Historia Mathematica Vol. 11, No. 2, 1998, 63-76

An Effective Algorithm of Power Transformation: Box-Cox Transformation*

Seo Kyeong University **Seung-Woo Lee**
Hanyang University **Kyung-Joon Cha**

Abstract

When teaching the linear regression analysis in the class, the power transformation must be introduced to fit the linear regression model for nonlinear data. Box and Cox (1964) proposed the attractive power transformation technique which is so called Box-Cox transformation.

In this paper, an effective algorithm selecting an appropriate value for Box-Cox transformation is developed which is considered to find a value minimizing error sum of squares. When the proposed algorithm is used to find a value for transformation, the number of iterations needs to be considered. Thus, the number of iterations is examined through simulation study. Since SAS is one of most widely used packages and does not provide the procedure that performs iterative Box-Cox transformation, a SAS program automatically choosing the value for transformation is developed. Hence, the students could learn how the Box-Cox transformation works, moreover, researchers can use this for analysis of data.

0. Introduction

Regression analysis was first developed by Sir F. Galton, well-known anthropologist and meteorologist, in the latter part of the 19th century. Galton has studied the relation

* The authors wish to acknowledge the financial support of Hanyang University, Korea, made in the program year of 1997.

between hights of fathers and sons and noted that the heights of sons of both tall and short fathers appeared to revert or regress to the mean of the group. Since then, many new methods, such as least squares method and transformations, for regression analysis have developed. Thus, the subject of linear regression analysis becomes one of major courses in teaching statistics. It mainly concerns the study of linear relations among variables, for the purpose of constructing models, prediction and making inferences. The relationship is expressed as an equation that predicts a response variable (dependent variable) from a function of regressor (independent variable) and parameters.

A simple linear regression model where there is only one independent variable can be stated as

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \quad i = 1, 2, \cdots, n. \tag{0.1}$$

Here, the ε_i are independently and identically distributed normal random errors with mean zero and common variance σ^2. The x_i is the level of the independent variable as a known constant and the y_i is the observed response in the ith trial. Also, the β_0 and β_1 are parameters to estimate so that a measure of fit is optimized.

The measure of variation in the data with the regression model is the sum of the squared deviation, which is expressed as $SSE = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$. Here SSE denotes error sum of squares and $\hat{y}_i = b_0 + b_1 x_i$ is the fitted regression line. If $SSE = 0$, all observations fall on the fitted regression line.

When we want to use linear regression model for analysis, it is necessary to consider the use of transformations of one or both of the original variables before carrying out the regression analysis. Simple transformations of either the dependent variable y or the independent variable x, or of both, are often sufficient to make the simple linear regression model appropriate for the transformed data.

There are two view points to consider for transformation. First, the relationship between y and x is nonlinear but the usual assumptions of normally and independently distributed responses with constant variance are at least approximately satisfied. Second, we wish to transform y to correct non-normality and nonconstant variance. Here, we consider the second case mentioned above, since it affects the full regression model more sensitively and is more complicate.

Box and Cox (1964) introduced a procedure for choosing a transformation from the family of power transformations on y. Recently, Box-Cox transformation is reexamined

by a few authors. Logothetis (1990) applied this procedure to Taguchi's method for the optimization of multiresponse plasma etching process. Fearn (1992), also, examined the same procedure to another Taguchi method.

As Seber (1977) mentioned, the useful family of power transformations which is used for correcting skewness of the distributions of error terms, unequal error variances, and nonlinearity of the regression function is of the form

$$y^{(\lambda)} = \begin{cases} y^{\lambda} & (\lambda \neq 0) \\ \log y & (\lambda = 0), \end{cases} \tag{0.2}$$

where λ is a parameter to be determined form the data.

The criterion for determining the appropriate parameter λ of the transformation of y in the Box-Cox approach is to find the value of λ that minimizes the SSE for a linear regression based on that transformation. In other words, we can select a number of values of λ, make the corresponding transformation for each, fit the linear regression function to the transformed y data, and calculate SSE for each fit. That value of λ is then chosen that minimizes SSE. This is the reason why we need an algorithm and a program which performs the iterative transformations with a few different values of λ so that the appropriate value for power transformation is automatically selected.

Thus, the technique of power transformation as exemplified by Box-Cox transformation that is used to choose a parameter for the power transformation could simultaneously achieves

(i) simplicity (linearity) of the model structure for $E(y)$;

(ii) constancy of error variance or, equivalently, independence between cell mean and cell variance, i.e. between the sample mean and the sample variance of the observations in each experimental trial;

(iii) normality of distributions;

(iv) independence of observations.

It has been recognized for a long time that data transformation methods capable of achieving (i)-(iv) could have a crucial role in statistical analysis, especially towards an efficient application of techniques such as analysis of variance (ANOVA) and multiple regression analysis.

Therefore, it is necessary to use some transformation techniques before carrying out linear regression analysis and the power transformation, Box-Cox method, is one of the best methods to achieve these goals.

1. Analytical procedures of power transformation

We work with a parametric family of transformations from y to $y^{(\lambda)}$, the parameter λ, possibly a vector, defining a particular transformation. As Box and Cox (1964) suggested and lately illustrated by a few authots such as Cook and Weisberg (1982), two important power transformations need to be considered are

$$y^{(\lambda)} = \begin{cases} \dfrac{y^{\lambda}-1}{\lambda} & (\lambda \neq 0) \\ \log y & (\lambda = 0), \end{cases} \tag{1.1}$$

and

$$y^{(\lambda)} = \begin{cases} \dfrac{(y+\lambda_2)^{\lambda_1}-1}{\lambda_1} & (\lambda_1 \neq 0) \\ \log(y+\lambda_2) & (\lambda_1 = 0). \end{cases} \tag{1.2}$$

The transformations (1.1) contains the usual log, square root and inverse transformation as special cases. It is assumed that for each λ, $y^{(\lambda)}$ is a monotonic function of y over the admissible range. In fact, (1.1) is identical to (0.2) and is a modification of (0.2) to avoid discontinuity at λ=0 when the regression model (0.1) contains a constant term β_0 (see Schlesselman, 1971). Also, for $y+\lambda_2>0$, the transformations (1.2) is the extended power family which can be used if the origin is artificial and negative responses occur. In some situations, it may be sufficient to substitute a convenient value for λ_2 and then proceed using (1.1) in combination with the shifted response $y+\lambda_2$.

Now, to investigate relation between minimizing *SSE* and the power transformation, suppose that we observe an $n\times 1$ vector of observations $\vec{y}=\{y_1,\cdots,y_n\}$, and that the appropriate linear model for the problem is specified by

$$E\{\vec{y}^{(\lambda)}\} = a\vec{\theta}, \tag{1.3}$$

where $\vec{y}^{(\lambda)}$ is the column vector of transformed observations, a is a known matrix and

$\vec{\theta}$ a vector of unknown parameters associated with the transformed observations.

We now assume that for some unknown λ, the transformed observations $y_i^{(\lambda)}$, $i=1,\cdots,n$, satisfy the full normal theory assumptions, i.e. are independently and normally distributed with constant variance σ^2, and with expectations (1.3). The probability density for the untransformed observations, and hence the likelihood in relation to these original observations, is obtained by multiplying the normal density by the Jacobian of the transformation. Thus, the likelihood in relation to the original observations $\vec{y}$ is

$$\frac{1}{(2\pi)^{1/2n}\sigma^n}\exp\left\{-\frac{(\vec{y}^{(\lambda)}-a\vec{\theta})'(\vec{y}^{(\lambda)}-a\vec{\theta})}{2\sigma^2}\right\}J(\lambda;\vec{y}), \tag{1.4}$$

where $J(\lambda;\vec{y})=\prod_{i=1}^{n}\left|\frac{dy_i^{(\lambda)}}{dy_i}\right|=\prod_{i=1}^{n}y_i^{\lambda-1}$.

In fact, we can use the maximum-likelihood theory to find parameters in (1.4) and is equivalent to a standard least-squares problem. Hence, for fixed λ, the estimate of σ^2, $\hat{\sigma}^2(\lambda)$, is

$$\hat{\sigma}^2(\lambda)=\vec{y}^{(\lambda)'}a_r\vec{y}^{(\lambda)}/n=S(\lambda)/n,$$

where $a_r=I-a(a'a)^{-1}a'$ with full rank matrix a and $S(\lambda)$ is the error sum of squares. For fixed λ, the maximized log likelihood is

$$L_{\max}(\lambda)=-\frac{1}{2}n\log\hat{\sigma}^2(\lambda)+\log J(\lambda;\vec{y}).$$

Since we need to find the value $\hat{\lambda}$, the derivative with respect to λ can be used. In the special care of one parameter power transformation, it is given by

$$\frac{d}{d\lambda}L_{\max}(\lambda)=-n\frac{\vec{y}^{(\lambda)'}a_r\vec{u}^{(\lambda)}}{\vec{y}^{(\lambda)'}a_r\vec{y}^{(\lambda)}}+\frac{n}{\lambda}+\sum\log y_i,$$

where $\vec{u}^{(\lambda)}$ is the vector of components $\{\lambda^{-1}y_i^{\lambda}\log y_i\}$.

Moreover, the above results can be expressed very simply if we work with the normalized transformation

$$\vec{Z}^{(\lambda)} = \vec{y}^{(\lambda)} / J^{1/n},$$

where $J = J(\lambda : \vec{y})$. Then,

$$L_{\max}(\lambda) = -\frac{1}{2} \log \hat{\sigma}^2(\lambda : \vec{z}),$$

where $\hat{\sigma}^2(\lambda : \vec{z}) = \dfrac{\vec{z}^{(\lambda)'} a_r \vec{z}^{(\lambda)}}{n} = \dfrac{S(\lambda : \vec{z})}{n}$ and $S(\lambda : \vec{z})$ is the error sum of squares of $z^{(\lambda)}$. Then, the maximum likelihood is the proportional to $\{S(\lambda : \vec{z})\}^{-n}$ and the maximum likelihood estimate, i.e., least squares estimate, is obtained by minimizing $S(\lambda : \vec{z})$ with respect λ. Hence, the appropriate value of λ for power transformation can be obtained.

Therefore, for the simple power transformation, the resulting normalized values are then expressed as

$$z^{(\lambda)} = \begin{cases} \dfrac{y^{\lambda} - 1}{\lambda \dot{y}^{\lambda - 1}} & (\lambda \neq 0) \\ \dot{y} \log y & (\lambda = 0), \end{cases}$$

where $\dot{y}$ is the geometric mean, $(\prod_{i=1}^{n} y_i)^{1/n}$, of the observations.

For the power transformation with shifted location, $z^{(\lambda)}$ is defined by

$$z^{(\lambda)} = \begin{cases} \dfrac{(y + \lambda_2)^{\lambda_1} - 1}{\lambda_1 gm(y + \lambda_2)^{\lambda_1 - 1}} & (\lambda_1 \neq 0) \\ gm(y + \lambda_2) \log(y + \lambda_2) & (\lambda_1 = 0), \end{cases}$$

where $gm(y + \lambda_2)$ is the sample geometric mean of the $(y + \lambda_2)$s.

2. Numerical analysis and conclusions

In this section, we perform the simulation study to see how the proposed algorithm works. The flowchart is made by procedures explained in section 1 and is programmed

by SAS, these are given in the Appendix.

Since it is important to examine the number iterations to find an appropriate value for the power transformation as well as the transformation itself, we consider the two parts for numerical analysis. First, we try to see whether the number of iterations changes a value of λ. Second, we use the λ that is chosen by the proposed algorithm, then check the transformation through plots of original and transformed data. Also, we consider two types of data, nonlinear and linear data, to see if the proposed algorithm picks an appropriate value of transformation for both cases.

For Table 1, the data is adapted from Chatterjee and Price (1977), where the dependent variable represents the number of bacteria as estimated by plate counts in an experiment with marine bacterium following exposure to 200 kilovolt x-rays for periods ranging from 1 to 15 intervals of 6 minutes.

It is easy to see that there exists nonlinear relation between independent and dependent variables from Figure 1. As we mentioned earlier, the number of iterations is important to find an appropriate value of λ, thus we have tried 11, 15, 21, 25, 31 different numbers of iterations and the results are shown in Table 1. We can see that each iteration chooses the λ=0 which is the equivalent to the transformation of $\log y$. We can see, from Figure 2, that the chosen transformation correctly transforms the original data as linear.

The another data is selected from Montgomery and Peck (1981). The data represents relation between shear strength as dependent variable and the age of propellant as independent variable. From Figure 3, we can expect the linear relation between two variables. However, from Table 2, the proposed algorithm picks the value of λ=0.8 or 0.9 for the power transformation. That is, the original data represents the linear-like relation, the proposed algorithm picks the value close to 1. The transformmed data is plotted in Figure 4 and it can be seen more like linear of transformmed data than of the original data.

As results, we can say that 10-20 iterations is enough to select a good transformation. That is, the proposed algorithm is not very sensitive for the number of iterations. Also, the proposed algorithm picks an appropriate value for the transformation of data that looks like linear, and gives better linear relation between two variables.

Moreover, in educational points of view, students could easily understand the power transformation by going through the algorithm without complete theoretical background. Also, researchers perform better analysis when they would use the linear regression models.

Table 1. Different numbers of iterations for surviving bacteria

OBS	λ	*SSE*	λ	*SSE*	λ	*SSE*	λ	*SSE*	λ	*SSE*
1	-1.0	16981.46	-1.00000	16981.46	-1.0	16981.46	-1.00000	16981.46	-1.00000	16981.46
2	-0.8	9624.55	-0.85714	11377.06	-0.9	12860.78	-0.91667	13480.59	-0.93333	14125.82
3	-0.6	5125.24	-0.71429	7416.89	-0.8	9624.55	-0.83333	10617.06	-0.86667	11693.60
4	-0.4	2479.33	-0.57143	4651.05	-0.7	7092.63	-0.75000	8280.13	-0.80000	9624.55
5	-0.2	1110.15	-0.42857	2769.33	-0.6	5125.24	-0.66667	6380.02	-0.73333	7867.71
6	0.0	736.72	-0.28571	1564.41	-0.5	3614.71	-0.58333	4844.25	-0.66667	6380.02
7	0.2	1315.48	-0.14286	907.99	-0.4	2479.33	-0.50000	3614.71	-0.60000	5125.24
8	0.4	3042.62	0.00000	736.72	-0.3	1658.65	-0.41667	2645.31	-0.53333	4072.98
9	0.6	6421.34	0.14286	1046.41	-0.2	1110.15	-0.33333	1900.20	-0.46667	3197.95
10	0.8	12416.84	0.28571	1893.98	-0.1	806.99	-0.25000	1352.40	-0.40000	2479.33
11	1.0	22749.38	0.42857	3407.66	0.0	736.72	-0.16667	982.74	-0.33333	1900.20
12			0.57143	5807.39	0.1	900.87	-0.08333	779.24	-0.26667	1447.20
13			0.71429	9438.65	0.2	1315.48	0.00000	736.72	-0.20000	1110.15
14			0.85714	14825.20	0.3	2012.55	0.08333	856.74	-0.13333	881.84
15			1.00000	22749.38	0.4	3042.62	0.16667	1147.82	-0.06667	757.90
16					0.5	4478.63	0.25000	1626.02	0.00000	736.72
17					0.6	6421.34	0.33333	2315.81	0.06667	819.43
18					0.7	9006.88	0.41667	3251.43	0.13333	1010.01
19					0.8	12416.84	0.50000	4478.63	0.20000	1315.48
20					0.9	16891.81	0.58333	6057.09	0.26667	1746.10
21					1.0	22749.38	0.66667	8063.47	0.33333	2315.81
22							0.75000	10595.34	0.40000	3042.62
23							0.83333	13776.27	0.46667	3949.28
24							0.91667	17762.15	0.53333	5063.99
25							1.00000	22749.38	0.60000	6421.34
26									0.66667	8063.47
27									0.73333	10041.42
28									0.80000	12416.84
29									0.86667	15264.03
30									0.93333	18672.38
31									1.00000	22749.38

Table 2. Different numbers of iterations for shear strength

OBS	λ	SSE	λ	SSE	λ	SSE	λ	SSE	λ	SSE
1	-1.0	1540.49	-1.00000	1540.49	-1.0	1540.49	-1.00000	1540.49	-1.00000	1540.49
2	-0.8	1003.06	-0.85714	1131.92	-0.9	1240.46	-0.91667	1285.70	-0.93333	1332.74
3	-0.6	665.02	-0.71429	839.06	-0.8	1003.06	-0.83333	1076.14	-0.86667	1155.11
4	-0.4	450.46	-0.57143	628.13	-0.7	814.74	-0.75000	903.48	-0.80000	1003.06
5	-0.2	313.36	-0.42857	475.53	-0.6	665.02	-0.66667	760.99	-0.73333	872.76
6	0.0	225.57	-0.28571	364.75	-0.5	545.71	-0.58333	643.21	-0.66667	760.99
7	0.2	169.82	-0.14286	284.16	-0.4	450.46	-0.50000	545.71	-0.60000	665.02
8	0.4	135.49	0.00000	225.57	-0.3	374.31	-0.41667	464.89	-0.53333	582.52
9	0.6	116.11	0.14286	183.19	-0.2	313.36	-0.33333	397.83	-0.46667	511.56
10	0.8	107.84	0.28571	152.94	-0.1	264.57	-0.25000	342.14	-0.40000	450.46
11	1.0	108.62	0.42857	131.92	0.0	225.57	-0.16667	295.87	-0.33333	397.83
12			0.57143	118.13	0.1	194.47	-0.08333	257.45	-0.26667	352.47
13			0.71429	110.19	0.2	169.82	0.00000	225.57	-0.20000	313.36
14			0.85714	107.20	0.3	150.46	0.08333	199.17	-0.13333	279.64
15			1.00000	108.62	0.4	135.49	0.16667	177.40	-0.06667	250.59
16					0.5	124.22	0.25000	159.54	0.00000	225.57
17					0.6	116.11	0.33333	145.02	0.06667	204.06
18					0.7	110.75	0.41667	133.37	0.13333	185.60
19					0.8	107.84	0.50000	124.22	0.20000	169.82
20					0.9	107.18	0.58333	117.26	0.26667	156.38
21					1.0	108.62	0.66667	112.25	0.33333	145.02
22							0.75000	109.01	0.40000	135.49
23							0.83333	107.38	0.46667	127.60
24							0.91667	107.27	0.53333	121.19
25							1.00000	108.62	0.60000	116.11
26									0.66667	112.25
27									0.73333	109.52
28									0.80000	107.84
29									0.86667	107.16
30									0.93333	107.43
31									1.00000	108.62

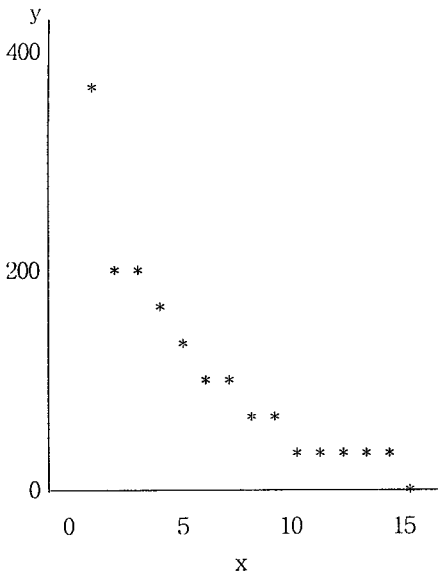


Fig. 1 Scatter Plot of Original Data

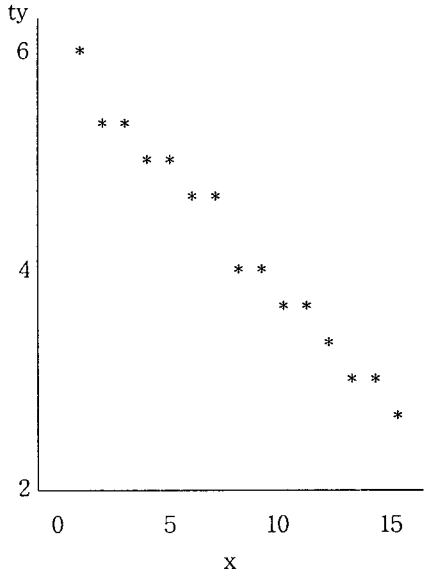


Fig. 2 Scatter Plot of Transformed Data

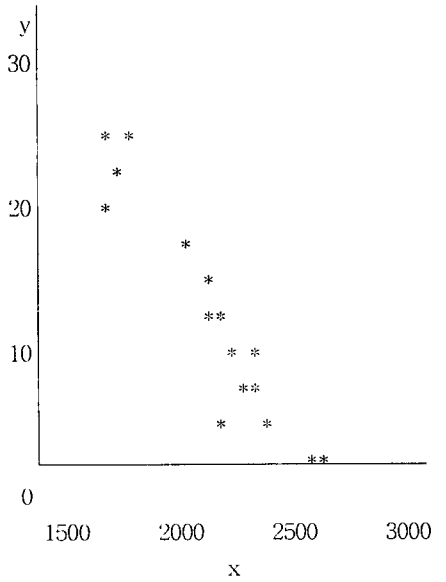


Fig. 3 Scatter Plot of Original Data

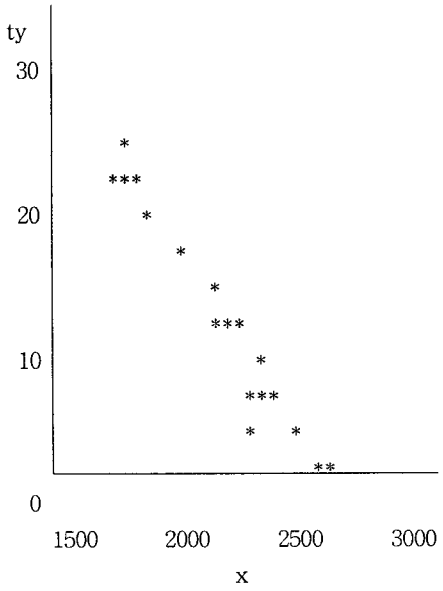


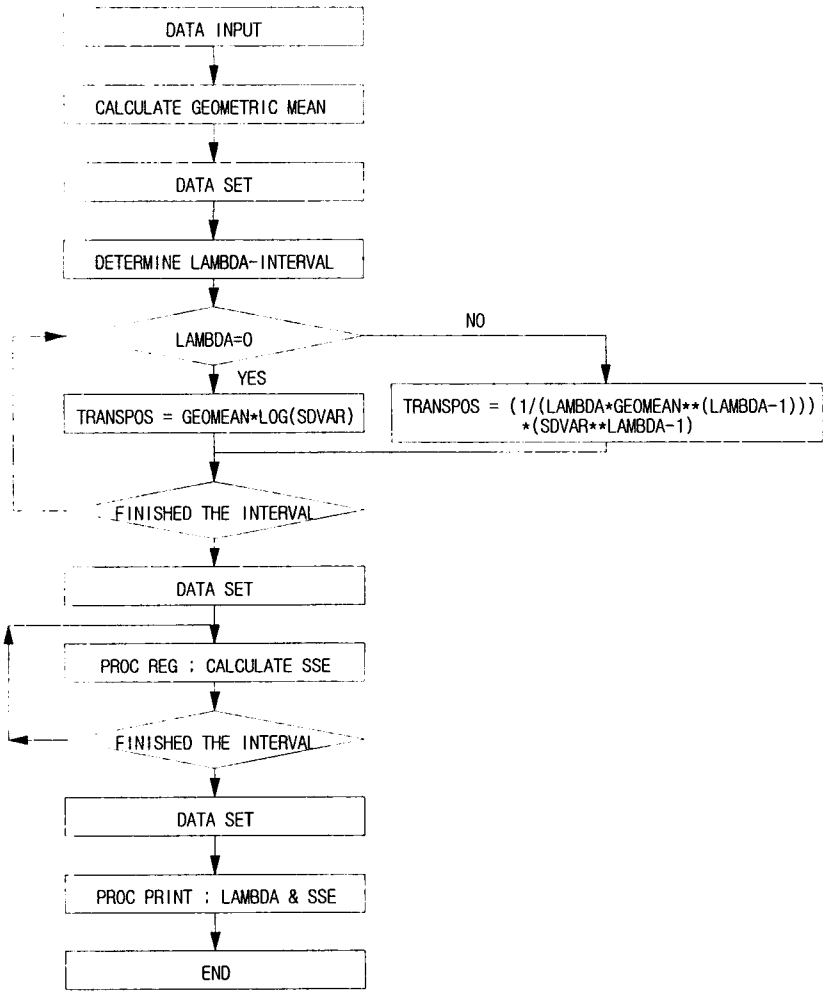
Fig. 4 Scatter Plot of Transformed Data

References

1. Montgomery, D.C. and Peck, E.A., *Introduction to Linear Regression Analysis,* New York: Wiley & Sons, 1982.
2. Seber, G.A.F., *Linear Regression Analysis*, New York: Wiley & Sons, 1977.
3. Box, G.E.P. and Cox, D.R., "An Analysis of Transformations," *Journal of Royal Statistical Society* B, 26(1964), 211-246.
4. Schlesselman, J., "Power Families: A Note on the Box-Cox Transformation," *Journal of Royal Statistical Society* B, 33(1971), 307-311.
5. Cook, R.D. and Weisberg, S., *Residuals and Influence in Regression*, London: Chapman and Hall, 1982.
6. Logothetis, N., "Box-Cox Transformations and the Taguchi Method," *Applied Statistics* 39, No. 1(1990), 31-48.
7. Chatterjee S. and Price, B., *Regression Analysis by Examples,* New York: Wiley & Sons, 1977.
8. Fearn, T., "Box-Cox Transformations and the Taguchi Method: An Alternative Analysis of a Taguchi Case Study," *Applied Statistics* 41, No. 3(1992), 553-559.

Appendix

1) Flow chart for algorithm



2) SAS program for algorithm

```
OPTIONS LINESIZE=80 PAGESIZE=60 ;
FILENAME OLDDATA ' A:\??LINEAR.DAT ' ;
/* using NOLINEAR.DAT if we want the automatic selection of λ in the nonlinear situation */
/* using LINEAR.DAT if we want the automatic selection of λ in the linear situation      */
DATA RAWDATA0 ;
        INFILE OLDDATA ;
        INPUT ID_NUM  SPECIES $ FTVAR  SDVAR ;
        LABEL ID_NUM=' OBSERVATION ' FTVAR=' FIRSTVARIABLE '
              SDVAR=' SECONDVARIABLE ' ;

/* calculate geometric mean by program GEOMEAN1 through GEOMEAN4 */
DATA GEOMEAN1 (KEEP=GEOMEAN) ;
        SET RAWDATA0 ;
        RETAIN AMOUNT 1 ;
        AMOUNT=AMOUNT * SDVAR ;
        GEOMEAN=AMOUNT ** (1/_N_) ;
        FILE ' A:\GEOMEAN1 ' ;
        PUT GEOMEAN ;

DATA GEOMEAN2 ;
        INFILE ' A:\GEOMEAN1 ' ;
        INPUT ROW1-ROW? ;

DATA GEOMEAN3 (DROP=I) ;
        SET GEOMEAN2 ;
        ARRAY COLUMN{?} ROW1-ROW? ;
     DO I = 1 TO ? ;
        COLUMN{I}=ROW? ;
        FILE ' A:\GEOMEAN3 ' ;
        PUT ROW? ;
     END ;
```

```
DATA GEOMEAN4 ;
        INFILE ' A:\GEOMEAN3 ' ;
        INPUT GEOMEAN ;

/*  sort all λ values and calculate sse by program RAWDATA0 and RAWDATA2 */
DATA RAWDATA1 ;
        MERGE RAWDATA0 GEOMEAM4 ;
        DIVISION=?? ;       /* ??=5, 7, 10, 12, 15 in this program */
     DO BEGIN=-DIVISION TO DIVISION BY 1 ;
        LAMBDA= BEGIN / DIVISION ;
     IF LAMBDA=0 THEN TRANSPOS=GEOMEAN*LOG(SDVAR) ;
        ELSE TRANSPOS=(1/(LAMBDA*GEOMEAN**(LAMBDA-1)))*(SDVAR**LAMBDA-1) ;
        OUTPUT ;
     END ;

PROC SORT ; BY LAMBDA ;
PROC REG DATA=RAWDATA1 OUTEST=RAWDATA2 NOPRINT;
        MODEL TRANSPOS = FTVAR / SELECTION=RSQUARE SSE ;
        BY LAMDBA ;

/* print all λ values and automatically select λ minimizing sse by program outdata*/
DATA OUTDAT (KEEP=LAMDA  _SSE_ ) ;
         SET RAWDATA2 ;

PROC PRINT DATA=OUTDAT ;
         RUN ;
```