Journal of Theoretical and Applied Information Technology
31st January 2019. Vol.97. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CALIBRATING THE NELSON-SIEGEL MODEL CLASSES AND THEIR ESTIMATION USING HYBRID-GENETIC ALGORITHM APPROACH: CASE STUDY OF INDONESIAN GOVERNMENT BONDS

[1]MUSLIM, [2]DEDI ROSADI, [3]GUNARDI, [4]ABDURAKHMAN

[1]Math Education, Jambi University, Jambi, Indonesia

[2,3,4]Mathematics, Gadjah Mada University, Yogyakarta, Indonesia

E-mail: [1]aliem_h@yahoo.co.id, dedirosadi@gadjahmada.edu

ABSTRACT

In this paper, we consider the problem of modelling the yield curve using Nelson-Siegel model classes. Nelson-Siegel model classes discussed here are NS model, BL model, NSS model, RF model, and our proposed NSSE models. NSSE model is a model which extends the standard NS model as Nelson-Siegel model class by adding some linear and non-linear parameters in which form the fourth hump of the model class. The purpose of adding the hump is to accommodate the possibility of having the following cases: the first, the condition when the short term and the medium term yields are higher than the long term yield. The second, the condition when the upper-value short term yields are higher than both the short term yields on average and the long term yields. The third, the case when the upper-value medium term yields are higher than both the medium term yields on average and the long term yields. These considered cases make the yield curve more likely to have minimum locals and therefore, the Nelson-Siegel model classes become more difficult to be estimated. To overcome this problem, in this paper we estimate the model using the hybrid-genetic algorithm approach and compare it with the standard estimation based on NLS method. We provide an empirical study using Indonesian Government-Bond Yield Curve (IGYC) data, and found that the best model for IGYC is 6-factors model.

Keywords: *Yield Curve, Nelson-Siegel Model, Hybrid Method, Genetic Algorithm, Nonlinear Least Square, and Constrained Optimization*

1. INTRODUCTION

Yield curve describes how much yields are obtained against time to maturity. It can be determined using various approaches, i.e., parametric and nonparametric methods (Stander [22]). In this paper, we consider a parametric model class called as Nelson-Siegel model. Nelson-Siegel (NS) model is NS model introduced by Charles R. Nelson and Andrew F. Siegel (Nelson-Siegel, [3]). The three factors are flat, hump, and *S* shaped curves. The NS model is extended in Svenson [15] by adding the second hump and it is known as Nelson-Siegel-Svensson or four factors model. Further extensions are available. In Diebold et.al. [8], it has been shown the connection between the dynamic factors and latent factor in NS model and resulting no-arbitrage model. Diebold et.al. [7] have studied NS model by comparing the yields in every country against the factors that influence the yields. After that, Pooter [17] discuss the extension of no-arbitrage NS model for the purpose of forecasting the interest rate. Christensen et.al. [13] have studied and compared dynamic NS model, no-arbitrage NS model, and the extended NS model. Rezende and Ferreira [21] proposed RF model by adding the second and the third hump into the model. In this paper, we propose the extension of Nelson-Siegel model class by adding the fourth hump into the RF, hence this model becomes NSSE model. The purpose of adding the hump is to accommodate the possibility of having the following cases in practical application: the first, the condition when the short term and the medium term yields are higher than the long term yield. The second, the condition when the upper-value short term yields are higher than both the short term yields on average and the long term yields. The third, the case when the upper-value medium term yields are higher than both the medium term yields on average and the long term yields.

In the above we mention various extensions of the standard NS model, that is NSS, 5-factors, and NSSE model. These extensions are more difficult to estimate due to the existence of multiple local minimum (maximum) values. To estimate the parameters, various approaches are available in the literature. Bolder and Streliski [4] considered full model estimation algorithm based on Sequential Quadratic Programming (SQP) approach and partial estimation by separating the linear and non linear parameters. Annaert et.al. [12] estimated the model classes by considering the multicollinearity problem and grid search method based OLS. Diebold et.al. [8] estimated the model class using Linear Least Square (LLS) with Kalman filter approach, while Landschoot [2], Diebold et.al. [10] consider the maximum likelihood method. Krippner [16], Maria, et.al. [1], Gilli et.al [18], Rezende and Ferreira [21], proposed the estimate of the model classes by Least Square methods. An example of implementation of estimation using R is provided, for instance, in Rosadi [5].

In the classic estimation methods mentioned above, it is necessary to specify the initial values of every parameters, in which their optimal starting points are in general different for each different data. In this paper we consider more flexible method based on genetic algorithm approach, which combines Nonlinear Least Square (NLS) estimation and constrained optimization and it is not required to specify the starting values of the parameters. More detail discussions about NLS, constrained optimization genetic algorithm can be found in for instance Björck [23], Sun and Yuan [25], Gimeno and the Nave [20], or Mitchell [19].

The rest of this paper is organized as follows. In the next section we describe NS model class and introduce our proposed 6-factors model. In section 3, we outline the estimation method which is the hybrid-genetic algorithm approach, where in section 4, we provide empirical studies using Indonesia Government Bond Data. In this part we also compare the performance of our estimation approach with the estimate based on the standard NLS method. Section 5 concludes.

2. NELSON-SIEGEL MODEL CLASS

Before we discuss the NS class model, we shall be given the basic concept of yield curve model. Let $P(t,T)$ be the price of zero coupon bond at time t with maturity time T, often referred to as discount function and $R(t,T)$ be the value of spot rate (i.e. the continuously compounded zero-coupon nominal yield to maturity). The discount curve can be obtained from the yield curve using the relation $P(t,T) = exp[-R(t,T)(T-t)]$, thus we obtain $R(t,T) = -\frac{\ln[P(t,T)]}{T-t}$. Let $f(t,T,T+\Delta t)$ be the forward rate of a contract specified at time t with reporting period T and time to maturity $T+\Delta t$. The instantaneous forward rate function $y(t,T,T+\Delta t)_{inst}$ defined as the value of forward rate with $\Delta t \to 0$, and we denote it as $f(t,T)$. The relationship between price of zero coupon bond with instantaneous forward rate be given by equation $P(t,T) = \exp(-\int_t^T y(t,u)du)$ or $y(t,T) = -\frac{d}{dt}\ln P(t,T)$ so that we obtain relationship between yield to maturity value and forward rate as $R(t,T) = \int_{x=t}^{T} y(t,u)du/(T-t)$. Thus, the value of zero coupon yields can be expressed as form the weighted average of forward rate. In a similar way, the value of coupon bond can be expressed as the sum of present value of all future cash flow (in the form of coupon and principal value payments), if the yield curve or forward rate curve is known (Stander [22]) .

One of the popular forward rate models is NS class model. The first model considered in the literature is Nelson-Siegel/NS model (Nelson and Siegel, [3]). In the NS model there are three factors to form the yield curve, namely flat, a slope, and hump, which comes the name 3-factors model. This model is formulated as

$$f_t(\lambda;\, \boldsymbol{\beta}, \tau) = \beta_0 + \beta_1\, exp\left(-\frac{\lambda}{\tau}\right) + \beta_2 \frac{\lambda}{\tau} \exp\left(-\frac{\lambda}{\tau}\right), \quad (1)$$

where f_t denote the yield of bonds, $\lambda = T - t$ is time to maturity, $\boldsymbol{\beta}$ is linear parameter vector, τ is nonlinear parameter thatdetermines the position of the hump,β_0 represents the long-run level of interest rate, β_1 the short-run component, and β_2 the medium-term component. If the time to maturity goes to infinity, the forward rate converges to β_0. If the time to maturity goes to zero, the forward rate converges to $\beta_0 + \beta_1$. To avoid negative forward rates, β_0 and $\beta_0 + \beta_1$ should be positive,β_0 can be interpreted as the long-run interest rate and $\beta_0 + \beta_1$ as the instantaneous forward rate. This implies that $-\beta_1$ can be interpreted as the slope of the yield curve. The curve will have a negative slope if β_1 is positive and vice versa,β_1 also indicates the speed with which the curve evolves towards its long-run

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

trend,β_2 determines the magnitude and the direction of the hump or through in the yield curve. The parameter τ is a time constant that should be positive in order to assure convergence to the long-term value β_0. This parameter specifies the position of the hump of the yield curve (Nelson-Siegel, [3]). This model can be illustrated as figure 1.

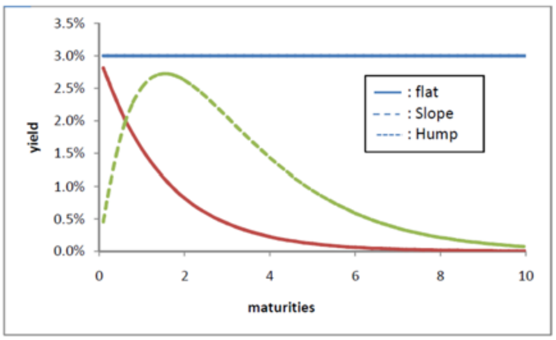

Figure 1: Yield Curve of NS model with $\tau = 1.54$

Bliss [27] proposes extension of NS model with distinguishing slope and hump parameters. This model is formed as follows

$$f_t(\lambda;\ \boldsymbol{\beta}, \tau) = \beta_0 + \beta_1\, exp\left(-\frac{\lambda}{\tau_1}\right) +$$

$$\beta_2 \frac{\lambda}{\tau_2} \exp\left(-\frac{\lambda}{\tau_2}\right), \qquad (2)$$

τ_1 determine the form slope, if τ_1 toward infinite than slope decrease and if τ_1 toward zero than slope form *U*. τ_2 determine hump position, if τ_2 toward zero than hump in short term and if τ_2 toward infinite than hump in middle term. This model can be described as follows

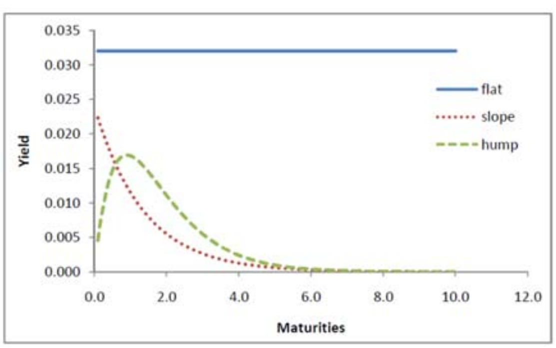

Figure 2: Yield Curve of Bliss model with $\tau_1 = 1.07$ *and* $\tau_2 = 1.17$.

Svensson [15] has expanded the NS model by adding the second hump such that it becomes NSS or Nelson-Siegel-Svensson/NSS model. The NSS model has the following form

$$f_t(\lambda;\ \dot{\boldsymbol{\beta}}, \boldsymbol{\tau}) = \beta_0 + \beta_1 \exp\left(-\frac{\lambda}{\tau_1}\right) + \beta_2\left[\frac{\lambda}{\tau_1}\exp\left(-\frac{\lambda}{\tau_1}\right)\right]$$

$$+\beta_3\left[\frac{\lambda}{\tau_2}\exp\left(-\frac{\lambda}{\tau_2}\right)\right], \qquad (3)$$

where $\dot{\boldsymbol{\beta}}$ is linear parameter vector that $\dot{\boldsymbol{\beta}} = (\beta_0,\ \beta_1,\ \beta_2,\ \beta_3)'$, $\boldsymbol{\tau} = (\tau_1, \tau_2)'$ nonlinear parameter that determines the position of the first and second hump and the other parameters are the same as NS model, β_0 must be positive, it is the asymptotic value of $f_t(\lambda;\ \dot{\boldsymbol{\beta}}, \boldsymbol{\tau})$. The curve will tend towards the asymptote as the λ approaches infinity,β_1 determines the starting (short-term) value of the curve in terms of deviation from the asymptote. It also defines the basic speed with which the curve tends toward its long-term trend. The curve will have a negative slope if this parameter is positive and vice versa. Note that the sum of β_0 and $\beta_1$is the vertical intercept,τ_1 must be positive, specifies the position of the first hump or U-shape on the curve,β_2 determines the magnitude and direction of the hump. If β_2 is positive, a hump will occur at τ_1 whereas, if β_2 negative, a *U*-shaped value will occur at τ_1,τ_2 must also be positive, specifies the position of the second hump or U-shape on the curve. And β_3 analogous to β_2, determines the magnitude and direction of the second hump (Bolder and Streliski [4]). This curve model can be illustrated as Figure 3.

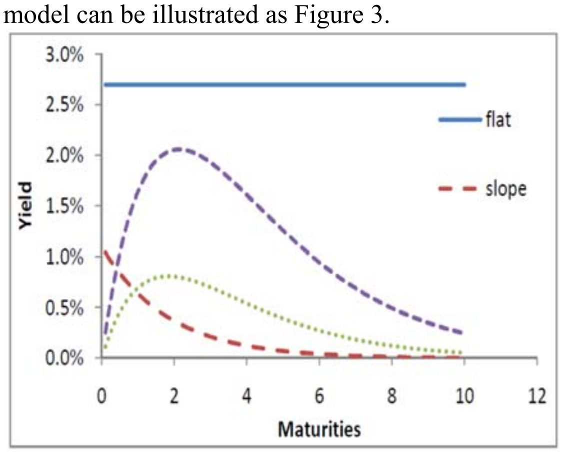

Figure 3: Yield Curve Component of NSS model with $\tau_1 = 1.68$ *and* $\tau_2 = 3.01$.

Rezende and Ferreira [21] have added the second slope into NSS model such that it becomes 5-factors model, as follows:

$$f_t(\lambda;\ \ddot{\boldsymbol{\beta}}, \ddot{\boldsymbol{\tau}}) = \beta_0 + \beta_1 \exp\left(-\frac{\lambda}{\tau_1}\right) + \beta_2\left[\frac{\lambda}{\tau_1}\exp\left(-\frac{\lambda}{\tau_1}\right)\right] + \beta_3\left[\frac{\lambda}{\tau_2}\exp\left(-\frac{\lambda}{\tau_2}\right)\right] + \beta_4\left[\frac{\lambda}{\tau_3}\exp\left(-\frac{\lambda}{\tau_3}\right)\right] + \beta_5\left[\frac{\lambda}{\tau_4}\exp\left(-\frac{\lambda}{\tau_4}\right)\right], \quad (4)$$

where f_t is shape of the curve the instantaneous forward rate in time *t*, λ is time to maturity, $\ddot{\boldsymbol{\beta}}$ is linear parameter that $\ddot{\boldsymbol{\beta}} = (\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5)'$, $\ddot{\boldsymbol{\tau}}$ is nonlinear parameter that $\ddot{\boldsymbol{\tau}} = (\tau_1, \tau_2, \tau_3, \tau_4)'$. In this model, β_0 is constant value of forward rate function and it will always be constant if the maturity is close to zero. β_1 determines the beginning of the curve (short term) in various forms of deviation where the curve will be negatively skewed if β_1 is positive and vice versa. β_2 determines the first hump where the curve will be positively skewed if β_2 is positive and negatively skewed if β_2 is negative. β_3 determines the second hump where the curve will be positively skewed if β_3 is positive and in *S* shape if β_3 is negative. β_4 determines the third hump where the curve will be positively skewed if β_4 is positive and in *S* shape if β_4 is negative. β_5 determines the shape of the fourth hump where the curve will be positively skewed if β_5 is positive and *S* if $\beta_5$is negative. τ_1, τ_2, τ_3 and τ_4 determine the position of first hump, the second hump, the third hump and the fourth hump, respectively. This model can be illustrated as follows

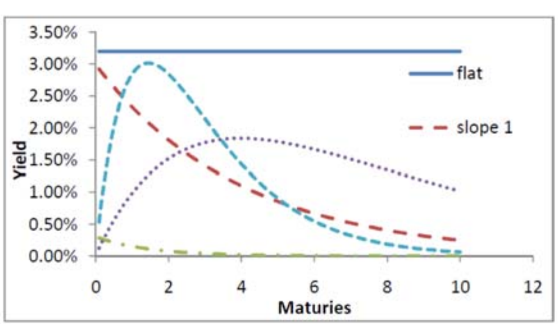

Figure 4: Yield Curve Component of RF Model with $\tau_1 = 3.98$*and* $\tau_2 = 1.36$.

In this paper, we propose NSSE model, by adding the first and the second hump into the third factor of NSS model, and this model can be defined as follows:

$$f_t(\lambda;\ \dot{\boldsymbol{\beta}}, \boldsymbol{\tau}) = \beta_0 + \beta_1\, exp\left(-\frac{\lambda}{\tau_1}\right) + \beta_2\left[\frac{\lambda}{\tau_1}\exp\left(-\frac{\lambda}{\tau_1}\right) + \frac{\lambda}{\tau_2}\exp\left(-\frac{\lambda}{\tau_2}\right)\right] + \beta_3\left[\frac{\lambda}{\tau_2}\exp\left(-\frac{\lambda}{\tau_2}\right)\right] \quad (5)$$

where y is yield of bond, λ is time to maturity, $\dot{\boldsymbol{\beta}}$ is linear parameter, $\boldsymbol{\tau}$ is nonlinear parameter. In this model, β_0 is constant value of forward rate function and it will always be constant if the maturity is close to zero,β_1 determines the beginning of the curve (short term) in various forms of deviation where the curve will be negatively skewed if β_1 is positive and vice versa,β_2 determines the first hump where the curve will be positively skewed if β_2 is positive and negatively skewed if β_2 is negative,β_3 determines the second hump where the curve will be positively skewed if β_3 is positive and in *U* shape if β_3 is negative. τ_1 determines slope position and the first hump, if τ_1 toward zero than slope and the first hump will approach zero and τ_1 toward infinite than slope will negatively skewed while the first hump will be located in medium term, τ_2 toward zero than the second hump will approach zero and the first hump will form *S* curve and if τ_2 toward infinite than the first hump will be in medium term and the second hump will be positively skewed. The NSSE model is illustrated in Figure 5 below:

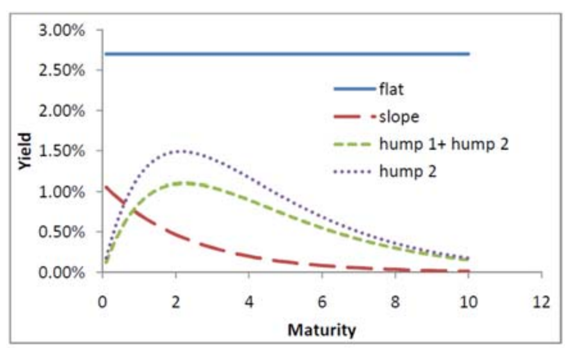

Figure 5:Yield Curve Components of the NSSE model with $\tau_1 = 2.35$ *and* $\tau_2 = 2.14$

From Figure 5 we can see the effect of each factor of NSSE. The flat curve is the first factor. The slope is the second factor, this curve

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

will be negatively skewed if β_1 positive and vice versa. Combination of the first and the second hump is the third factor of the NSSE model, and the fourth factor, the second hump, is the factor proposed in Svensson [15].

3. ESTIMATION AND OPTIMIZATION

To determines the parameters in NS, BL, NSS, RF and NSSE models as equation (1), (2), (3), (4), and (5). These models estimate using nonlinear least square by GA approach, in the estimation, we obtain the sum of square error (SSE) with it have conditional based on the parameters in the model class, the SSE conditional resolved optimization. We discuss estimation and optimization as follows.

Estimation Method

In section previous, we discussed that Nelson-Siegel class model has four models and adding with one model extended. The model has parameters and constraints different. Respectively, can be explained that NS model has the four parameters and three constraints, BL model has five parameters and four constraint, NSS model have the six parameters and four constraints, while RF model have the eight parameters and five constraints, and NSSE model have the six parameters and four constraints. In this section, we only discuss the estimate of NSSE model. Let

$$y(\theta) = f(\theta) + \varepsilon, \quad (6)$$

where θ is the parameters of models and ε is residual,

and let

$$g_j(\theta) > 0, j = 1,2,\cdots, \quad (7)$$

where g is the constrain of models, so estimation NLS of equation (6) subject to (7) is

$$\psi(\theta) = min\frac{1}{2}\sum_{i=1}^{p}\left(y_i(\theta) - f(\theta)\right)^2, \quad (8)$$

subject to

$$g_j(\theta) > 0$$

where ψ is function of the sum square error, and θ is parameters of model. To minimize equation (8), we do optimization as follows.

To minimize equation (8), we use optimization with inequality constraint arediscussedas follows.

3.2 Optimization Method

To minimize equation (8), we use inequality constrain optimization, discussion ofthisoptimization in Chong and Zak [26]. The optimization is performed by transforming the inequality constraint become equality constraint. This problem discussed Rao [24]. Constraint in equation (8) can be transformed by adding slack variable into constraints, then the constraints became;

$$g_j(\theta) - s_j^2 = 0, \quad (9)$$

for $s_j \geq 0$. Thus, equation (6) can be optimized subject to (9) as follows;

$$\mathcal{L}(\theta,\mu,s) = \frac{1}{2}\sum_{i=1}^{p}\left(y_i(\theta) - f(\theta)\right)^2 - \sum_{j=1}^{n}\mu_j\left(g_j(\theta) - s_j^2\right), \quad (10)$$

where $\mathcal{L}$ is Lagrange function, μis Lagrange constant with $\mu \leq 0$, and s is slack variable. Equation (10) is minimized on the parameters, Lagrange constant and slack variable. To minimize equation (10), we use genetics algorithm approach. This algorithm satisfy the condition: $\mu \leq 0,$; $s \geq 0,$; $\partial\mathcal{L}(\theta,\mu,s)/\partial\theta = 0'$; $\partial\mathcal{L}(\theta,\mu,s)/\partial\mu = 0'$; and $\partial\mathcal{L}(\theta,\mu,s)/\partial s = 0'$.

The stages of this approach are discussed in Chong and Zak [26] as follows.

1. Initializing the population

In this step, we optimize Equation (6) for the four models; they are 3-factors, 4-factors, 5-factors, and 6-factors models. These models have 3 Lagrange constants and 3 slack variables, 4 Lagrange constants and 4 slack variables, 5 Lagrange constants and 5 slack variables, as well as 6 Lagrange constants and 6 slack variables, respectively. We obtain the total number of parameters to be optimized, including Lagrange constants and slack variables of each model are 10, 14, 18, and 22, respectively. For each parameter, we encode it into 20 bits of real number [0,1] which are chosen randomly, and all parameters together form one chromosome. We generate 50 chromosomes for optimization here.

2. Convert of the chromosome code to real number

For each chromosome, we convert the [0,1] encoded values of each parameter into real number, which is called the fitness value, using the following formula:

$$fitness\ value = b_b + (b_a - b_b) \quad (11)$$

where b_a and b_b denote the lower and the upper bound of the interval of parameter, c = the value of bits, k is the position of bits. The interval of parameters used here are [-20, 20] for linear parameters, [0, 15] for nonlinear parameters and [-1,1] for Lagrange constants and slack variables.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

From this step we obtain the fitness value of every parameter in each chromosome.

3. Individual Evaluation
For every chromosome, we calculate the value of the following function

$$f_{eval} = \frac{1}{\mathcal{L}(\theta,\mu) + d}, \quad (12)$$

where d is an arbitrary small number and $\mathcal{L}(\theta,\mu)$ is obtained from (6). Based on their values from this step, we rank the fitness values from the biggest to the smallest.

4. Elitism
If the fitness values and their ranks have been obtained, in this stage, we select two chromosomes with the smallest rank and saving the chromosomes.

5. Linear Fitness Ranking (LFR);
In this step, we scale the fitness values obtained from individual evaluation (step 3). The purpose is to avoid the convergence tendency in local optima by obtaining the new fitness values that have greater variance. The formula is presented below:

$$g(j) = g_{maks} - (g_{maks} - g_{min})\left(\frac{R(j)-1}{N-1}\right), (13)$$

$g(j)$ denotes the new fitness value of chromosome j, g_{maks} is the maximum fitness value from step 2, g_{min} is the minimum fitness value from step 2, $R(j)$ is the chromosome rank and N is number of chromosomes (N=50 in this case).
For each chromosome, we calculate the value of relative fitness value (i.e. $g_{rel}(j) = g(j)/\sum_{i=1}^{50} g(i)$ and its cumulative values).

7. Selection
From step 4, we obtain two chromosomes (first pair of the chromosomes) to be the parent. We select the rest of the pairs (24 pairs in our case) using what so called the Roulette-Wheel scheme. For selection of the pairs, we generate 2 random numbers in [0,1] consecutively. The pairs of chromosome numbers are obtained by matching the random numbers generated here with the closest values of cumulative fitness values obtained from step 6. Repeat the process until we obtain all 24 chromosome pairs.

8. Crossover
In this step, 25 pairs of chromosomes in step 7 serve as parents. The crossover process occurs with certain probability (denoted by pcross). We expect the crossover process always occurs by specifying large crossover probability, e.g. we use 0.8 in our application. One of methods used in crossover process is called one-cut point crossover. The cut point is obtained by generating random integer from one to L-1=19, here L denotes the length of chromosomes (20 in our case). In the crossover operation we exchange substrings of the parents to the left of the cut points. The crossover process will generate the two new chromosomes called as offspring.

9. Mutation
This step changes one of the genes in a chromosome to be its inverse. The mutation process occurs with the specified mutation probability. Typically, the process of mutation is not always expected to happen, so that the mutation probability value (pmut) is specified to be very small, e.g. 0.1.

10. Population Replacement
In this step, we replace the members of initial population using the chromosomes obtained from step 2-9 above.

These steps (step 2 to 10) are repeated until it converges or the maximum iteration number is reached.

4. EMPIRICAL STUDIES

To illustrate the empirical application of the results discussed in the previous section, in this part we use empirical data of Indonesian government's bonds observed on May 2010. It consists of data about yields and maturity time. This data can be obtained from http://www.idx.co.id.
Based on Nelson-Siegel class model in section 2, we apply hybrid method estimation with genetics algorithm approach and compare it with the performance of estimation based on SQP method. The estimation is implemented in software R.2.15.2 using function *ga* in the package *GA* and function *optim* in package *stats*. The GA method is implemented using 100 iterations, with replication 50 times.
The comparison of mean square error (MSE) between various optimum Nelson-Siegel class models where we apply hybrid-genetic algorithm estimation method is shown in Table 1 below. In Table 1, it shows the MSE of NSSE model smaller than NS, BL, NSS, and RF models. We also

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

compare the MSE using SQP method, as shown in the Table 2. It shows that NSSE model also have MSE smaller than other models. Here, we found that MSE of SQP method in general is smaller than MSE with GA method. On the other hand, we found that the optimal choice for the starting value of the SQP method strictly depends on the data, where for GA, it does not depend on the starting value.

Table 1. MSE of NS class model with GA

Model	May					
	5	6	19	24	25	26
NS	0.096	0.084	0.072	0.119	0.136	0.076
BL	0.070	0.076	0.072	0.096	0.086	0.067
NSS	0.065	0.073	0.237	0.096	0.081	0.065
RF	0.062	0.072	0.061	0.095	0.082	0.065
NSSE	0.051	0.069	0.054	0.088	0.077	0.057

Table 2. MSE of NS class model with SQP Method

Model	May					
	5	6	19	24	25	26
NS	0.023	0.049	0.023	0.045	0.045	0.029
BL	0.020	0.029	0.017	0.026	0.026	0.015
NSS	0.019	0.049	0.023	0.026	0.026	0.013
RF	0.023	0.049	0.023	0.045	0.045	0.029
NSSE	0.014	0.027	0.009	0.0251	0.025	0.0130

In the following tables, we summarize the AIC and BIC values from both methods. In Table 3 it is shown that AIC/BIC of NSSE model smaller than other models. The same result is also obtained using SQP method, see Table 3.

Table 3. AIC/BIC of NS Class Model with GA Method

Model	May					
	5	6	19	24	25	26
NS	-1.435	-1.533	-1.752	-1.066	-0.977	-1.637
	-2.329	-2.426	-2.657	-1.936	-1.835	-2.560
BL	-1.719	-1.604	-1.661	-1.279	-1.368	-1.758
	-2.586	-2.542	-2.542	-2.117	-2.190	-2.662
NSS	-1.676	-1.606	-0.252	-1.240	-1.369	-1.758
	-2.516	-2.533	-1.027	-2.045	-2.155	-2.643
RF	-1.746	-1.640	-1.795	-1.251	-1.340	-1.730
	-2.559	-2.554	-2.629	-2.057	-2.091	-2.597
NSSE	-1.926	-1.665	-1.874	-1.332	-1.431	-1.836
	-2.739	-2.591	-2.708	-2.138	-2.218	-2.721

Table 4. AIC/BIC of NS class model with SQP Method

Model	May					
	5	6	19	24	25	26
NS	-2.678	-1.999	-2.724	-1.984	-2.1458	-2.505
	-3.571	-2.949	-3.629	-2.854	-3.004	-3.428
BL	-2.826	-2.500	-3.033	-2.433	-3.133	-3.138
	-3.719	-3.450	-3.938	-3.103	-3.191	-4.062
NSS	-2.636	-1.985	-2.689	-2.493	-2.902	-2.480
	-3.476	-2.911	-3.547	-3.299	-3.688	-3.365
RF	-2.595	-1.972	-2.655	-1.868	-2.008	-2.455
	-3.381	-2.874	-3.465	-2.609	-2.724	-3.302
NSSE	-3.117	-2.559	-3.647	-2.508	-3.2641	-3.267
	-3.957	-3.485	-4.505	-3.314	-4.051	-4.152

To give overview of the shape of the yield curve obtained from data, we provide the empirical curve estimated using GA method as shown, for instance, in Figure 6. It shows that the NSSE model in general can model the shape of the data better than other methods.

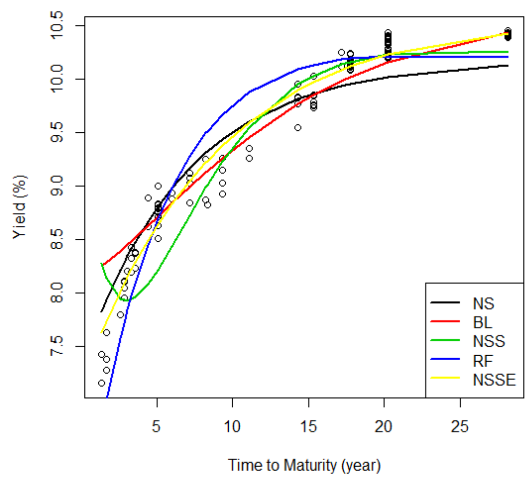

Figure 6. *Yield curve of Nelson-Siegel model Classes*

In figure 6. We show that the yield curve of each model is on the distribution of bond data. The first curve is the NS model which is a black curve, we can see that the beginning of the curve lies outside the distribution of bond data, the curve monotonically through some data and ends with the curve below the spread of bond data. The second curve is the BL model represented by red curve, we can see the curve of the BL model that starts from outside the distribution of data and rises up on the distribution of data. This curve is not too curved compared to the other curves, the next third curve is the NSS model, the curve of this model is represented by a green curve, this curve starts outside the data distribution and curves down after it rises and ends below the spread of bond data. The fourth curve is the RF model which is represented by a blue curve, the curve of this model starts from the bottom of the data obstruction and the curve rises beyond the distribution of data and ends below the bond data distribution, the last NSSE model curve, this curve is represented by a yellow curve, this model curve starts with the right movement on the bond data and curves following the bond data and ends with a fixed end on the bond data, this curve shows that the NSSE model has precise accuracy in determining the yield curve, with AIC, BIC, and MSE having a value smaller than other models, from the comparison of these curves, we conclude that the NSEE model is the best model in determining the yield curve.

5. CONCLUSION

In this paper, we already consider a parametric model class for modelling yield curve, called as Nelson-Siegel model. Various NS models have been considered here, where in particular we propose a new 6-factors model. This model intended to increase the accuracy of the previous considered models in the literature. To estimate the model class, we already considered the hybrid-GA approach. This method does not require the initial value of the parameter and it is able to multiple local optimum of the model. We provide application of various NS model (i.e., 3-factors, 4-factors, 5-factors and 6-factors models) to model Indonesian Government bond data. From this study, we conjectures 6-factors model is the best model that can be used as a tool to determine the yield curve.

REFRENCES:

[1] A. Maria, C. Leanez, and M. Moreno, Estimating Term Structure of Interest Rates: The Venezuelan Case, Working paper series No. 119 (2009).
[2] A. V. Landschoot, Determinants of Euro Term Structure of Credit Spreads, Germany, Working paper, Ghent University (2003)
[3] C. R. Nelson, and A. F. Siegel, Parsimonious Modeling of Yield Curve, Journal of Business **60**(1987), 473-489
[4] D. Boldier, and D. Streliski, Yield Curve Modeling at the Bank of Canada, Canada, (1999).
[5] D. Rosadi, Modeling of the Yield Curve and Its Computation by RcmdrPlugin.Econometrics Package, Proceedings UNDIP 2011, 514-523 (in Indonesian).
[6] F. X. Diebold and C. Li, Forecasting the Term Structure of Government Bonds Yields, *Journal* of Econometrics, **130**(2006), 337-364.
[7] F. X. Diebold, C. Li, and Yue, Global Yield Curve Dynamics and Interactions: A Dynamic Nelson-Siegel Approach, *Journal* of Econometrics, Elsevier, **146** (2007), 351-363.
[8] F. X. Diebold, G. D. Rudebusch, and S. B. Arouba, (2005). The Macroeconomics and the Yield Curve: A Dynamic Latent Factor Approach, Journal of Econometrics, **131**(2005), 309-38.
[9] F. X. Diebold, G. D. Rudebusch, and S. B. Arouba, The Macroeconomics and the Yield Curve: A Nonstructural Analysis, American Economic Review (Papers and Proceedings) **95**(2005), 415-20.
[10] F. X. Diebold, L. Ji, and C. Li, A Three-factors Yield Curve Model: Non-Affine Structure, Systematic Risk Sources, and Generalized Duration, California, Cheltenham, U.K.: Edward Elgar (2004), 240-274.
[11] F. X. Diebold, M. Piazzesi, and G. D. Rudesbusch, Modeling Bond Yields in Finance and Macroeconomics, American Economic Review, **95** (2), 2005, 415–420.
[12] J. Annaert, A. G. P. Claes, M. J. K. D. Ceuster, and H. Zang, Estimating the Yield Curve Using the Nelson Siegel Model, Belgium, Universiteit Antwerpen, (2000).
[13] J. H. Christensen, F. X. Diebold, and G. D. Rudebusch, An Arbitrage-Free Generalized Nelson-Siegel Term Structure Model, Econometrics *Journal*, Royal Economic Society, **12** (2008), C33-C64.
[14] L . Scrucca ,GA: A Package for Genetic Algorithms in R.Journal of Statistical Software,**53**(2012), 1-37, http://www.jstatsoft.org/v53/i04/
[15] L. E. Svensson, Estimating and Interpreting Forward Interest Rates: Sweden 1992-1994, Working paper No. 114(1994).
[16] L. Krippner, A theoretical foundation for the Nelson and Siegel class of yield curve models, New Zealand, Reserve Bank DP 2009/10
[17] M. D. Pooter, Examining the Nelson-Siegel Class of Term Structure Models, Working paper, 2007.
[18] M. Gilli, S. Grobe, and E. Schumann, Calibrating the Nelson-Siegel-Svensson Model, COMISEF, Working Paper WPS-031, (2010).
[19] M. Mitchell, *An Introduction to Genetic Algorithms*, MIT Press, Cambridge, 1999.
[20] R. Gimeno, and J. M. Nave, Genetic Algorithm Estimation of interest Rate Term Structure , Banco De Espana Document de Trabajo N. 0634, (2006).
[21] R.B. Rezende, and M.S. Ferreira, Modeling and Forecasting the Yield Curve an Extended Nelson-Siegel Class of Models: A Quantile Autoregression Approach, Journal *of Forecasting*,**32**(2011),111-123..

[22] Y. S. Stander, Yield Curve Modeling, Palgrave Macmillan, New York, 2005.
[23] Å. Björck, Numerical Methods for Least Squares Problems, Siam, Amsterdam, 1996.
[24] S.S. Rao, Engineering Optimization Theory and Practice, John Wiley & Sons.Inc, New Jersey, 2009.
[25] W. Sun and Y. X. Yuan, Optimization Theory and Methods Nonlinear Programming, Springer, New York, 2006.
[26] E. K. P. Chong and S. H. Zak, An Introduction to Optimization, Jhon Wiley & Sons, Inc, New York, 2001.

APPENDIX

1. List R program of the NS model

```
rm(list=ls()) #
library(GA)
par(mfrow=c(1, 2))
setwd("D:/Data")
#dataall<-
    read.table("Datamei08.txt",header=TRUE)
#colnames(dataall)<-c("Tgl.Transaksi",
    "seri","yield","maturity")
#data=dataall[which(dataall$Tgl.Transaksi=="30-
    May-08"),]
data<-
    read.table("D:/Disertasi/Data/obligasi24.txt",hea
    der=TRUE)
attach(data)
f<- function(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10){1/
     (0.5*sum((yield-x1-x2*exp(-maturity/x4)-
    x3*maturity/x4*exp(-maturity/x4))^2)+
        x5*(x1-x8^2)+x6*(x1+x2-x9^2)+x7*(x4-
    x10^2))
     }
x1 <- x4 <- c(0,15)
x2 <- x3 <- c(-15, 15)
x5 <- x6 <- x7 <- c(-1,0)
x8 <- x9 <- x10 <- c(0,1)
GA <- ga(type = "real-valued", fitness =
    function(x) -1/(f(x[1], x[2], x[3], x[4],
     x[5], x[6], x[7], x[8], x[9], x[10])),
min = c(0, -15, -15, 0, -1, -1, -1, 0, 0, 0), max =
    c(15, 15, 15, 15,
     0, 0, 0, 1, 1, 1),
popSize = 50, maxiter = 100,keepBest = TRUE)
summary(GA)
plot(GA)

NS=function(data,N){
attach(data)
f<- function(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10){1/
     (0.5*sum((yield-x1-x2*exp(-maturity/x4)-
    x3*maturity/x4*exp(-maturity/x4))^2)+
        x5*(x1-x8^2)+x6*(x1+x2-x9^2)+x7*(x4-
    x10^2))
     }
x1 <- c(0,15)
x2 <- x3 <- c(-15, 15)
x4 <- c(0,15)
x5 <- x6 <- x7 <- c(-1,0)
x8 <- x9 <- x10 <- c(0,1)
GA <- ga(type = "real-valued", fitness =
    function(x) -1/(f(x[1], x[2], x[3], x[4],
     x[5], x[6], x[7], x[8], x[9], x[10])),
    min = c(0, -15, -15, 0, -1, -1, -1, 0, 0, 0), max =
    c(15, 15, 15, 15,
     0, 0, 0, 1, 1, 1),
    popSize = 50, maxiter = 100,keepBest = TRUE)
x=GA@solution

for(i in 2:50){
GA <- ga(type = "real-valued", fitness =
    function(x) -1/(f(x[1], x[2], x[3], x[4],
     x[5], x[6], x[7], x[8], x[9], x[10])),
min = c(0, -15, -15, 0, -1, -1, -1, 0, 0, 0), max =
    c(15, 15, 15, 15,
     0, 0, 0, 1, 1, 1),
popSize = 50, maxiter = 100,keepBest = TRUE)
x=rbind(x,GA@solution)
colnames(x)<-
    c("beta0","beta1","beta2","tau","mu1","mu2","
    mu3","a1","a2","a3")}
print(x)
yfit=matrix(0,nc=nrow(x),nr=length(maturity))
mse=mae=mape=aic=bic=NULL
for(i in 1:50){
yfit[,i]=x[i,1]+x[i,2]*exp(-
    maturity/x[i,4])+x[i,3]*maturity/x[i,4]*exp(-
    maturity/x[i,4])
mse[i]=mean((yield-yfit[,i])^2)
mae[i]=mean(sum(abs(yield-yfit[,i])))
mape[i]=mean(sum(abs((yield-yfit[i])/yield)))*100
aic[i]=log(mse[i])+((length(yield)+8)/length(yield))
bic[i]=log(mse[i])+(4*log(length(yield))/length(yiel
    d))

}
plot(maturity,yield,
ylim=c(min(c(yield,yfit)),max(c(yield,yfit))),xlab="
    Maturity (year)",
    ylab="Yield (%)", main="Yield Curve of
    Nelson-Siegel Model")
lines(maturity,rowMeans(yfit),col=2)

output=data.frame(MSE=mse, AIC=aic, BIC=bic)
output
}
NS(data)
```

2. List R program of the BL model

```
rm(list=ls()) # membersih direktori
library(GA)
par(mfrow=c(1, 2))
#setwd("D:/Disertasi/Data")
#dataall<-
    read.table("obligasimay2010.txt",header=TRUE
    )
#colnames(dataall)<-c("date",
    "seri","yield","maturity","ihsg","kurs")
#data=dataall[which(dataall$date=="5-May-10"),]
setwd("D:/Data")
dataall<-read.table("Datamei08.txt",header=TRUE)
colnames(dataall)<-c("Tgl.Transaksi",
    "seri","yield","maturity")
data=dataall[which(dataall$Tgl.Transaksi=="30-
    May-08"),]
#data<-
    read.table("D:/Disertasi/Data/obligasi31.txt",hea
    der=TRUE)
attach(data)
f<-
    function(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x1
    2,x13){1/
     (0.5*sum((yield-x1-x2*exp(-maturity/x4)-
    x3*maturity/x5*exp(-maturity/x5))^2)-
       x6*(x1-x10^2)-x7*(x1+x2-x11^2)-x8*(x4-
    x12^2)-x9*(x5-x13^2))
     }
GA <- ga(type = "real-valued", fitness =
    function(x) -1/(f(x[1], x[2], x[3], x[4],
     x[5], x[6], x[7], x[8], x[9],
    x[10],x[11],x[12],x[13])),
min = c(0,-15,-15,0,0,0,-1,-1,-1,-1,0,0,0), max =
    c(15, 15, 15, 15,15,0,0,0,0,1,1,1,1),
popSize = 50, maxiter = 50,keepBest = TRUE)
summary(GA)
plot(GA)

BL=function(data,N){
attach(data)
f<-
    function(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x1
    2,x13){1/
     (0.5*sum((yield-x1-x2*exp(-maturity/x4)-
    x3*maturity/x5*exp(-maturity/x5))^2)-
       x6*(x1-x10^2)-x7*(x1+x2-x11^2)-x8*(x4-
    x12^2)-x9*(x5-x13^2))
     }
GA <- ga(type = "real-valued", fitness =
    function(x) -1/(f(x[1], x[2], x[3], x[4],
     x[5], x[6], x[7], x[8], x[9],
    x[10],x[11],x[12],x[13])),
min = c(0,-15,-15,0,0,0,-1,-1,-1,-1,0,0,0), max =
    c(15, 15, 15, 15,15,0,0,0,0,1,1,1,1),
popSize = 50, maxiter = 50,keepBest = TRUE)
x=GA@solution

for(i in 2:50){
GA <- ga(type = "real-valued", fitness =
    function(x) -1/(f(x[1], x[2], x[3], x[4],
     x[5], x[6], x[7], x[8], x[9],
    x[10],x[11],x[12],x[13])),
min = c(0,-15,-15,0,0,0,-1,-1,-1,-1,0,0,0), max =
    c(15, 15, 15, 15,15,0,0,0,0,1,1,1,1),
popSize = 50, maxiter = 50,keepBest = TRUE)
x=rbind(x,GA@solution)
colnames(x)<-
    c("beta0","beta1","beta2","tau1","tau2","mu1","
    mu2","mu3","mu4","a1","a2","a3","a4")
}
print(x)
yfit=matrix(0,nc=nrow(x),nr=length(maturity))
mse=mae=mape=aic=bic=NULL
for(i in 1:50){
yfit[,i]=x[i,1]+x[i,2]*exp(-
    maturity/x[i,4])+x[i,3]*maturity/x[i,5]*exp(-
    maturity/x[i,5])
mse[i]=mean((yield-yfit[,i])^2)
mae[i]=mean(sum(abs(yield-yfit[,i])))
mape[i]=mean(sum(abs((yield-yfit[i])/yield)))*100
aic[i]=log(mse[i])+((length(yield)+10)/length(yield
    ))
bic[i]=log(mse[i])+(5*log(length(yield))/length(yiel
    d))

}
plot(maturity,yield,
ylim=c(min(c(yield,yfit)),max(c(yield,yfit))),xlab="
    Maturity (year)",
   ylab="Yield (%)", main="Yield Curve of Bliss
    Model")
lines(maturity,rowMeans(yfit),col=2)

output=data.frame(MSE=mse, AIC=aic, BIC=bic)
output
}
BL(data)
```

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3. List R program of the NSS model

```
rm(list=ls())
library(GA)
#par(mfrow=c(1, 2))
#setwd("D:/Data")
#dataall<-
    read.table("Datamei08.txt",header=TRUE)
#colnames(dataall)<-c("Tgl.Transaksi",
    "seri","yield","maturity")
#data=dataall[which(dataall$Tgl.Transaksi=="30-
    May-08"),]
data<-
    read.table("D:/Disertasi/Data/obligasi19.txt",hea
    der=TRUE)
attach(data)
f<-
    function(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x1
    2,x13,x14){
     0.5*sum((yield-x1-x2*exp(-maturity/x5)-
    x3*maturity/x5*exp(-maturity/x5)-
        x4*maturity/x6*exp(-maturity/x6))^2)-
    x7*(x1-x11^2)-x8*(x1+x2-x12^2)-
        x9*(x5-x13^2)-x10*(x6-x14^2)
     }
x1 <- c(0,15)
x2 <- x3 <- x4 <- c(-15, 15)
x5 <- x6 <- c(0,15)
x7 <- x8 <- x9 <- x10 <- c(-1,0)
x11 <- x12 <- x13 <- x14 <- c(0,1)
GA <- ga(type = "real-valued", fitness =
    function(x) -f(x[1], x[2], x[3], x[4],
x[5],x[6],x[7],x[8],x[9],x[10],x[11],x[12],x[13],x[1
    4]),
min = c(0, -15, -15, -15, 0, 0, -1, -1, -1, -1, 0, 0, 0,
    0),
max = c(15, 15, 15, 15,15,15,0,0,0,0,1,1,1,1),
popSize = 50, maxiter = 100,keepBest = TRUE)
summary(GA)
plot(GA)

NSS=function(data,N){
attach(data)
f<-
    function(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x1
    2,x13,x14){
     0.5*sum((yield-x1-x2*exp(-maturity/x5)-
    x3*maturity/x5*exp(-maturity/x5)-
        x4*maturity/x6*exp(-maturity/x6))^2)-
    x7*(x1-x11^2)-x8*(x1+x2-x12^2)-
        x9*(x5-x13^2)-x10*(x6-x14^2)
     }
x1 <- c(0,15)
x2 <- x3 <- x4 <- c(-15, 15)
x5 <- x6 <- c(0,15)
x7 <- x8 <- x9 <- x10 <- c(-1,0)
x11 <- x12 <- x13 <- x14 <- c(0,1)
GA <- ga(type = "real-valued", fitness =
    function(x) -f(x[1], x[2], x[3], x[4],
x[5],x[6],x[7],x[8],x[9],x[10],x[11],x[12],x[13],x[1
    4]),
min = c(0, -15, -15, -15, 0, 0, -1, -1, -1, -1, 0, 0, 0,
    0),
max = c(15, 15, 15, 15,15,15,0,0,0,0,1,1,1,1),
popSize = 50, maxiter = 100,keepBest = TRUE)
x = GA@solution

for(i in 2:50){
GA <- ga(type = "real-valued", fitness =
    function(x) -f(x[1], x[2], x[3], x[4],
x[5],x[6],x[7],x[8],x[9],x[10],x[11],x[12],x[13],x[1
    4]),
min = c(0, -15, -15, -15, 0, 0, -1, -1, -1, -1, 0, 0, 0,
    0),
max = c(15, 15, 15, 15,15,15,0,0,0,0,1,1,1,1),
popSize = 50, maxiter = 100,keepBest = TRUE)
x=rbind(x,GA@solution)
colnames(x)=c("beta0","beta1","beta2","beta3","tau
    1","tau2","mu1","mu2","mu3","mu4","a1",
        "a2","a3","a4")
}
print(x)
yfit=matrix(0,nc=nrow(x),nr=length(maturity))
mse=mae=mape=aic=bic=NULL
for(i in 1:50){
yfit[,i]=x[i,1]+x[i,2]*exp(-
    maturity/x[i,5])+x[i,3]*maturity/x[i,5]*exp(-
    maturity/x[i,5])+
   x[i,4]*maturity/x[i,6]*exp(-maturity/x[i,6])

mse[i]=mean((yield-yfit[,i])^2)
aic[i]=log(mse[i])+((length(yield)+12)/length(yield
    ))
bic[i]=log(mse[i])+(6*log(length(yield))/length(yiel
    d))
}
plot(maturity,yield,ylim=c(min(c(yield,yfit)),max(c
    (yield,yfit))),xlab="Maturity (year)",
   ylab="Yield (%)",main="Yield Curve of
    Svensson Model")
lines(maturity,rowMeans(yfit),col=2)
output=data.frame(MSE=mse, AIC=aic, BIC=bic)
output
}
NSS(data)
```

4. List R program of the RF model

```
rm(list=ls
require("GA")
```

```
par(mfrow=c(1, 2))
#setwd("D:/Disertasi/Data")
#dataall<-
    read.table("obligasi2012.txt",header=TRUE)
#colnames(dataall)<-
    c("date","code","yield","maturity","kurs","ihsg"
    )
#data=dataall[which(dataall$date=="3-Jan-12"),]
setwd("D:/Data")
dataall<-read.table("Datamei08.txt",header=TRUE)
colnames(dataall)<-c("Tgl.Transaksi",
    "seri","yield","maturity")
data=dataall[which(dataall$Tgl.Transaksi=="30-
    May-08"),]
#data<-
    read.table("D:/Disertasi/Data/obligasi7.txt",head
    er=TRUE)
attach(data)
f<-
    function(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x1
    2,x13,x14,x15){1/
     (0.5*sum((yield-x1-x2*exp(-maturity/x6)-
    x3*exp(-maturity/x7)-x4*maturity/x6*exp(-
    maturity/x6)-
           x5*maturity/x7*exp(-maturity/x7))^2)-
     x8*(x1-x12^2)-x9*(x1+x2+x3-x13^2)-
    x10*(x6-x14^2)-x11*(x7-x15^2))
     }
x1 <- c(0,15)
x2 <- x3 <- x4 <-x5<- c(-15,15)
x6 <- x7<-c(0,15)
x8<-x9<-x10<-x11<- c(-1,0)
x12<-x13<-x14<-x15<- c(0,1)
GA <- ga(type = "real-valued", fitness =
    1(function(x) -f(x[1], x[2], x[3], x[4],
x[5],x[6],x[7],x[8],x[9],x[10],x[11],x[12],x[13],x[1
    4],x[15])),
min = c(0,-15,-15,-15,-15,0,0,-1,-1,-1,-1,0,0,0,0),
max = c(15,15,15,15,15,15,15,0,0,0,0,1,1,1,1),
popSize = 50, maxiter = 100,keepBest = TRUE)
summary(GA)
plot(GA)

RF=function(data,N){
attach(data)
f<-
    function(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x1
    2,x13,x14,x15){1/
     (0.5*sum((yield-x1-x2*exp(-maturity/x6)-
    x3*exp(-maturity/x7)-x4*maturity/x6*exp(-
    maturity/x6)-
           x5*maturity/x7*exp(-maturity/x7))^2)-
             x8*(x1-x12^2)-x9*(x1+x2-x13^2)-
    x10*(x6-x14^2)-x11*(x7-x15^2))
     }
x1 <- c(0,15)
x2 <- x3 <- x4 <-x5<- c(-15,15)
x6 <- x7<-c(0,15)
x8<-x9<-x10<-x11<- c(-1,0)
x12<-x13<-x14<-x15<- c(0,1)
GA <- ga(type = "real-valued", fitness =
    function(x) -1/(f(x[1], x[2], x[3], x[4],
x[5],x[6],x[7],x[8],x[9],x[10],x[11],x[12],x[13],x[1
    4],x[15])),
min = c(0,-15,-15,-15,-15,0,0,-1,-1,-1,-1,0,0,0,0),
max = c(15,15,15,15,15,15,15,0,0,0,0,1,1,1,1),
popSize = 50, maxiter = 100,keepBest = TRUE)
x = GA@solution

for(i in 2:50){
GA <- ga(type = "real-valued", fitness =
    function(x) -1/(f(x[1], x[2], x[3], x[4],
x[5],x[6],x[7],x[8],x[9],x[10],x[11],x[12],x[13],x[1
    4],x[15])),
min = c(0,-15,-15,-15,-15,0,0,-1,-1,-1,-1,0,0,0,0),
max = c(15,15,15,15,15,15,15,0,0,0,0,1,1,1,1),
popSize = 50, maxiter = 100,keepBest = TRUE)
x=rbind(x,GA@solution)
colnames(x)<-
    c("beta0","beta1","beta2","beta3","beta4","tau1
    ","tau2","mu1","mu2",
           "mu3","mu4","a1","a2","a3","a4")
}
print(x)
yfit=matrix(0,nc=nrow(x),nr=length(maturity))
mse=mae=mape=aic=bic=NULL
for(i in 1:50){
yfit[,i]=x[i,1]+x[i,2]*exp(-
    maturity/x[i,6])+x[i,3]*exp(-maturity/x[i,7])+
   x[i,4]*maturity/x[i,6]*exp(-
    maturity/x[i,6])+x[i,5]*maturity/x[i,7]*exp(-
    maturity/x[i,7])
mse[i]=mean((yield-yfit[,i])^2)
aic[i]=log(mse[i])+((length(yield)+14)/length(yield
    ))
bic[i]=log(mse[i])+(7*log(length(yield))/length(yiel
    d))
}
plot(maturity,yield,ylim=c(min(c(yield,yfit)),max(c
    (yield,yfit))),xlab="Maturity (year)",
   ylab="Yield (%)",main="Yield Curve of
    Rezende & Ferreira Model")
lines(maturity,rowMeans(yfit),col=2)

output=data.frame(MSE=mse, AIC=aic, BIC=bic)
output
}
RF(data)
```

5. List R program of the NSSE model

```
rm(list=ls()) # membersih direktori
```

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

```
library(GA)
par(mfrow=c(1, 2))
#setwd("D:/Disertasi/Data")
#dataall<-read.table("opdes12.txt",header=TRUE)
#colnames(dataall)<-
    c("date","code","yield","maturity","kurs","ihsg"
    )
#data=dataall[which(dataall$date=="3-Jan-2012"),]
#setwd("D:/Data")
#dataall<-
    read.table("Datamei08.txt",header=TRUE)
#colnames(dataall)<-c("Tgl.Transaksi",
    "seri","yield","maturity")
#data=dataall[which(dataall$Tgl.Transaksi=="30-
    May-08"),]
data<-
    read.table("D:/Disertasi/Data/obligasi19.txt",he
    ader=TRUE)
attach(data)
f<-
    function(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x1
    2,x13,x14){
    0.5*sum((yield-x1-x2*exp(-maturity/x5)-
    x3*(maturity/x5*exp(-
    maturity/x5)+maturity/x6*exp(-maturity/x6))-
        x4*maturity/x6*exp(-maturity/x6))^2)-
    x7*(x1-x11^2)-x8*(x1+x2-x12^2)-
        x9*(x5-x13^2)-x10*(x6-x14^2)
    }
GA <- ga(type = "real-valued", fitness = function(x)
    -f(x[1], x[2], x[3], x[4],
x[5],x[6],x[7],x[8],x[9],x[10],x[11],x[12],x[13],x[14
    ]),
min = c(0, -15, -15, -15, 0, 0, -0.1, -0.1, -0.1, -0.1, 0,
    0, 0, 0),
max = c(15, 15, 15, 15,15,15,0,0,0,0,0.1,0.1,0.1,0.1),
popSize = 50, maxiter = 100,keepBest = TRUE)
summary(GA)
plot(GA)
SVE=function(data,N){
attach(data)
f<-
    function(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x1
    2,x13,x14){
    0.5*sum((yield-x1-x2*exp(-maturity/x5)-
    x3*(maturity/x5*exp(-
    maturity/x5)+maturity/x6*exp(-maturity/x6))-
        x4*maturity/x6*exp(-maturity/x6))^2)-
    x7*(x1-x11^2)-x8*(x1+x2-x12^2)-
        x9*(x5-x13^2)-x10*(x6-x14^2)
    }
GA <- ga(type = "real-valued", fitness = function(x)
    -f(x[1], x[2], x[3], x[4],
x[5],x[6],x[7],x[8],x[9],x[10],x[11],x[12],x[13],x[14
    ]),
min = c(0, -15, -15, -15, 0, 0, -0.1, -0.1, -0.1, -0.1, 0,
    0, 0, 0),
max = c(15, 15, 15, 15,15,15,0,0,0,0,0.1,0.1,0.1,0.1),
popSize = 50, maxiter = 100,keepBest = TRUE)
x = GA@solution

for(i in 2:50){
GA <- ga(type = "real-valued", fitness = function(x)
    -f(x[1], x[2], x[3], x[4],
x[5],x[6],x[7],x[8],x[9],x[10],x[11],x[12],x[13],x[14
    ]),
min = c(0, -15, -15, -15, 0, 0, -0.1, -0.1, -0.1, -0.1, 0,
    0, 0, 0),
max = c(15, 15, 15, 15,15,15,0,0,0,0,0.1,0.1,0.1,0.1),
popSize = 50, maxiter = 100,keepBest = TRUE)
x=rbind(x,GA@solution)
colnames(x)=c("beta0","beta1","beta2","beta3","tau
    1","tau2","mu1","mu2","mu3","mu4","a1",
        "a2","a3","a4")
}
print(x)
yfit=matrix(0,nc=nrow(x),nr=length(maturity))
mse=mae=mape=aic=bic=NULL
for(i in 1:50){
yfit[,i]=x[i,1]+x[i,2]*exp(-maturity/x[i,5])+
     x[i,3]*(maturity/x[i,5]*exp(-
    maturity/x[i,5])+maturity/x[i,6]*exp(-
    maturity/x[i,6]))+
     x[i,4]*maturity/x[i,6]*exp(-maturity/x[i,6])
mse[i]=mean((yield-yfit[,i])^2)
aic[i]=log(mse[i])+((length(yield)+12)/length(yield)
    )
bic[i]=log(mse[i])+(6*log(length(yield))/length(yiel
    d))
}
plot(maturity,yield,ylim=c(min(c(yield,yfit)),max(c(
    yield,yfit))),xlab="Maturity (year)",
   ylab="Yield (%)",main="Yield Curve of
    Svensson Extended Model")
lines(maturity,rowMeans(yfit),col=2)
output=data.frame(MSE=mse, AIC=aic, BIC=bic)
output
}
SVE(data)
```