Analysis of transformations

Project 1
Regression Analysis

Amal Agarwal
Roll. no.: 09D11001

Sagar Chordia
Roll. no.: 09005013

Tuhin Sarkar
Roll. no.: 09007030

Advisor: Prof. Siuli Mukhopadhyay

Department of Mathematics
Indian Institute of Technology, Bombay
2013

Chapter 1

Review of Box-Cox Transformations

1.1 Introduction

The usual techniques for linear regression are justified by the following underlying assumptions:

- simplicity of structure of E(y)
- constancy of error variance
- normality of distributions
- independence of observations

However, sometimes these assumptions may not hold. The motivation behind this paper is to derive transformations such that these not only justify these assumptions but rather find, wherever possible, a metric in terms of which the finding may be succinctly expressed.
Tukey (1949, 1950) used transformations to achieve additive in the analysis of variance. To stabilize the variance, the usual method of transformation by Bartlett (1947) is used where the relation between mean and variance is determined. Anscombe (1961) and Tukey and Anscombe (1983) have employed analysis of residuals to detect departures from standard assumptions and have indicated how transformations might be devised from functions of the residuals. There are many problems where both dependent and independent variables are transformed. Box and Tidwell (1962) transformation can be employed without affecting the constancy of variance and normality of error distribution. It is useless to try to linearize a relation which is not monotonic, but a transformation is sometimes useful in such cases.

1.2 General Idea

The general idea is to study transformed variables y^λ indexed by an unknown parameter λ, estimate λ and other parameter of our model. There are two primary lines of analyses. First is where the particular λ is of interest. Second is where we study the factor effects of a choice of λ.
The important family of transformations discussed here are:

$$y^\lambda = \frac{y^\lambda - 1}{\lambda}$$

when $\lambda \neq 0$

$$y^\lambda = \log(y)$$

when $\lambda = 0$

Now this transformation is valid only when $y \geq 0$. There is also a second set of transformations which hold for $y \geq \lambda_2$

$$y^\lambda = \frac{(y+\lambda_2)^{\lambda_1} - 1}{\lambda_1}$$

when $\lambda_1 \neq 0$

$$y^\lambda = \log(y+\lambda_2)$$

when $\lambda_1 = 0$

But the second transformation is nothing but a linear transformation of the first one. Therefore we can analyze the first one itself.

The basic idea is that once we have the transformation we can use our general linear regression model:

$$E[y^\lambda] = a\theta$$

where a is the independent variable matrix and θ are our parameters.

There are two approaches to find λ:

- Use the maximum likelihood method
- Use a Bayesian approach

In the first method we use the likelihood function:

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

$$J(\lambda; y) = \Pi_{i=1}^n \left(\frac{dy_i^\lambda}{dy_i}\right)$$

Note that $J(\lambda; y)$ is the Jacobian of transformation from y to y^λ. In this first, we apply "orthodox" large-sample maximum likelihood theory. This approach leads to point estimates of the parameters and to approximate tests and confidence intervals based on chi-squared distribution.

Now, we can use the normal linear regression techniques on the transformed dependent variables, where we get:

$$\hat{\sigma^2} = y^{\lambda'} a_r y^\lambda / n$$

$$a_r = I - a(a'a)^{-1}a'$$

The log-likelihood function looks like:

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

To get λ from this, plot the maximized log-likelihood function for a series of trial λ values. From the plot the maximizing value of λ can be taken as:

$$L_{max}(\hat{\lambda}) - L_{max}(\lambda) < \frac{1}{2}\chi_\nu^2(\alpha)$$

Here ν is the number of degrees of freedom of λ

The above results can be simplified by using the further transformation given by:

$$z^\lambda = y^\lambda / J^{1/n}$$

We now discuss how to approach the problem of finding λ using the Bayesian analysis. Consider

$$p(y|\theta,\sigma^2) = \frac{1}{(2\pi)^{n/2}\sigma^n}\exp\{-\frac{\nu s^2(\lambda) + (\theta-\hat{\theta_\lambda})'a'a(\theta-\hat{\theta_\lambda})}{2\sigma^2}\}J(\lambda;y)$$

Where $\hat{\theta_\lambda}$ is the least square estimate of θ for given λ.
In this method we use the Bayes theorem to get the posterior distribution of λ by using the following:

$$I(\lambda|y) = \int_{-\infty}^{\infty} d(\log\sigma)\int_{-\infty}^{\infty} d\theta p(y|\theta,\sigma^2)$$

$$p(\lambda|y) = K'_y\frac{I(\lambda|y)p_0(\lambda)}{\{J(\lambda;y)\}^{(n-\nu)/n}}$$

Note K'_y is a normalizing constant. Here is also assumed that θ and $\log\sigma$ are effectively uniform over the range where likelihood is appreciable. By taking $p_0(\lambda)$ as uniform (prior distribution), we maximize the posterior distribution. That is required λ.

1.3 Further Analysis of Transformations

This analysis indicates the following things:

- how simple a model we are justified in using
- what weight is given to considerations (simple expectation structure, constant variance and normal distributions) in choosing λ
- whether different transformations are really needed

The basic ideas behind this analysis is to use the two approaches as before. We first discuss the log-likelihood method. Consider the constraint C. Now:

$$L_{max}(\lambda|C) = L_{max}(\lambda) + \{L_{max}(\lambda|C) - L_{max}(\lambda)\}$$

The two terms are examined separately. Similarly when more number of constraints are imposed.
Now in the Bayesian case:

$$p(\lambda|C) = p(\lambda)\frac{p(C|\lambda)}{p(C)}$$

Chapter 2

Simulation

2.1 Aim

For non-trivial application of Box-cox transformation, data should not satisfy all properties of Linear normal regression model. So we consider various datasets as cases where one or more assumptions are violated.

2.2 Box-Cox transformation

We apply box-cox transformation on given dataset. To estimate λ we use maximum likelihood method. Note X is generated from uniform distribution and hence rank of X is most of times equal to N number of data points. Our choice of dataset ensures that we can use maximum likelihood method.

On evaluation of concepts in section 2 it can be shown that $\widehat{\lambda}$ can be found by minimizing following function

$$RSS(V(\lambda)) = [V(\lambda) - \widehat{V}(\lambda)]^t [V(\lambda) - \widehat{V}(\lambda)]$$

where

$$\widehat{V}(\lambda) = X(X^tX)^{-1}X^tV(\lambda)$$

$$V(\lambda) = \begin{bmatrix} V_1(\lambda) \\ V_2(\lambda) \\ \vdots \\ V_n(\lambda) \end{bmatrix}$$

$$V_i(\lambda) = (Y_i^\lambda - 1)/\lambda \overbrace{Y}^{\lambda-1}, \lambda \neq 0$$

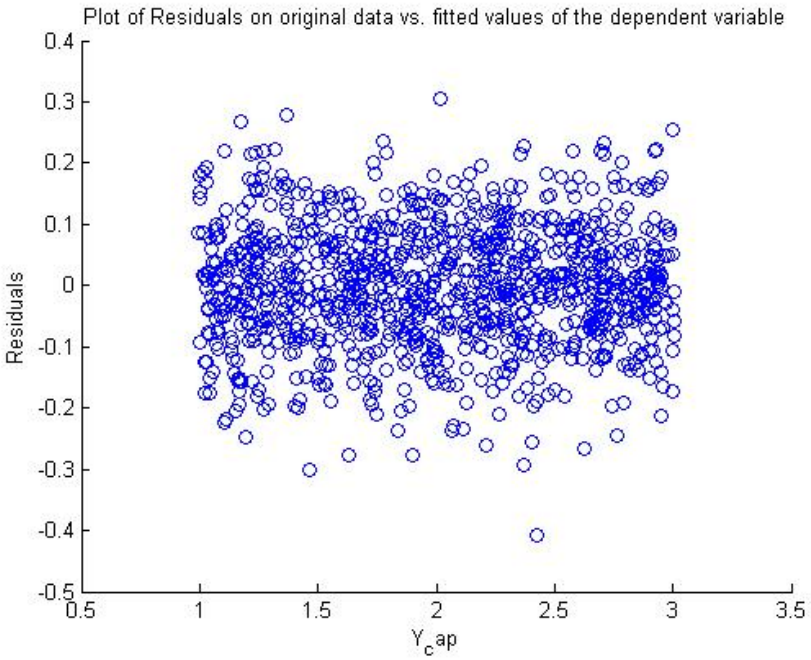
$$V_i(\lambda) = \overbrace{Y} log(Y_i), \lambda = 0$$

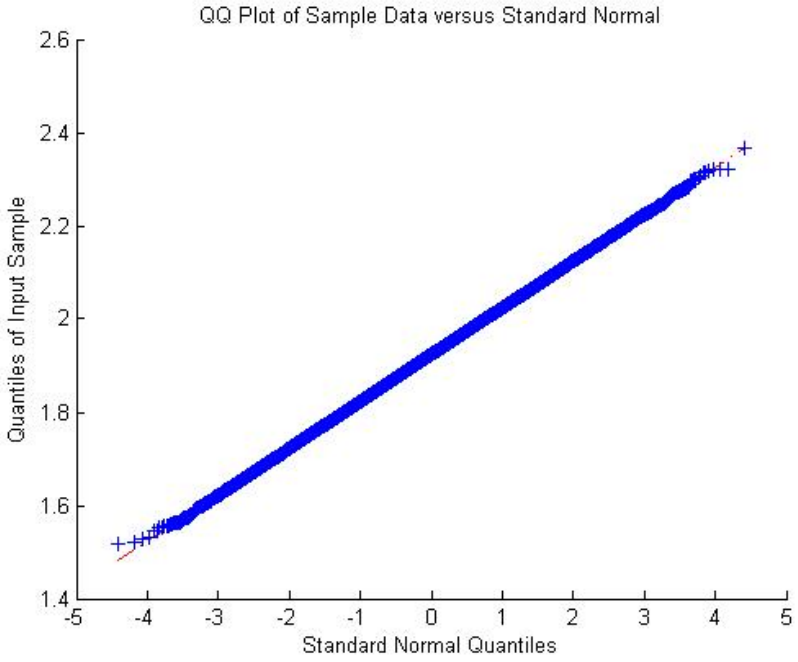
$$\overbrace{Y} = (Y_1 Y_2 \cdots Y_n)^{1/N}$$

2.3 Analysis of the Simulation Results

2.3.1 Normal linear data with constant variance over X

This dataset satisfies all conditions of linear regression. For this data we plot graph of semi studentized residuals vs X value and QQplot for data. We also perform lack of fit test of both data sets.

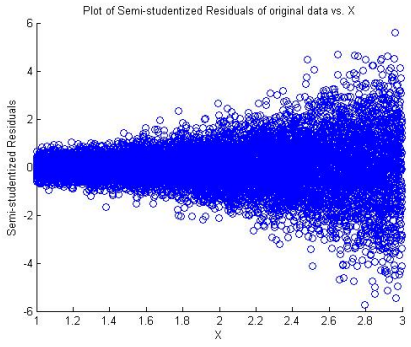


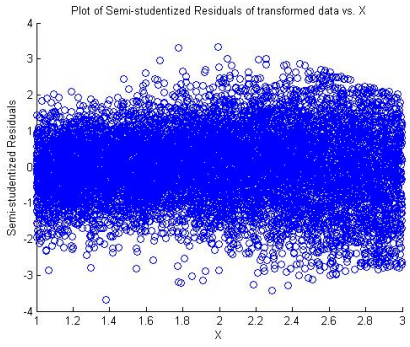


As we can see residual plot is spread uniformly across X over range of Y. QQplot is linear and lack of fit test confirms that model is linear. Thus when we perform box-cox transformation on data which doesn't satisfy some of the linear regression conditions should return graphs similar to above one in ideal case. The estimated parameters of model are:
β_0=1
β_1=2

2.3.2 Normal linear data with heteroscedasticity

As we can see in residual plot the variance changes as X changes. We plot residual graph (semi-studentized residual vs X) of original and transformed data.



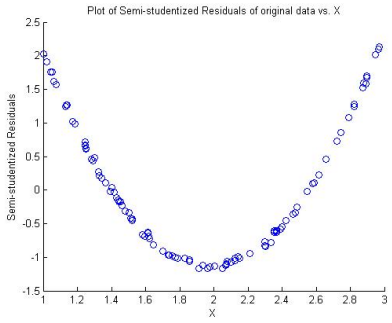


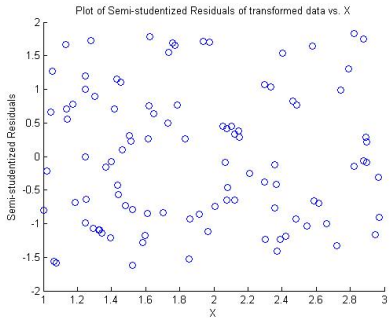
From the uniform span of residual plot in transformed graph, it is clear that box-cox transformation is helpful in removing the heteroscedasticity from the data. The estimated parameters of model are:
β_0=-2.13
β_1=7.6
λ_{max}=0.34

2.3.3 Normal non-linear data with constant variance over X

Here we have used some non-linear function to generate Y with respect to X. Here we have used quadratic function. Again we plot residual graph (semi-studentized residual vs X) for original and transformed data.

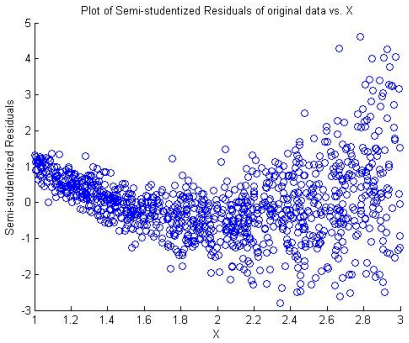
From the uniform span of residual plot in transformed graph, it is clear that box-cox transformation is helpful in removing the non-linearity from the data. The estimated parameters of model are:
β_0=50.4
β_1=4.5
λ_{max}=0.48

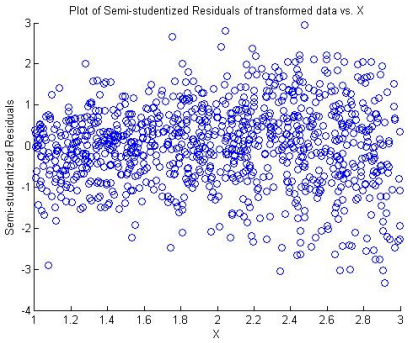




2.3.4 Normal non-linear data with heteroscedasticity

Original data follows quadratic equation with heteroscedasticity of variance. We plot residual graph (semi-studentized residual vs X) for original and transformed data.





From the uniform span of residual plot in transformed graph, it is clear that box-cox transformation is helpful in removing the non-linearity and heteroscedasticity simultaneously. The estimated parameters of model are:
β_0=-1.27
β_1=7.4
λ_{max}=0.29

The matlab code for this case is given in next chapter.

2.4 Outlier detection and removal

To remove outliers we use semi-studentized graph. All data points whose semi-studentized variance is greater than +3 or less than -3 we term them as outliers and remove them from original dataset. Then we perform the box-cox transformation for transformed modified data and display its semi-studentized residual plot again.

2.5 Lack of Fit test

We did lack of fit test for Case 3 and Case 4 to check linearity of transformed data. The test results were negative for the initial model but after transformation it gave positive results thus justifying the aim of the transformation. We did hypothesis test for $\beta_0 = 0$ and $\beta_1 = 0$ vs.

corresponding alternatives for 95 percent confidence coefficient that is $\alpha = 0.05$ Calculated t-statistic in each case was mush greater leading to rejection of null hypothesis. Thus we conclude $\beta_0 \neq 0$ and $\beta_1 \neq 0$ after transformation.

2.6 Unresolved case of Non-normality

We performed tranformations on non-normal data with different ditributions like chi-square, t, F and beta. However in each case the qq plot for the transformed data was devoid of any improvement compared to the original data thus refuting the assumption given in the paper that Box-Cox transformation can be used to make non-normal data again normal.

2.7 Conclusions

This paper gives an idea of a transformation that can be employed to modify dependent variables so that they are workable (standard linear regression techniques can be applied). This can be used for removing non-linearity and heteroscedasticity from the data as seen from the Simulation results. Unfortuneately this fails to make non-normal data again normal. At the end of the paper there is also some discussion as to how we can also use a transformation where even the independent variables are modified.

Chapter 3

Matlab Code

clear all; clc;

n = 1000; alpha = 0.05; $X_1 = ones(n,1); X_2 = 1 + 2 * rand(n,1); X = [X_1, X_2];$

beta0 = 1; beta1 = 2;

var = 0.1*exp(X_2); $er = normrnd(0, var, n, 1);$

Y = beta0 + beta1*(X_2.^{2}) + er;

[b,bint,r,rint,stats] = regress(Y,X); $Y_cap = b(1) + b(2) * (X_2);$

rs = r/(sqrt(stats(4))); figure, scatter(X_2, rs); $title('PlotofSemi-studentizedResidualsoforiginalda$
$studentizedResiduals');$

m1 = 10; y = n+1; $er_lf = zeros(m1,(y-1)); Y_lf = zeros(m1,(y-1)); Ymean_lf =$
$zeros(1,(y-1)); YSSPE_lf = zeros(m1,(y-1)); YSSE_lf = zeros(m1,(y-1)); forj = 1:$
$(y-1)fori = 1:m1er_lf(i,j) = normrnd(0,var(j)); Y_lf(i,j) = beta0 + beta1 * (X_2(j)) +$
$er_lf(i,j); endYmean_lf(j) = mean(Y_lf(:,j)); fori = 1:m1YSSPE_lf(i,j) = (Y_lf(i,j) -$
$Ymean_lf(j))^2; YSSE_lf(i,j) = (Y_lf(i,j) - (Y_cap(j)))^2; endendSSPE = sum(sum(YSSPE_lf)); SSE =$
$sum(sum(YSSE_lf)); SSLF = SSE - SSPE; MSLF = SSLF/((y-1)-2); MSPE =$
$SSPE/((m1*(y-1))-(y-1)); F_star = MSLF/MSPE; F = finv((1-alpha),((y-1)-$
$2),((m1*(y-1))-(y-1))); ifF_star > Fdisp('RegressionFunctionisnotlinearfororiginalmodel'); elseda$

k = 0; for i = 1:n if (abs(rs(i))¿3) k = [i,k]; end end

s = length(k); km = zeros(1, s-1); for i = 1:s-1 km(i) = k(i); end

$X_{2t}emp1 = X_2; var_temp1 = var; er_temp1 = er; Y_lf_temp1 = Y_lf; fori = 1:nforj =$
$1:s-1if(km(j) == i)X_{2t}emp1(i) = 0; var_temp1(i) = 0; er_temp1(i) = 0; Y_lf_temp1(:,i) =$
$zeros(m1,1); endendend$

$X_{2t}emp2 = 0; var_temp2 = 0; er_temp2 = 0; Y_lf_temp2 = zeros(m1,1); forp = 1:nif(X_{2t}emp1(p) =$
$0)X_{2t}emp2 = [X_{2t}emp1(p); X_{2t}emp2]; endif(var_temp1(p) = 0)var_temp2 = [var_temp1(p); var_temp2]; en$
$0)er_temp2 = [er_temp1(p); er_temp2]; endif(Y_lf_temp1(:,p) = zeros(m1,1))Y_lf_temp2 = [Y_lf_temp1(:$
$,p), Y_lf_temp2]; endend$

y = length($X_{2t}emp2$); $X_{2m} = zeros((n-s+1),1); var_m = zeros((n-s+1),1); er_m =$
$zeros((n-s+1),1); Y_lf_m = zeros(m1,(n-s+1)); fori = 1:y-1X_{2m}(i) = X_{2t}emp2(i); var_m(i) =$
$var_temp2(i); er_m(i) = er_temp2(i); Y_lf_m(:,i) = Y_lf_temp2(:,i); end$

$X_{1m} = ones(y-1,1); X_m = [X_{1m}, X_{2m}];$

$Y_m = beta0 + beta1 * (X_{2m}.^2) + er_m;$

m2 = 100000; mid = round((y-1)/2); $er_d = normrnd(0, var(mid), m2, 1); Y_d = beta0 +$
$beta1 * (X_2(mid)) + er_d; figure, qqplot(Y_d);$

GM = geomean(Y_m); $V = @(lambda)(((Y_m.^lambda) - 1)./(lambda * (GM^(lambda -$
$1)))); V_cap = @(lambda)(X_m*(inv(X'_m*X_m))*(X'_m)*V(lambda)); RSS = @(lambda)((V(lambda)-$
$V_cap(lambda))'*(V(lambda)-V_cap(lambda))); lambda_max = fminbnd(RSS,-100,100); W =$

$V(lambda_max); W_d = (((Y_d.^lambda_max) - 1)./(lambda_max * (GM^(lambda_max - 1)))); W_lf =$
$(((Y_lf_m.^lambda_max) - 1)./(lambda_max * (GM^(lambda_max - 1))));$

$[b_t, bint_t, r_t, rint_t, stats_t] = regress(W, X_m); W_cap = b_t(1) + b_t(2) * (X_{2m}); rs_t = r_t/(sqrt(stats_t(4)));$

figure, scatter(X_{2m}, rs_t); $title('Plot of Semi - studentized Residuals of transformed data vs. X'); xlabel$
$studentized Residuals');$

figure, qqplot(W_d);

$Wmean_lf = zeros(1, (y-1)); WSSPE_lf = zeros(m1, (y-1)); WSSE_lf = zeros(m1, (y-1)); for j = 1 : (y-1) Wmean_lf(j) = mean(W_lf(:, j)); for i = 1 : m1 WSSPE_lf(i, j) = (W_lf(i, j) - Wmean_lf(j))^2; WSSE_lf(i, j) = (W_lf(i, j) - (W_cap(j)))^2; end end SSPE_t = sum(sum(WSS$
$sum(sum(WSSE_lf)); SSLF_t = SSE_t - SSPE_t; MSLF_t = SSLF_t/((y-1) - 2); MSPE_t =$
$SSPE_t/((m1 * (y-1)) - (y-1)); F_star_t = MSLF_t/MSPE_t; F_t = finv((1 - alpha), ((y-1) -$
$2), ((m1 * (y-1)) - (y-1))); if F_star_t > F_t disp('Regression Function is not linear after transformation');$ e

Bibliography

[1] G.E.P. Box and D.R. Cox, An Analysis of Transformations, Journal of the Royal Statistical Society. Series B (Methodological), Vol.26, No.2 (1964), pp.211-252

[2] MATLAB, version 7.10.0.499 (R2010a)