

Estimation of the Parameters of a Mixture Weibull Model for Analyze Cure Rate

Nurkaromah Dwidayati ¹⁾, Sri Haryatmi Kartiko ²⁾ and Subanar ²⁾

¹⁾ Department of Mathematics, Gadjah Mada University, Indonesia
Department of Mathematics, Semarang State University, Indonesia
²⁾ Department of Mathematics, Gadjah Mada University, Indonesia

Copyright © 2013 Nurkaromah Dwidayati, Sri Haryatmi Kartiko and Subanar. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

Mixture models have been used to analyze clinical trials with potentially cured patients. Model parameters are estimated via an appropriate EM (Expectation Maximization) algorithm that perform the ML (Maximum Likelihood) in presence of missing data. The basic idea of the EM algorithm is to associate a complete data model to the incomplete structure that is observed in order to simplify the computation of maximum likelihood estimates. We will investigate estimators of the parameters of a mixture Weibull model for analyze cure rate of the breast cancer patient. The problem to estimated of parameters of a mixture Weibull model is solution of derivatives of loglikelihood expectation function is not close form. In this paper we introduce solution this problem with iteration support by Matlab program.

Keywords: mixture Weibull model, cure rate, maximum likelihood, EM algorithm

1. Introduction

In this paper, we study a mixture Weibull model for analyze cure rate with missing data. Model parameters are estimated via an appropriate EM (Expectation Maximization) algorithm that perform the ML (Maximum Likelihood) in presence

of missing data. A two step iterative approach that estimates the parameters of a model starting from an initial guess. Each iteration consists of two steps: (1) an expectation step that finds the distribution for the missing data based on the known values for the observed variables and the current estimate of the parameters; and (2) a maximization step that substitutes the missing data with the expected value.

We will investigate estimators of the parameters of a mixture Weibull model for analyze cure rate of the breast cancer patient. The problem to estimated of parameters of a mixture Weibull model is solution of derivatives of loglikelihood expectation function is not close form. In this paper we introduce solution this problem with iteration support by Matlab program.

2. Mixture Model

Cure models are survival models basically developed to estimate the proportion of patients cured in a clinical trial. These models estimate the cured proportion and also the probability of survival of the uncured patients up to a given point of time. The model developed by Boag [1] was to estimate the proportion of patients cured among those who were receiving treatment for cancer of mouth and throat, cervix, uteri and breast. This model is called the mixture model since it can estimate the proportion of patients cured and the survival function of the uncured patients.

The mixture model is applicable and useful for both continuous and discrete (probability) distributions. This model is based on representation of the (cumulative) probability distribution function or, more simply, the (discrete or continuous) probability density function $f(\cdot)$ from with realizations are desired in terms of mixture. The definition of mixture model [2] is:

Definition 2.1

Let $Y = \{Y_1, \dots, Y_n\}$ denote a random sample of size n , where Y_i is a vector of \mathfrak{R}^q and y_i its realization. $f(y_i)$ its density function. In the mixture model context the density of Y_i is supposed to be a mixture of P parametric densities such that

$$f(y_i; \psi) = \sum_{p=1}^P \pi_p f_i(y_i; \theta_p) \quad (1)$$

with the constraint $\sum_{p=1}^P \pi_p = 1$, P being fixed.

Coefficients π_p can be viewed as the weights of the p^{th} component of the mixture, which is characterized by parameter θ_p and $\psi = (\pi_1, \dots, \pi_{p-1}, \theta_1, \dots, \theta_p)$ denotes the vector of parameters of the model.

In incomplete data problem, mixture model are reformulated since the assignment of the observed data is unknown. We note $X_t = \{Y_t, Z_t\}$ the complete data vector whose only component being observed is Y_t , its density function is then,

$$g(x_t; \psi) = \prod_{p=1}^P [\pi_p f(y_t; \theta_p)]^{z_p} \tag{2}$$

Given a sample of n independent observation from a mixture model defined in (1) the likelihood function is

$$L(y; \psi) = \prod_{i=1}^n \left\{ \sum_{p=1}^P \pi_p f(y_i; \theta_p) \right\} \tag{3}$$

The mixture model is said to be a parametric mixture cure model when standard probability distributions such as exponential, Weibull, Gompertz and generalized F are used. This model can be found in [1], and [3-7]. More general distributions, such as the extended generalized gamma and generalized F, are proposed to reduce their parametric constraints [8]and [9]

The mixture model used without any standard probability distribution is called a nonparametric mixture cure model. The model developed by Kuk and Chen [10] with applied a proportional hazard (PH) assumption to the failure time distribution of uncured patients, and Taylor [11] employed the Kaplan –Meier survivor estimator to estimate the failure time distribution of uncured patients and EM algorithm to estimate parameters in model.

The mixture model is one of several fundamental tools used in constructing efficient exact algorithms for generating repeated realizations of random variables on computer. Its key feature is generates with high probability a realization from a simple distribution that is close to the desired one. Mixture models have been used to analyze clinical trials with potentially cured patients can be seen at Farewell [5], [6], [10],[11-13].

These models are useful when a proportion of study subjects never experience the event of interest. Mixture modeling approach is commonly used to formulate a cure model, which assumes that the underlying population is a mixture of susceptible and nonsusceptible subjects. Mixture model have been used to analyze clinical trials with potentially cured patients.

Mixture model have been used to model the failure time data with the existence of the long-term survivors. The mixture model assumes that a fraction of survivors are cured from the disease of interest. The failure time distribution for uncured individuals (latency) can be modeled by either parametric models or a semi-parametric proportional hazards model.

3. EM Algorithm for Analyze with Missing Data

The EM algorithm was introduced by Dempster, Laird, and Rubin [7]. The EM algorithm is a very general iterative algorithm for maximum likelihood (ML)

estimation in incomplete-data problem. Unknown parameters in this mixture model are estimated via an appropriate EM (Expectation Maximization) algorithm that perform the ML (Maximum Likelihood) in presence of missing data. The basic idea of the EM algorithm is to associate a complete data model to the incomplete structure that is observed in order to simplify the computation of maximum likelihood estimates.

The EM algorithm formalizes for handling missing data: (1) replace missing values by estimated values, (2) estimate parameters, (3) reestimate the missing values assuming the new parameter estimates are correct, (4) reestimate parameters, and so forth, iterating until convergence.

A two step iterative approach that estimates the parameters of a model starting from an initial guess. Each iteration consists of two steps: (1) an expectation step (E-step) that finds the distribution for the missing data based on the known values for the observed variables and the current estimate of the parameters; and (2) a maximization step (M-step) that substitutes the missing data with the expected value

In the incomplete data formulation of mixture models, let us note X the complete data sample space from which x arises, Y the observed sample space and Z the hidden sample space. It follows that $X = Y \times Z$ and $x = (y, z)$. The density of the observed data X can be written:

$$g(x, \psi) = f(y; \psi)k(z | y; \psi) \quad (4)$$

where $f(y; \psi)$ is density of the observed data and $k(z | y; \psi)$ is conditional density of missing observation given the data.

Let $L(y; \psi)$ is the observed/incomplete-data likelihood and $L^c(x; \psi)$ is the unobserved/complete-data likelihood. These likelihoods are linked with the relationship:

$$\log L^c(x; \psi) = \log L(y; \psi) + \log k(z | y; \psi) \quad (5)$$

with

$$\log L^c(x; \psi) = \sum_{i=1}^n \log g(x_i; \psi) \quad (6)$$

and

$$\log k(z | y; \psi) = \sum_{t=1}^n \sum_{p=1}^P z_{tp} \log E\{Z_{tp} | Y_t = y_t\} \quad (7)$$

The hidden variables are not observed, so the EM machinery consists of the indirect optimization of the incomplete-data likelihood via the iterative optimization of the conditional expectation of the complete-data likelihood using the current fit for ψ . We note $\psi^{(h)}$ the value of the parameter at iteration h .

$$\log L(y; \psi) = Q(\psi; \psi^{(h)}) - H(\psi; \psi^{(h)}) \quad (8)$$

with conventions:

$$Q(\psi; \psi^{(h)}) = E_{\psi^{(h)}} \{ \log L^c(X; \psi) | Y \} \quad (9)$$

$$H(\psi; \psi^{(h)}) = E_{\psi^{(h)}} \{ \log k(Z | Y; \psi) | Y \} \quad (10)$$

where $E_{\psi^{(h)}}\{\cdot\}$ denotes the expectation operator, taking the current fit $\psi^{(h)}$ for ψ

The EM algorithm consists of two steps, E-step calculate $Q(\psi; \psi^{(h)})$ and M-step choose $\psi^{(k+1)} = \underset{\psi}{\text{Arg max}}\{Q(\psi; \psi^{(k)})\}$. The E-steps and M-steps are repeated alternatively until the difference $|\psi^{(h+1)} - \psi^{(h)}|$ changes by an arbitrarily small amount.

4. Application on the Lifetimes of Breast Cancer Patients

The data used for the application is lifetime of breast cancer patient from medical record breast cancer patient at Dr. Sardjito Hospital-Yogyakarta, Indonesia. The lifetime were observed from 100 patient who is being treated at the hospital (five year survival). The four-way probability plot for the lifetime of breast cancer patient can see at Figure 4.1

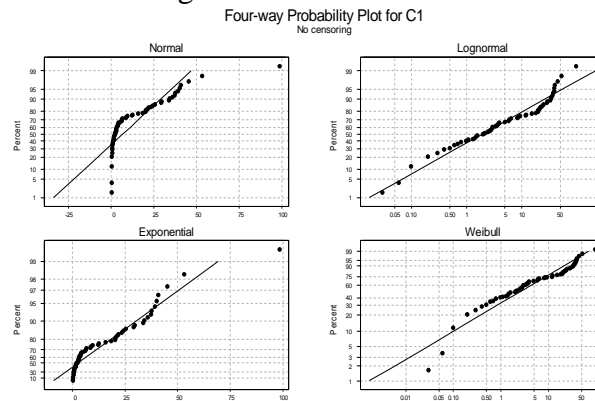


Figure 4.1. Four-way Probability Plot for Lifetime of Breast Cancer Patient

Figure 4.1 given assumption that the lifetime of breast cancer patient have been Weibull distribution. The survival, hazard and Cum-hazard function can be seen at Figure 4.2.

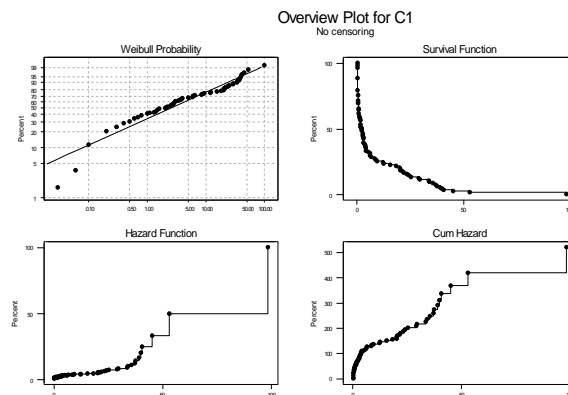


Figure 4.2. Overview Plot for Lifetime of Breast Cancer Patient

The above assumption were used to formulate the *mixture* Weibull model. The density function of T given by:

$$f(t|\psi) = p \frac{\beta_1}{\theta_1} \left(\frac{t}{\theta_1}\right)^{\beta_1-1} \exp\left\{-\left(\frac{t}{\theta_1}\right)^{\beta_1}\right\} + (1-p) \frac{\beta_2}{\theta_2} \left(\frac{t}{\theta_2}\right)^{\beta_2-1} \exp\left\{-\left(\frac{t}{\theta_2}\right)^{\beta_2}\right\},$$

$$t > 0; \theta_1 > 0; \theta_2 > 0; \beta_1 > 0 \text{ and } \beta_2 > 0 \quad (11)$$

with $\psi = (p, \theta_1, \theta_2, \beta_1, \beta_2)$ parameters in model.

Therefore the likelihood function expression for the complete-data is given by:

$$L(\psi | T, Y) = \prod_{i=1}^n \left\{ p \frac{\beta_1}{\theta_1} \left(\frac{t_i}{\theta_1}\right)^{\beta_1-1} \exp\left\{-\left(\frac{t_i}{\theta_1}\right)^{\beta_1}\right\} \right\}^{Y_i} \left\{ (1-p) \frac{\beta_2}{\theta_2} \left(\frac{t_i}{\theta_2}\right)^{\beta_2-1} \exp\left\{-\left(\frac{t_i}{\theta_2}\right)^{\beta_2}\right\} \right\}^{(1-Y_i)}$$

and the corresponding log-likelihood function for the density from the data become:

$$\begin{aligned} \text{Log } L(\psi | T, Y) &= \sum_{i=1}^n Y_i \log p + \sum_{i=1}^n Y_i \log \beta_1 - \sum_{i=1}^n Y_i \log \theta_1 + (\beta_1 - 1) \sum_{i=1}^n Y_i \log t_i \\ &\quad - (\beta_1 - 1) \sum_{i=1}^n Y_i \log \theta_1 - \left(\frac{t_i}{\theta_1}\right)^{\beta_1} \sum_{i=1}^n Y_i + \sum_{i=1}^n (1 - Y_i) \log(1 - p) \\ &\quad + \sum_{i=1}^n (1 - Y_i) \log \beta_2 - \sum_{i=1}^n (1 - Y_i) \log \theta_2 + (\beta_2 - 1) \sum_{i=1}^n (1 - Y_i) \log t_i \\ &\quad - (\beta_2 - 1) \sum_{i=1}^n (1 - Y_i) \log \theta_2 - \left(\frac{t_i}{\theta_2}\right)^{\beta_2} \sum_{i=1}^n (1 - Y_i) \end{aligned} \quad (12)$$

The indicator variable (cure status) as a missing data is:

$$Y_i = \begin{cases} 1; & \text{if } i^{\text{th}} \text{ patient is cured} \\ 0; & \text{if otherwise} \end{cases} \quad (13)$$

where $Y_i \sim \text{BIN}(1, p)$

We will apply the EM algorithm and find the expectation of Y_i . Since the conditional distribution of Y_i given T_i is:

$$Y_i | T_i, \psi^{(k)} \sim \text{BIN}(1, p_i^{(k)})$$

so:

$$\begin{aligned} E(Y_i | T_i, \psi^{(k)}) &= p_i^{(k)} = P[Y_{ip} = 1 | T_i = t_i] \\ &= \frac{p^{(k)} \left(\frac{\beta_1^{(k)}}{\theta_1^{(k)}}\right) \left(\frac{t}{\theta_1^{(k)}}\right)^{\beta_1^{(k)}-1} \exp\left\{-\left(\frac{t}{\theta_1^{(k)}}\right)^{\beta_1^{(k)}}\right\}}{p^{(k)} \left(\frac{\beta_1^{(k)}}{\theta_1^{(k)}}\right) \left(\frac{t}{\theta_1^{(k)}}\right)^{\beta_1^{(k)}-1} \exp\left\{-\left(\frac{t}{\theta_1^{(k)}}\right)^{\beta_1^{(k)}}\right\} + (1-p)^{(k)} \left(\frac{\beta_2^{(k)}}{\theta_2^{(k)}}\right) \left(\frac{t}{\theta_2^{(k)}}\right)^{\beta_2^{(k)}-1} \exp\left\{-\left(\frac{t}{\theta_2^{(k)}}\right)^{\beta_2^{(k)}}\right\}} \end{aligned}$$

where $p^{(k)}$ is a set of known or estimated parameters at k^{th} step, and $p^{(0)}$ is an initial value.

By substituting $p^{(k)}$ for Y_i we obtain the expectation function as

$$\begin{aligned}
 Q(\psi | \psi^{(k)}) = & \sum_{i=1}^n p_i^{(k)} \log p + \sum_{i=1}^n p_i^{(k)} \log \beta_1 - \sum_{i=1}^n p_i^{(k)} \log \theta_1 + (\beta_1 - 1) \sum_{i=1}^n p_i^{(k)} \log t_i \\
 & - (\beta_1 - 1) \sum_{i=1}^n p_i^{(k)} \log \theta_1 - \left(\frac{t_i}{\theta_1} \right) \beta_1 \sum_{i=1}^n p_i^{(k)} + \sum_{i=1}^n (1 - p_i^{(k)}) \log (1 - p) \\
 & + \sum_{i=1}^n (1 - p_i^{(k)}) \log \beta_2 - \sum_{i=1}^n (1 - p_i^{(k)}) \log \theta_2 + (\beta_2 - 1) \sum_{i=1}^n (1 - p_i^{(k)}) \log t_i \\
 & - (\beta_2 - 1) \sum_{i=1}^n (1 - p_i^{(k)}) \log \theta_2 - \left(\frac{t_i}{\theta_2} \right) \beta_2 \sum_{i=1}^n (1 - p_i^{(k)})
 \end{aligned}$$

In the maximization step, setting the first derivatives of $Q(\psi | \psi^{(k)})$ with respect to each parameters equal to zero, but the result is not close form, so we will used iteration to find the solution with Matlab program (see appendix). For the initial value of ψ , after the EM algorithm we will be obtain estimators for unknown parameters in model. The following table shows the results this algorithm.

Table 1. Estimates Using EM Algorithm

Initial value of ψ	Estimator for ψ
$p^{(0)} = 0.75$	$\hat{p} = 0.713$
$\theta_1^{(0)} = 120$	$\hat{\theta}_1 = 1.130$
$\theta_2^{(0)} = 100$	$\hat{\theta}_2 = 1.546$
$\beta_1^{(0)} = 50$	$\hat{\beta}_1 = 0.162$
$\beta_2^{(0)} = 65$	$\hat{\beta}_2 = 0.025$

After the EM algorithm, can be shown that the estimators is convergence, because $\psi^{(k+1)} = \underset{\psi}{\text{Arg max}} \{Q(\psi; \psi^{(k)})\}$.

5. Conclusion

The purpose of the EM algorithm is the iterative computation of maximum likelihood estimators when observation can be viewed as incomplete data. The basic idea of the EM algorithm is to associate a complete data model to the incomplete structure that is observed in order to simplify the computation of maximum likelihood estimates.

The Matlab program which has been developed can be used to find solution of the problem of estimation of parameters of a mixture model for analyzing cure rate. Since it's important to choose appropriate initial value in the EM algorithm, so is needed finding a procedure of choosing initial value for unknown parameters in model.

References

- [1] J.W. Boag. Maximum likelihood estimates of the proportion of patients cured by cancer therapy. *Journal of the Royal Statistical Society*, 11 (1949), 15-53
- [2] F. Picard, F. An introduction to mixture models. *Statistics for Systems Biology Group. Research Report No.7* (2007)
- [3] D.R. Jones, R.L. Powles, D. Machin, and R.J. Sylvester. On estimating the proportion of cured patients in clinical studies. *Biometric-Praximetric*, 21 (1981), 1-11
- [4] V.T. Farewell. The use of mixture models for analysis of survival data with long-term survivors. *Biometric*, 38 (1982), 1041-1046
- [5] V.T. Farewell. Mixture models in survival analysis. Are they worth the risk? *The Canadian Journal of Statistics*, 14 (1986), 257 - 262.
- [6] A.B. Cantor and J.J. Shuster. Parametric Versus Nonparametric Methods for Cure Rates Based on Censored Survival Data. *Statistics in Medicine*, 11(1992), 931-937
- [7] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum Likelihood from incomplete data via the EM algorithm (with discussion). *Journal of Royal Statistical Society, Series B* (1977), 39.1-38
- [8] K. Yamaguchi. Accelerated failure-time regression models with a regression model of surviving fraction. An application to the analysis of 'permanent employment' in Japan. *Journal of the American Statistical Association*, 37 (1992), 284-292
- [9] Y. Peng and K.G.B. Dear. 2000. A Nonparametric Mixture Model for Cure Rate Estimation. *Biometric*, 56 (2000), 237- 243
- [10] A.Y. Kuk and C. Chen. A mixture model combining logistic regression and life model. *Biometrika*, 79 (1992), 531-541.
- [11] J.M.G. Taylor. Semi-parametric estimation in failure time mixture models. *Biometric*, 51 (1995), 899-907
- [12] R.A. Maller and S. Zhou. 1992. Estimating the proportion of immunes in a censored sample. *Biometrika*, 79 (1992), 731-739.
- [13] J.P. Sy and J.M.G. Taylor, J.M.G. Estimation in a Cox Proportional Hazards Cure Model, *Biometric*, 56 (2000), 227-236.

Appendix

```

dataq=load(handles.pathfile);
dataq=dataq.';
findnegative = datapositive(dataq);
if findnegative==0
    p0 = str2num(get(handles.txtp, 'string'));
    t10 = str2num(get(handles.txtt1, 'string'));
    t20 = str2num(get(handles.txtt2, 'string'));
    b10 = str2num(get(handles.txtb1, 'string'));
    b20 = str2num(get(handles.txtb2, 'string'));
    epsilon = str2num(get(handles.txte, 'string'));
    if isempty(p0) || isempty(t10) || isempty(t20) || isempty(b10)
        || isempty(b20) || isempty(epsilon)
            else

                y = [p0,t10,t20,b10,b20];
                savefile='result.mat';
                ts1=[];
                ts2=[];
                ts3=[];
                ts4=[];
                result={ts1,ts2,ts3,ts4};
                save(savefile, 'result');
                savefile='Output.mat';
                p=[];
                teta1=[];
                teta2=[];
                beta1=[];
                beta2=[];
                galat=[];
                iterasi=[];
                Output={p,teta1,teta2,beta1,beta2,galat,iterasi};
                save(savefile, 'Output');
                n=length(dataq);
                jomt1=0;
                jomt2=0;
                jomt3=0;
                jomt4=0;
                jomt5=0;
                iterasi=0;
                difference=epsilon;
                deviation = epsilon;
                procesmessage;
                syms x;
                while(difference>= epsilon) & (y(4)>0) & (y(5)>0) &
(y(1)>0) & (y(1)<1)
                    clc;
                    tStart=tic;
                    plasting=y(1);
                    tetallasting=y(2);
                    teta2lasting=y(3);
                    betallasting=y(4);

```

```

        beta2lasting=y(5);
        Ep
        =(y(1).*wblpdf(dataq,y(2),y(4))./((y(1).*wblpdf(dataq, y(2),
        y(4)))+(1-y(1)).*wblpdf(dataq, y(3),y(5))));
        p= mean(Ep);
        syms b1 b2 x1 x2;
        for i=1:n
            jumt1=jumt1+dataq(i).^(y(4).*Ep(i));

jumt2=jumt2+((dataq(i).^y(5))./(dataq(i).^y(5).*Ep(i)));
            jumt3=jumt3+(Ep(i).*log(dataq(i)));
            jumt4=jumt4+log(dataq(i));
            jumt5=jumt5+(Ep(i).*log(dataq(i)));
            t1(i)=dataq(i).^Ep(i);
            t2(i)= Ep(i).*log(dataq(i));
            t3(i)=dataq(i).^(1-Ep(i));
            t4(i)=(1-Ep(i)).*log(dataq(i));
            ts1 =
strcat('(',num2str(t1(1,i)),'^x)*((',num2str(t2(1,i)),'')+');
            ts2 = strcat('(',num2str(t1(1,i)),'^x')+');
            ts3 =
strcat('((' ,num2str(t3(1,i)),'^x)*((',num2str(t4(1,i)),''))+');
            ts4 = strcat('(',num2str(t3(1,i)),'^x')+');

            st=sum(t2);
            savefile='result.mat'; %name of data file data
            [v,w]=size(result);
            if isempty(result{1,1}) && isempty(result{1,2}) &&
            isempty(result{1,3}) && isempty(result{1,4})
                result(1,:)= {ts1,ts2,ts3,ts4}; %add to new
record
            else
                result(v+1,:)= {ts1,ts2,ts3,ts4}; % add to new
record
            end
            save(savefile,'result');
        end;
        load('result.mat');
        gab1=cell2mat(result(:,1).');
        gab2=cell2mat(result(:,2).');
        gab3=cell2mat(result(:,3).');
        gab4=cell2mat(result(:,4).');
        eq1=strcat('(',num2str(sum(Ep)),'/x)-
        (',num2str(sum(Ep)*log(y(2))),')+((',num2str(st),')-
        ((' ,gab1,'0)/(',num2str(y(2)),'^(x*',num2str(sum(Ep)),')))+((',ga
        b2,'0)*',num2str(sum(Ep)*log(y(2))),')/(',num2str(y(2)),'^(x*',num
        2str(sum(Ep)),'))');
        eq2=strcat('(',num2str(n-sum(Ep)),'/x)-(',num2str((n-
        sum(Ep))*log(y(3))),')+((',num2str(jumt4),')-(',num2str(jumt5),')-
        ((' ,num2str(y(3)),')^((-x)*((',num2str(n-
        sum(Ep)),')))*((',gab3,'0))+((' ,gab4,'0)*',num2str((n-
        sum(Ep))*log(y(3))),')*((' ,num2str(y(3)),')^((-x)*((',num2str(n-
        sum(Ep)),'))));
        teta1 = exp(log(jumt1)./(y(4).*sum(Ep)));
        teta2 = exp(log(jumt2)./(y(5).*(-sum(Ep)+n)));

```

```

% ===== Initial value of beta 1 FSOLVE
=====

opts=optimset('Display','off','Diagnostics','off','Display','off',
'TolX',0.00001,'TolFun',0.0001);
for i=-5:5
    [h1,j,k]= fsolve(eq1,i,opts);
    if k==1
        beta1=h1;
    end
end
for i=-5:5
    [h2,j,k]= fsolve(eq2,i,opts);
    if k==1
        beta2=h2;
    end
end

% ===== Estimators FSOLVE =====

    deviation(iterasi+1) = sum(abs(plasting - p)) +
sum(abs(tetallasting - tetal)) + sum(abs(teta2lasting - teta2))+
sum(abs(betallasting - beta1)) + sum(abs(beta2lasting - beta2));
    difference= sum(abs(plasting - p)) +
sum(abs(tetallasting - tetal)) + sum(abs(teta2lasting - teta2))+
sum(abs(betallasting - beta1)) + sum(abs(beta2lasting - beta2));
    tElapsed = toc(tStart);
    iteration=iteration+1;
    y=[p, tetal, teta2, beta1, beta2];

    savefile='Output.mat'; %name of data file
    [v,w]=size(Output);
    if isempty(Output{1,1}) && isempty(Output{1,2}) &&
isempty(Output{1,3}) && isempty(Output{1,4}) &&
isempty(Output{1,5}) && isempty(Output{1,6}) &&
isempty(Output{1,7})

Output(1,:)={p,tetal,teta2,beta1,beta2,selisih,iterasi}; % add to
new record
    else

Output(v+1,:)={p,tetal,teta2,beta1,beta2,difference,iteration}; %
add to new record
    end
        save(savefile,'Output');
    end

    close (procesmessage);
    %show to result
load 'Output.mat';
[v,w]=size(Output);
if v==1
    p=Output{1,1};

```

```

        teta1=Output{1,2};
        teta2=Output{1,3};
        beta1=Output{1,4};
        beta2=Output{1,5};
        set(handles.txtpest, 'string', p);
        set(handles.txttlest, 'string', teta1);
        set(handles.txtt2est, 'string', teta2);
        set(handles.txtblest, 'string', beta1);
        set(handles.txtb2est, 'string', beta2);
        drawnow;
    else
        if (Output{v,1} > 1) || (Output{v,1} > 1) || (Output{v,2}
< 0) || (Output{v,3} < 0) || (Output{v,4} < 0) || (Output{v,5} <
0) || isnan(Output{v,1}) || isnan(Output{v,2}) ||
isnan(Output{v,3}) || isnan(Output{v,4}) || isnan(Output{v,5}) ||
(Output{v,4} > 100) || (Output{v,4} > 100)
            n=v-1;
            for i=1:n
                p=Output{i,1};
                teta1=Output{i,2};
                teta2=Output{i,3};
                beta1=Output{i,4};
                beta2=Output{i,5};
                set(handles.txtpest, 'string', p);
                set(handles.txttlest, 'string', teta1);
                set(handles.txtt2est, 'string', teta2);
                set(handles.txtblest, 'string', beta1);
                set(handles.txtb2est, 'string', beta2);
                drawnow;
            end
        else
            n=v;
            for i=1:n
                p=Output{i,1};
                teta1=Output{i,2};
                teta2=Output{i,3};
                beta1=Output{i,4};
                beta2=Output{i,5};
                difference=Output{i,6};
                set(handles.txtpest, 'string', p);
                set(handles.txttlest, 'string', teta1);
                set(handles.txtt2est, 'string', teta2);
                set(handles.txtblest, 'string', beta1);
                set(handles.txtb2est, 'string', beta2);
                drawnow;
            end
        end
    end
end
else

```

Received: August 14, 2013