$$Y_i = f_i^0 + \sum_{p=1}^{J}Z_{pi}^0\beta_p^0 + \varepsilon_i \, , \; i = 1, 2, \cdots, n \qquad (2.9)$$

$$Y_i - f_i^0 = \sum_{p=1}^{J}Z_{pi}^0\beta_p^0 + \varepsilon_i \, , \; i = 1, 2, \cdots, n \qquad (2.10)$$

In a matrix form, we have

$$\begin{bmatrix} Y_1 - f_1^0 \\ Y_2 - f_2^0 \\ \vdots \\ Y_n - f_n^0 \end{bmatrix} = \begin{bmatrix} Z_{11}^0 & Z_{21}^0 & \cdots & Z_{J1}^0 \\ Z_{12}^0 & Z_{22}^0 & \cdots & Z_{J1}^0 \\ \vdots & \vdots & \cdots & \vdots \\ Z_{1n}^0 & Z_{2n}^0 & \cdots & Z_{Jn}^0 \end{bmatrix} \begin{bmatrix} \beta_1^0 \\ \beta_2^0 \\ \vdots \\ \beta_j^0 \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} \qquad (2.11)$$

Compactly, equation (2.11) becomes

$$Y - f^0 = Z^0\beta^0 + \varepsilon \qquad (2.12)$$

where

$$Y - f^0 = \left[Y_1 - f_1^0, \; Y_2 - f_2^0 \cdots Y_n - f_n^0 \right]', \; Z^0 = \begin{bmatrix} Z_{11}^0 & \cdots & Z_{J1}^0 \\ Z_{12}^0 & \cdots & Z_{J2}^0 \\ \vdots & \cdots & \vdots \\ Z_{1n}^0 & \cdots & Z_{Jn}^0 \end{bmatrix} \; \beta^0 = \left(\beta_1^0 \cdots \beta_J^0 \right)', \; \varepsilon = \left(\varepsilon_1, \cdots, \varepsilon_n \right)'$$

We obtain the Sum of squares error $(SS\varepsilon)$

$$SS\varepsilon = (\varepsilon'\varepsilon) = \left(\left(Y - f^0 \right) - Z^0 \beta^0 \right)' \left(\left(Y - f^0 \right) - Z^0 \beta^0 \right)$$

$$\frac{\partial SS\varepsilon}{\partial \beta^0} = -2\left(Y - f^0 \right)' Z^0 + 2\left(Z^0 \hat{\beta}^0 \right)' Z^0 = 0$$

$$\left(Y - f^0 \right)' Z^0 = Z^0 \left(Z^0 \hat{\beta}^0 \right) \tag{2.13}$$

Hence,

$$\hat{\beta}^0 = \left(Y - f^0 \right)' Z^0 \left(Z^{0'} Z^0 \right)^{-1} \tag{2.14}$$

Therefore, the least square estimates of β^0 is

$$\hat{\beta}^0 = \left(Z^{0'} Z^0 \right)^{-1} Z^{0'} \left(Y - f^0 \right) \tag{2.15}$$

Thus, $\hat{\beta}^0 = \left(\hat{\beta}_1^0, \; \hat{\beta}_2^0, \cdots, \hat{\beta}_J^0 \right)'$ minimizes the error sum of squares,

$$S^*(\varepsilon) = \sum_{i=1}^{n} \left(Y_i - f_i^0 - \sum_{p=1}^{J} Z_{pi}^0 \hat{\beta}_p^0 \right)^2 \tag{2.16}$$

Now, the estimates of parameters θ_p of non linear regression (2.1) are

$$\theta_p^1 = \hat{\beta}_p^0 + \theta_p^0 \; ; \; p = 1, 2, \cdots, J \tag{2.17}$$

Iteratively, equation (2.17) reduces to

$$\theta^1 = \hat{\beta}^0 + \theta^0$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$\theta^r = \hat{\beta}^{r-1} + \theta^{r-1}$$

$$\theta^{r+1} = \hat{\beta}^r + \theta^r$$

Thus

$$\theta^{r+1} = \theta^r + \left(Z^{r'} Z^r \right)^{-1} Z^{r'} \left(Y - f^r \right) \tag{2.18}$$

where $\hat{\beta}^r = \left(Z^{r'} Z^r \right)^{-1} Z^{r'} \left(Y - f^r \right)$ are the least squares estimates of β obtained at the

$(r+1)^{th}$ iterations. The iterative process continues until $\left| \dfrac{\theta^{r+1} - \theta^r}{\theta^r} \right| < \delta$

where $\delta = 10^{-5}$ is the error tolerance (See Smith (1998) and Nduka (1999)).

$S^*(\varepsilon)$ is evaluated after each iteration to see if a reduction in its value has actually been achieved. At the end of the $(r+1)^{th}$ iteration, we have

$$S^*(\varepsilon)^r = \sum_{i=1}^{n} \left(Y_i - f_i^r - \sum_{j=1}^{p} Z_{pi}^r \hat{\beta}_p^r \right)^2 \tag{2.19}$$

and iteration is stopped if convergence is achieved. The final estimates of the parameters at the end of the $(r+1)^{th}$ iteration are:

$$\theta^{r+1} = \left(\theta_1^{r+1}, \ \theta_2^{r+1}, \cdots, \theta_j^{r+1} \right)$$

(2.20)

The Bootstrap Algorithm based on the Resampling Observations for the Estimation of Non- linear Regression Parameters

$W_i = \left(Y_i, \ Z_{ji} \right)'$ being the original sample of size n for the resampling, $W_i s$ are drawn independently and identically from a distribution of F. $Y_i = \left(y_1, \ y_2, \cdots, y_n \right)'$ is the column vector of the response variables, $Z_{ji} = \left(z_{j1}, \ z_{j2}, \cdots, z_{jn} \right)'$ is the matrix of dimension $n \times k$ for the predictor variables, where $j = 1, \ 2, \cdots, k \ and \ i = 1, \ 2, \cdots, n$.

Let the $(k \times 1) \times 1$ vector $W_i = \left(Y_i, \ Z_{ji} \right)'$ denote the values associated with i^{th} $\left(w_1, \ w_2, \cdots, w_n \right)$ observation sets.

Step 1: Draw bootstrap sample $\left(w_1^{(b)}, \ w_2^{(b)}, \cdots, w_n^{(b)} \right)$ of size n with replacement from the observation giving n^{-1} probability of each W_i value being sampled from the population and label the elements of each vector $W_i^{(b)} = \left(Y_i^{(b)}, \ X_{ji}^{(b)} \right)'$

where. $j = 1, \ 2, \cdots, k \ and \ i = 1, \ 2, \cdots, n$. From these form the vector for the response variable $Y_i^{(b)} = \left(y_1^{(b)}, \ y_2^{(b)}, \cdots, y_n^{(b)} \right)$ and the matrix of the predictor variables $Z_{ji}^{(b)} = \left(z_{ji}^{(b)}, \ z_{jz}^{(b)}, \cdots, z_{jn}^{(n)} \right)$.

Step2: Calculate the least square estimates for nonlinear regression coefficient from the bootstrap sample; $\hat{\beta}^0 = \left(Z'Z \right)^{-1} Z' \left(Y - f \right)$.

Step3: Compute $\hat{\theta}^1 = \hat{\theta}^0 + \hat{\beta}^0$ using the Gauss-Newton method, the $\hat{\theta}^1$ value is treated as the initial value in the first approximated linear model.

Step4: We return to the second step and again compute $\hat{\beta}'s$. At each iteration, new $\hat{\beta}'s$ represent increments that are added to the estimates from the previous iteration according to step 3 and eventually find $\hat{\theta}^2$, which is $\hat{\theta}^2 = \hat{\theta}^1 + \hat{\beta}^1 \ up \ to \ \theta^{r+1} = \theta^r + \hat{\beta}^r$.

Step 5: Stopping Rule; the iteration process continues until $\left| \dfrac{\theta^{r+1} - \theta^r}{\theta^r} \right| < \delta$, where $\delta = 10^{-5}$, for the values of θ_0, $\theta_1, \cdots, \theta_{p-1}$ from the first bootstrap sample $\hat{\theta}^{(b1)}$.

Step6: Repeat steps 1 to 5 for $r = 1, \ 2, \cdots, B$, where B is the number of repetition.

Step 7: Obtain the probability distributions $F\left(\hat{\theta}^{(b)} \right)$ of bootstrap estimates $\hat{\theta}^{(b1)}$, $\hat{\theta}^{(b21)}, \cdots \hat{\theta}^{(bB)}$ and use $F\left(\hat{\theta}^{(b)} \right)$ to estimate regression coefficients, variances. The bootstrap estimates of regression coefficient is the mean of the distribution $F\left(\hat{\theta}^{(b)} \right)$, (see Topuz and Sahinler,2007; Obiora-Ilouno and Mbegbu, 2012).

$$\hat{\theta}^{(b)} = \frac{\displaystyle\sum_{r=1}^{B} \hat{\theta}^{(br)}}{B} = \bar{\bar{\theta}}^{(br)}$$

(2.21)

The bootstrap standard deviation from $F\left(\hat{\beta}^{(b)} \right)$ distribution is

$$s\left(\hat{\theta}^{(b)}\right) = \left[\frac{\sum_{b=1}^{B}\left[\left(\hat{\theta}^{(dr)} - \hat{\theta}^{(b)}\right)\left(\hat{\theta}^{(dr)} - \hat{\theta}^{(b)}\right)'\right]}{B}\right]^{1} \quad ; \; r = 1, 2, \cdots, B \qquad (2.22)$$

The Computer Program In R For Bootstrapping Non-Linear Regression:

```
#x is the vector of independent variable
#theta is the vector of parameters of the model
#This function calculates the matrix of partial derivatives
F=function(x,theta)
{
output=matrix(0,ncol=2,nrow=length(x))
for(i in 1:length(x)) output[i,]=c(exp(theta[2]*x[i]),theta[1]*x[i]*exp(theta[2]*x[i]))
output
}
#This function calculates the regression coefficients using the Gauss-Newton Method
gaussnewton=function(y,x,initial,tol)
{
theta=initial
count=0
eps=y-(theta[1]*exp(theta[2]*x))
SS=sum(eps**2)
diff=1
while(tol<diff)
{
S=SS
ff=F(x,theta)
theta=c(theta+solve(t(ff)%*%ff)%*%t(ff)%*%eps)
eps=y-(theta[1]*exp(theta[2]*x))
SS=sum(eps**2)
diff=abs(SS-S)
count=count+1
if(count==100) break
pp=c(theta,SS)
#at each iteration
}
pp
}
#This part of the code does the bootstrap
boot=function(data,p,b,initial)
{
n=length(data[,1])
z=matrix(0,ncol=p,nrow=n)
output=matrix(0,ncol=p+1,nrow=b)
for (i in 1:b)
{
u=sample(n,n,replace=T)
for (j in 1:n) z[j,]=data[u[j],]
y=z[,1]
x=z[,2:p]
logreg=gaussnewton(y,x,initial,0.00001)
coef=logreg
output[i,]=c(coef)
}
output
}
```

```
#Then to run the code we use the following
y <- c(data)
x <- c(data)
data=cbind(y,x)
initial=c(initial)
expo=boot(data,p,B,initial)
#Run the following to view the bootstrap results
theta_0=mean(expo[,1])
theta_0
theta_1=mean(expo[,2])
theta_1
SSE=mean(expo[,3])
SSE
```

Problem[GUJARATI AND PORTER, 2009]
The data below relates to the management fees that a leading mutual fund pays to its investment advisors to manage its assets. The fees paid depend on the net asset value of the fund.
 Develop a regression model for the management fees to the advisors.

Fee %	0.520	0.508	0.484	0.46	0.4398	0.4238	0.4115	0.402	0.3944	0.388	0.3825	0.3738
Asset	0.5	5.0	10.0	15	20	25	30	35	40	45	55	60

III.   Results And Discussion

From the data, the higher the net asset values of the fund, the lower are the advisory fees. (see Gujarati and Porter ,2009).

The Analytical Result of R Program (Without Bootstrapping)
```
 y <- c(0.520,0.508,0.484,0.46,0.4398,0.4238,0.4115,0.402,0.3944,0.388,0.3825,0.3738)
x <- c(0.5,5,10,15,20,25,30,35,40,45,55,60)
 run=gaussnewton(y,x,c(0.5028,-0.002),0.00001)
[1]   0.505805756 -0.005487479  0.001934099
[1]   0.508724700 -0.005949598  0.001699222
[1]   0.508897316 -0.005964811  0.001699048
```

 The estimated model from the result is:
$$\hat{Y}_i = 0.50890 X^{-0.00596}$$
where Y and X are the fee and assets respectively.
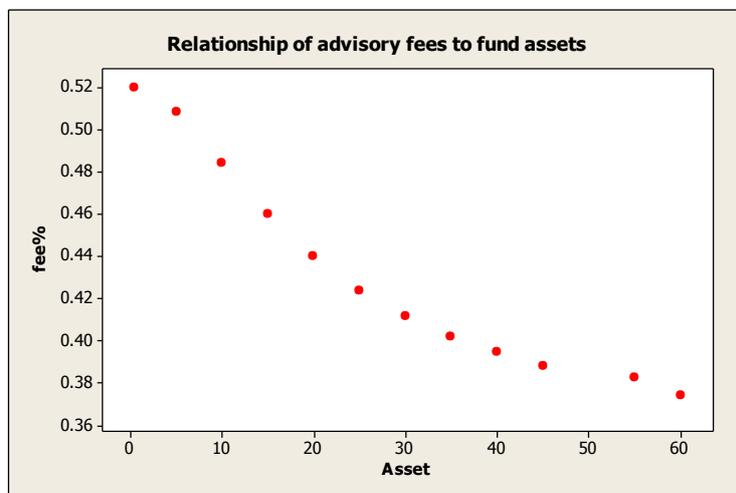
The Result of R Program Using Bootstrapping Algorithm
```
> # run the code use the following for the bootstrap algorithm
> y <- c(0.520,0.508,0.484,0.46,0.4398,0.4238,0.4115,0.402,0.3944,0.388,0.3825,0.3738)
```

```
> x <- c(0.5,5,10,15,20,25,30,35,40,45,55,60)
> data=cbind(y,x)
> initial=c(0.5028,-0.002)
> expo=boot(data,2,1000,initial)
>
> #Run the following to view the bootstrap results
> theta_0=mean(expo[,1])
> theta_0
[1] 0.5075696
> theta_1=mean(expo[,2])
> theta_1
[1] -0.005964939
> SSE=mean(expo[,3])
> SSE
[1] 0.001328289
```

IV.   Discussion

$\theta_0 = 0.50889$ and $\theta_1 = -0.00596$ are the revised parameter estimates at the end of the last iteration. The least squares criterion measure $SS\varepsilon$ for the starting values has been reduced in the first iteration and also further reduced in second, third iterations respectively. The third iteration led to no change in either the estimates of the coefficient or the least squares $SS\varepsilon$ criterion measure. Hence, convergence is achieved, and the iteration end.

The fitted regression function is, $\hat{Y} = 0.50889 \exp\left(-0.00586X\right)$

The results of the analytical and the bootstrap computations are shown in Table 2

Table 2: Analytical and Bootstrap Results

	θ_0	θ_1	$SS\varepsilon$
Analytical	0.50889	-0.00596	0.001699
Bootstrap	0.50757	-0.00596	0.001328

The fitted regression function for both the analytical bootstrapping computation are $\hat{Y} = 0.50889 \exp\left(-0.00596X\right)$ and $\hat{Y} = 0.50757 \exp\left(-0.00596X\right)$ respectively.

V.   Conclusion

From the result of our analysis the bootstrap approach yields approximately the same inference as the analytical method. Also the bootstrap algorithm yielded a better reduced error sum of squares $SS\varepsilon$ than the analytical method (see Table 2). With these results, we have a greater confidence in the result obtained by bootstrap approach.

References

[1].   Casella, G., 2003, Introduction to the Silver Anniversary of the Bootstrap, Statistical Science, Vol.18, No.2, pp. 133-134.
[2].   Efron, B., (1979). Bootstrap Methods: Another look at the Jackknife. The Annuals of Statistics, 7, 1-26.
[3].   Efron, B., 2003, Second Thought on Bootstrapping, Statistical Science, Vol. 18, No. 2,   135-140.
[4].   Efron, B. and Tibshirani, R.J., 1998, An Introduction to the Bootstrap, Library of Congress Cataloging in Publication Data, CRC Press, Boca Raton, Florida, 60-82.
[5].   Freedman, . A., 1981, Bootstrapping Regression Models, Annals of Statistical Statistics,   Vol. 9, 2, 158-167.
[6].   Nduka, E. C., 1999, Principles of Applied Statistics 1, Regression and Correlation Analyses, Crystal Publishers, Imo State, Nigeria.
[7].   Obiora-Ilouno H.O. and Mbegbu J.I (2012), Jackknife Algorithm for the Estimation of Logistic Regression Parameters, Journal of Natural Sciences Research, IISTE UK, Vol.2, No.4, 74-82.
[8].   Sahinler, S. and Topuz, D., 2007, Bootstrap and Jackknife Resampling Algorithm for Estimation of Regression Parameters, Journal of Applied Quantitative Method, Vol. 2, No. 2, pp. 188-199.
[9].   Smith, H. and Draper, N.R., 1998, Applied Regression Analysis, Third Edition, Wiley- Interscience, New York.
[10].   Venables, W. N and Smith, 2007, An Introduction to R, A Programming Environment for Data Analysis and Graphics, Version 2.6.1, pp. 1-100.