The Stata Journal (2010)
10, Number 1, pp. 69–81

Power transformation via multivariate Box–Cox

Charles Lindsey
Texas A & M University
Department of Statistics
College Station, TX
lindseyc@stat.tamu.edu

Simon Sheather
Texas A & M University
Department of Statistics
College Station, TX
sheather@stat.tamu.edu

Abstract. We present a new Stata estimation program, `mboxcox`, that computes the normalizing scaled power transformations for a set of variables. The multivariate Box–Cox method (defined in Velilla, 1993, *Statistics and Probability Letters* 17: 259–263; used in Weisberg, 2005, *Applied Linear Regression* [Wiley]) is used to determine the transformations. We demonstrate using a generated example and a real dataset.

Keywords: st0184, mboxcox, mbctrans, boxcox, regress

1 Theory and motivation

Box and Cox (1964) detailed normalizing transformations for univariate y and univariate response regression using a likelihood approach. Velilla (1993) formalized a multivariate version of Box and Cox's normalizing transformation. A slight modification of this version is considered in Weisberg (2005), which we will use here.

The multivariate Box–Cox method uses a separate transformation parameter for each variable. There is also no independent/dependent classification of the variables. Since its inception, the multivariate Box–Cox transformation has been used in many settings, most notably linear regression; see Sheather (2009) for examples. When variables are transformed to joint normality, they become approximately linearly related, constant in conditional variance, and marginally normal in distribution. These are very useful properties for statistical analysis.

Stata currently offers several versions of Box–Cox transformations via the `boxcox` command. The multivariate options of `boxcox` are limited to regression settings where at most two transformation parameters are allowed. We present the `mboxcox` command as a useful complement to `boxcox`. We will start by explaining the formal theory of what `mboxcox` does.

First, we define a scaled power transformation as

$$\psi_s(y,\lambda) = \begin{pmatrix} \frac{y^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0 \\ \log y & \text{if } \lambda = 0 \end{pmatrix}$$

Scaled power transformations preserve the direction of associations that the transformed variable had with other variables. So scaled power transformations will not switch collinear relationships of interest.

Next, for n-vector $\mathbf{x}$, we define the geometric mean: $\mathrm{gm}(\mathbf{x}) = \exp\left(1/n \sum_{i=1}^{n} \log x_i\right)$.

Suppose the random vector $\mathbf{x} = (x_1, \ldots, x_p)'$ takes only positive values. Let $\Lambda = (\lambda_1, \ldots, \lambda_p)$ be a vector of real numbers, such that $\{\psi_s(x_1, \lambda_1), \ldots, \psi_s(x_p, \lambda_p)\}$ is distributed $N(\mu, \Sigma)$.

Now we take a random sample of size n from the population of $\mathbf{x}$, yielding data $X = (\mathbf{x}_1, \ldots, \mathbf{x}_p)$. We define the transformed version of the variable X_{ij} as ${X_{ij}}^{(\lambda_j)} = \psi_s(X_{ij}, \lambda_j)$. This yields the transformed data matrix $X^{(\Lambda)} = \left\{{\mathbf{x}_1}^{(\lambda_1)}, \ldots, {\mathbf{x}_p}^{(\lambda_p)}\right\}$.

Finally, we define the normalized transformed data:

$$Z^{(\Lambda)} = \left\{\mathrm{gm}(\underline{x}_1)^{\lambda_1} {\underline{x}_1}^{(\lambda_1)}, \ldots, \mathrm{gm}(\underline{x}_p)^{\lambda_p} {\underline{x}_p}^{(\lambda_p)}\right\}$$

Velilla (1993, eq. 3) showed that the concentrated log likelihood of Λ in this situation was given by

$$L_c(\Lambda) = -\frac{n}{2} \log \left| {Z^{(\Lambda)}}' \left(I_n - \frac{1_n 1_n'}{n}\right) Z^{(\Lambda)} \right|$$

Weisberg (2005) used modified scaled power transformations rather than plain scaled power transformations for each column of the data vector.

$$\psi_m(y_i, \lambda) = \mathrm{gm}(\mathbf{y})^{1-\lambda} \psi_s(y_i, \lambda)$$

Under a modified scaled power transformation, the scale of the transformed variable is invariant to the choice of transformation power. So the scale of a transformed variable is better controlled under the modified scaled power transformation than under the scaled power transformation. Inference on the optimal transformation parameters should be similar under both scaled and modified scaled methods. The transformed data under a scaled power transformation is equivalent to the transformed data under an unscaled power transformation with an extra location/scale transformation. A multivariate normal random vector yields another multivariate normal random vector when a location/scale transformation is applied to it. So the most normalizing scaled transformation essentially yields as normalizing a transformation as its unscaled version. We thus expect great similarity between the optimal scaled, modified scaled, and unscaled parameter estimates.

The new concentrated likelihood (Weisberg 2005, 291, eq. A.36) is

$$L_c(\Lambda) = -\frac{n}{2} \log \left| {Z_*}^{(\Lambda)'} \left(I_n - \frac{1_n 1_n'}{n}\right) {Z_*}^{(\Lambda)} \right|$$

Here $Z^{(\Lambda)}$ has been replaced by the actual transformed data.

$${Z_*}^{(\Lambda)} = \left\{\mathrm{gm}(\underline{x}_1)^{1-\lambda_1} {\underline{x}_1}^{(\lambda_1)}, \ldots, \mathrm{gm}(\underline{x}_p)^{1-\lambda_p} {\underline{x}_p}^{(\lambda_p)}\right\}$$

In terms of the sample covariance of ${Z_*}^{(\Lambda)}$, $L_c(\Lambda)$ is a simple expression. In terms of Λ, it is very complicated. The `mboxcox` command uses $L_c(\Lambda)$ to perform inference on Λ, where the elements of Λ are modified scaled power transformation parameters. Because of the complexity of $L_c(\Lambda)$, a numeric optimization is used to estimate Λ. The second derivative of $L_c(\Lambda)$ is computed numerically during the optimization, and this yields the covariance estimate of Λ.

We should take note of the situation in which the data does not support a multivariate Box–Cox transformation. Problems in data collection may manifest as outliers. As Velilla (1995) states, "it is well known that the maximum likelihood estimates to normality is very sensitive to outlying observations." Additionally, the data or certain variables from it could simply come from a nonnormal distribution. Unfortunately, the method of transformation we use here is not sensitive to these problems. Our method of Box–Cox transformation is not robust. For methods that are robust to problems like these, see Velilla (1995) and Riani and Atkinson (2000). We present the basic multivariate Box–Cox transformation here, as a starting point for more robust transformation procedures to be added to Stata at a later date.

2 Use and a generated example

The `mboxcox` command has the following basic syntax:

`mboxcox` *varlist* [*if*] [*in*] [`, level(#)`]

Like other estimation commands, the results of `mboxcox` can be redisplayed with the following simpler syntax:

`mboxcox` [`, level(#)`]

The syntax of `mboxcox` is very simple and straightforward. We also provide the `mbctrans` command to create the transformed variables. This command is used to streamline the data transformation process. It takes inputs of the variables to be transformed and a list of transformation powers, and saves the transformed variables under their original names with a `t_` prefix. The command supports unscaled, scaled, and modified scaled transformations. Accomplish scaled transformations by specifying the `scale` option. To obtain modified scaled transformations, specify the `mscale` option.

`mbctrans` *varlist* [*if*] [*in*] [`, power(`*numlist*`) mscale scale`]

We generate 10,000 samples from a three-variable multivariate normal distribution with means $(10, 14, 32)$ and marginal variances $(1, 3, 2)$. The first and second variables are correlated with a covariance of 0.3.

```
. set obs 10000
obs was 0, now 10000
. set seed 3000
```

```
. matrix Means = (10,14,32)
. matrix Covariance = (1,.3,0)\(.3,3,0)\(0,0,2)
. drawnorm x1 x2 x3, means(Means) cov(Covariance)
. summarize
```

Variable	Obs	Mean	Std. Dev.	Min	Max
x1	10000	10.00191	.9943204	5.42476	13.72735
x2	10000	13.9793	1.713186	7.683866	21.38899
x3	10000	31.98648	1.41477	26.26886	38.04641

Next we transform the data using unscaled power transformations $(2, -1, 3)$. Note that the correlation direction between the first and second variable changes.

```
. mbctrans x1 x2 x3, power(2 -1 3)
. correlate t_x1 t_x2
(obs=10000)
```

	t_x1	t_x2
t_x1	1.0000	
t_x2	-0.1585	1.0000

We will use `mboxcox` to determine the optimal modified scaled power transformation estimates for normalizing the transformed data. The optimal unscaled power transformation vector is $(1/2, -1, 1/3)$, each element being the inverse of the variable's original transformation power.

```
. mboxcox t_x1-t_x3
Multivariate boxcox transformations
                                                Number of obs   =      10000
Likelihood Ratio Tests
```

Test	Log Likelihood	Chi2	df	Prob > Chi2
All powers -1	-67280.73	2078.173	3	0
All powers 0	-66461.51	439.7275	3	0
All powers 1	-66837.99	1192.704	3	0

	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
lambda						
t_x1	.5318023	.0402718	13.21	0.000	.452871	.6107336
t_x2	-.9715714	.065297	-14.88	0.000	-1.099551	-.8435915
t_x3	.3647025	.0613916	5.94	0.000	.2443772	.4850278

We find that the modified scaled transformation parameter estimates of `mboxcox` are close to the unscaled parameters. The postestimation features of `mboxcox` tell us that there is no evidence to reject the assertion that the optimal modified scaled transformation parameters are identical to the unscaled parameters. This correspondence between modified scaled and unscaled is not surprising, as we detailed in the last section.

```
. test (t_x1= .5) (t_x2= -1) (t_x3 = 1/3)
 ( 1)  [lambda]t_x1 = .5
 ( 2)  [lambda]t_x2 = -1
 ( 3)  [lambda]t_x3 = .3333333
           chi2(  3) =    1.08
         Prob > chi2 =    0.7831
```

3 Real example

Sheather (2009) provides an interesting dataset involving 2004 automobiles. We wish to perform a regression of the variable `highwaympg` on the predictors `enginesize`, `cylinders`, `horsepower`, `weight`, `wheelbase`, and the dummy variable `hybrid`.

```
. use cars04, clear
. summarize highwaympg enginesize cylinders horsepower weight wheelbase hybrid
```

Variable	Obs	Mean	Std. Dev.	Min	Max
highwaympg	234	29.39744	5.372014	19	66
enginesize	234	2.899145	.925462	1.4	5.5
cylinders	234	5.517094	1.471374	3	12
horsepower	234	199.7991	64.03424	73	493
weight	234	3313.235	527.0081	1850	4474
wheelbase	234	107.1154	5.82207	93	124
hybrid	234	.0128205	.1127407	0	1

```
. regress highwaympg enginesize cylinders horsepower weight wheelbase hybrid
```

Source	SS	df	MS		
Model	5343.19341	6	890.532235	Number of obs =	234
Residual	1380.84505	227	6.08301785	F(6, 227) =	146.40
Total	6724.03846	233	28.8585342	Prob > F =	0.0000
				R-squared =	0.7946
				Adj R-squared =	0.7892
				Root MSE =	2.4664

highwaympg	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
enginesize	.166796	.5237721	0.32	0.750	-.8652809	1.198873
cylinders	-.1942966	.3171983	-0.61	0.541	-.8193262	.4307331
horsepower	-.0182825	.0052342	-3.49	0.001	-.0285963	-.0079687
weight	-.00662	.0007513	-8.81	0.000	-.0081003	-.0051397
wheelbase	.1797597	.0570666	3.15	0.002	.0673117	.2922078
hybrid	20.33805	1.468368	13.85	0.000	17.44467	23.23142
_cons	36.05649	4.726131	7.63	0.000	26.7438	45.36919

The model is not valid. It has a number of problems. Nonconstant variance of the errors is one. As explained in Sheather (2009), this problem can be detected by graphing the square roots of the absolute values of the standardized residuals versus the fitted values and continuous predictors. Trends in these plots suggest that the variance changes at different levels of the predictors and fitted values. We graph these plots and see a variety of increasing and decreasing trends.

```
. predict rstd, rstandard
. predict fit, xb
. generate nsrstd = sqrt(abs(rstd))
. local i = 1
. foreach var of varlist fit enginesize cylinders horsepower weight wheelbase {
  2. twoway scatter nsrstd `var´ || lfit nsrstd `var´,
> ytitle("|Std. Residuals|^.5") legend(off)
> ysize(5) xsize(5) name(gg`i´) nodraw
  3. local i = `i´ + 1
  4. }
. graph combine gg1 gg2 gg3 gg4 gg5 gg6, rows(2) ysize(10) xsize(15)
```

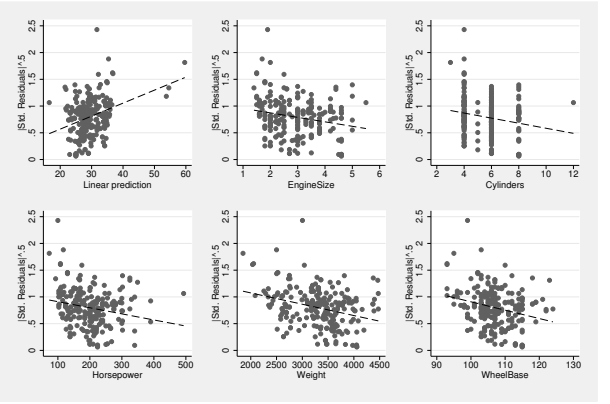

Figure 1. $\sqrt{|\,\text{Standard residuals}\,|}$ versus predictors and fitted values.

Data transformation would be a strategy to solve the nonconstant variance problem. As suggested in Weisberg (2005, 156), we should first examine linear relationships among the predictors. If they are approximately linearly related, we can use the fitted values to find a suitable transformation for the response, perhaps through an inverse response plot (Sheather 2009). A matrix plot of the response and predictors shows that we will not be able to do that. Many appear to share a monotonic relationship, but it is not linear.

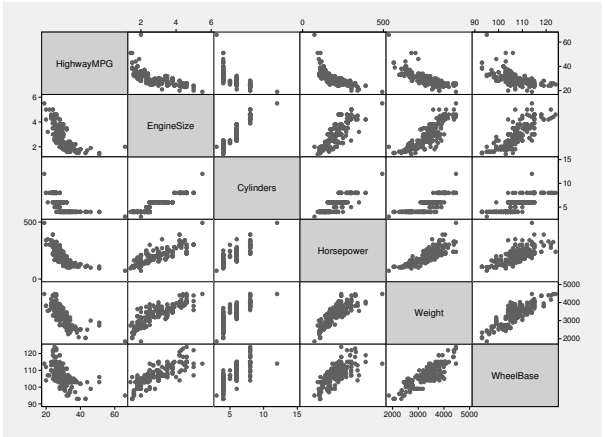

Figure 2. Matrix plot original response and predictors.

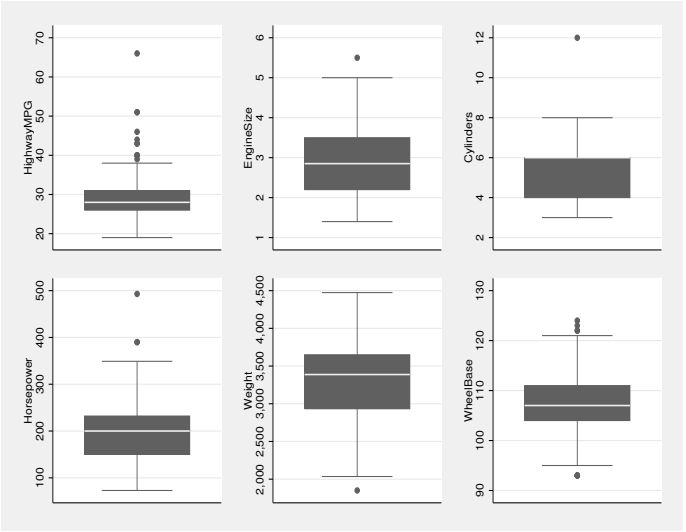

Figure 3. Box plots original response and predictors.

In addition, a look at the box plots reveals that several of the predictors and the response are skewed. The data are not consistent with a multivariate normal distribution. If the predictors and response were multivariate normal conditioned on the value of `hybrid`, then it would follow that the errors of the regression would have constant variance. The conditional variance of multivariate normal variables is always constant with regard to the values of the conditioning variables.

There are actually only three observations of `hybrid` that are nonzero. Data analysis not shown here supports the contention that `hybrid` only significantly affects the

location of the joint distribution of the remaining predictors and response. Successful inference on other more complex properties of the joint distribution, conditional on `hybrid` = 1, would require more data. Hence, we ignore the value of `hybrid` in calculating a normalizing transformation. In the first section, we mentioned that outliers could be a serious problem for our method. Our approach here could lead to outliers that would cause the transformation to fail.

If the marginal transformation that we estimate is suitably equivalent to the transformations obtained by conditioning on `hybrid` and approximately normalizes the other predictors and the response, then the errors of the regression will be at least approximately constant and its predictors and response more symmetric.

```
. mboxcox enginesize cylinders horsepower highwaympg weight wheelbase
Multivariate boxcox transformations
                                                Number of obs   =       234
Likelihood Ratio Tests
```

Test	Log Likelihood	Chi2	df	Prob > Chi2
All powers -1	-2431.978	202.6359	6	0
All powers 0	-2369.889	78.45681	6	7.438e-15
All powers 1	-2483.247	305.1733	6	0

	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
lambda						
enginesize	.2550441	.1304686	1.95	0.051	-.0006697	.5107579
cylinders	-.0025143	.1745643	-0.01	0.989	-.344654	.3396255
horsepower	-.0169707	.1182906	-0.14	0.886	-.2488161	.2148747
highwaympg	-1.375276	.1966211	-6.99	0.000	-1.760646	-.9899057
weight	1.069233	.226236	4.73	0.000	.6258187	1.512647
wheelbase	.0674801	.6685338	0.10	0.920	-1.242822	1.377782

```
. test (enginesize=.25)(cylinders=0)(horsepower=0)(highwaympg=-1)
> (weight=1)(wheelbase=0)
 ( 1)  [lambda]enginesize = .25
 ( 2)  [lambda]cylinders = 0
 ( 3)  [lambda]horsepower = 0
 ( 4)  [lambda]highwaympg = -1
 ( 5)  [lambda]weight = 1
 ( 6)  [lambda]wheelbase = 0
           chi2(  6) =    3.99
         Prob > chi2 =    0.6777
```

Following the advice of Sheather (2009), we round the suggested powers to the closest interpretable fractions. We will use the `mbctrans` command to create the transformed variables so that we can rerun our regression. We demonstrate it here for all cases on `highwaympg`. The relationship it holds with the variable `dealercost` is used as a reference. Recall how the unscaled transformation may switch correlation relationships with other variables, and how the modified scaled transformation maintains these relationships and the scale of the input variable. The unscaled transformed `highwaympg` is referred to as `unscaled_hmpg`. The scaled transformed version of `highwaympg` is

named `scaled_hmpg`. The modified scaled transformed version of `highwaympg` is named `mod_scaled_hmpg`.

```
. summarize highwaympg
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
  highwaympg |        234    29.39744    5.372014          19         66

. correlate dealercost highwaympg
(obs=234)
             | dealer~t highwa~g
-------------+------------------
  dealercost |   1.0000
  highwaympg |  -0.5625   1.0000

. mbctrans highwaympg,power(-1)

. rename t_highwaympg unscaled_hmpg

. summarize unscaled_hmpg
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
unscaled_h~g |        234    .0349275    .0052762   .0151515   .0526316

. correlate dealercost unscaled_hmpg
(obs=234)
             | dealer~t unscal~g
-------------+------------------
  dealercost |   1.0000
unscaled_h~g |   0.6779   1.0000

. mbctrans highwaympg,power(-1) scale

. rename t_highwaympg scaled_hmpg

. summarize scaled_hmpg
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
 scaled_hmpg |        234    .9650725    .0052762   .9473684   .9848485

. correlate dealercost scaled_hmpg
(obs=234)
             | dealer~t scaled~g
-------------+------------------
  dealercost |   1.0000
 scaled_hmpg |  -0.6779   1.0000

. mbctrans highwaympg,power(-1) mscale

. rename t_highwaympg mod_scaled_hmpg

. summarize mod_scaled_hmpg
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
mod_scaled~g |        234    810.9419    4.433584   796.0653   827.5595

. correlate dealercost mod_scaled_hmpg
(obs=234)
             | dealer~t mod_sc~g
-------------+------------------
  dealercost |   1.0000
mod_scaled~g |  -0.6779   1.0000
```

Both the scaled and modified scaled transformation kept the same correlation relationship between `highwaympg` and `dealercost`. The unscaled transformation did not. Additionally, the modified scaled transformation maintained a scale much closer to that of the original than either of the other transformations. Now we will use `mbctrans` on all the variables.

```
. mbctrans enginesize cylinders horsepower highwaympg weight wheelbase,
> power(.25 0 0 -1 1 0) mscale
```

The box plots for the transformed data show a definite improvement in marginal normality.

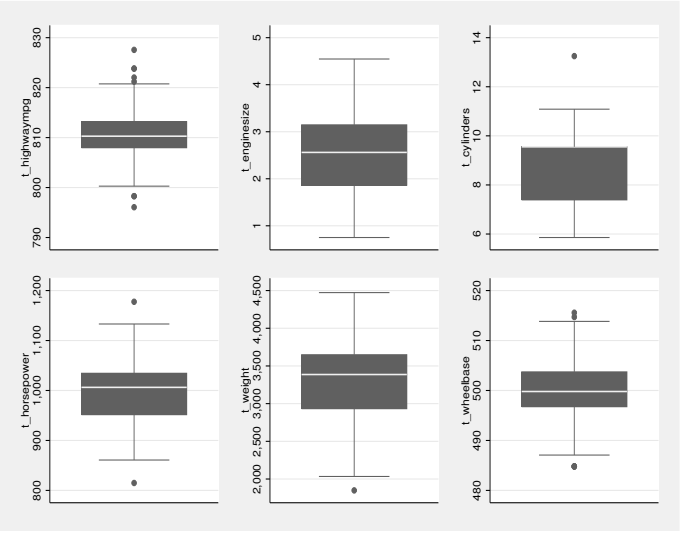

Figure 4. Box plots transformed response and predictors.

A matrix plot of the predictors and response shows greatly improved linearity.

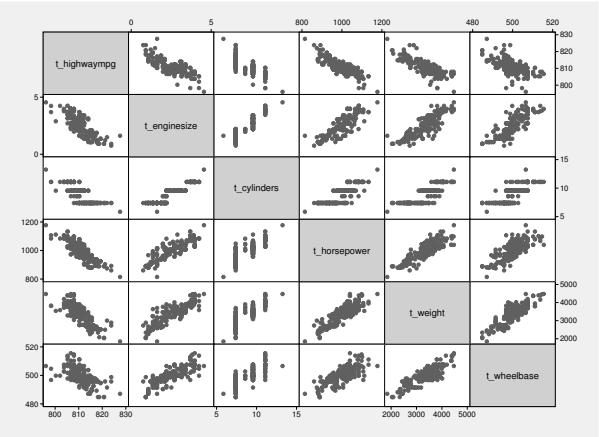

Figure 5. Matrix plot transformed response and predictors.

Now we refit the model with the transformed variables.

```
. regress t_highwaympg t_enginesize t_cylinders t_horsepower t_weight
> t_wheelbase hybrid
```

Source	SS	df	MS		
Model	3581.57374	6	596.928957	Number of obs	= 234
Residual	998.430492	227	4.39837221	F(6, 227)	= 135.72
				Prob > F	= 0.0000
Total	4580.00424	233	19.6566705	R-squared	= 0.7820
				Adj R-squared	= 0.7762
				Root MSE	= 2.0972

t_highwaympg	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
t_enginesize	-.406318	.4557007	-0.89	0.374	-1.304262	.4916264
t_cylinders	-.5353418	.2622172	-2.04	0.042	-1.052033	-.0186507
t_horsepower	-.0280757	.0051522	-5.45	0.000	-.038228	-.0179234
t_weight	-.0042486	.0006911	-6.15	0.000	-.0056103	-.0028868
t_wheelbase	.2456528	.0490344	5.01	0.000	.1490321	.3422736
hybrid	6.552501	1.276605	5.13	0.000	4.03699	9.068012
_cons	735.9331	23.74779	30.99	0.000	689.1388	782.7274

```
. predict trstd, rstandard
. predict tfit, xb
. generate tnsrstd = sqrt(abs(trstd))
. local i = 1
. foreach var of varlist tfit t_enginesize t_cylinders t_horsepower t_weight
> t_wheelbase {
  2. twoway scatter tnsrstd `var´ || lfit tnsrstd `var´,
> ytitle("|Std. Residuals|^.5") legend(off) ysize(5) xsize(5) name(gg`i´)
> nodraw
  3. local i = `i´ + 1
  4. }
. graph combine gg1 gg2 gg3 gg4 gg5 gg6, rows(2) ysize(10) xsize(15)
```

The nonconstant variance has been drastically improved. The use of `mboxcox` helped improve the fit of the model.

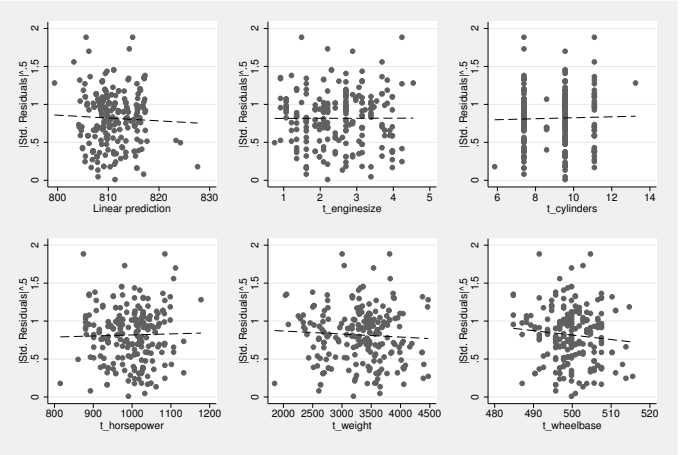

Figure 6. $\sqrt{|\,\text{Standard residuals}\,|}$ versus transformed predictors and fitted values.

4 Conclusion

We explored both the theory and practice of the multivariate Box–Cox transformation. Using both generated and real datasets, we have demonstrated the use of the multivariate Box–Cox transformation in achieving multivariate normality and creating linear relationships among variables.

We fully defined the `mboxcox` command as a method for performing the multivariate Box–Cox transformation in Stata. We also introduced the `mbctrans` command and defined it as a method for performing the power transformations suggested by `mboxcox`. Finally, we also demonstrated the process of obtaining transformation power parameter estimates from `mboxcox` and rounding them to theoretically appropriate values.

5 References

Box, G. E. P., and D. R. Cox. 1964. An analysis of transformations. *Journal of the Royal Statistical Society, Series B* 26: 211–252.

Riani, M., and A. C. Atkinson. 2000. Robust diagnostic data analysis: Transformations in regression. *Technometrics* 42: 384–394.

Sheather, S. J. 2009. *A Modern Approach to Regression with R*. New York: Springer.

Velilla, S. 1993. A note on the multivariate Box–Cox transformation to normality. *Statistics and Probability Letters* 17: 259–263.

———. 1995. Diagnostics and robust estimation in multivariate data transformations. *Journal of the American Statistical Association* 90: 945–951.

Weisberg, S. 2005. *Applied Linear Regression*. 3rd ed. New York: Wiley.

About the authors

Charles Lindsey is a PhD candidate in statistics at Texas A & M University. His research is currently focused on nonparametric methods for regression and classification. He currently works as a graduate research assistant for the Institute of Science Technology and Public Policy within the Bush School of Government and Public Service. He is also an instructor of a course on sample survey techniques in Texas A & M University's Statistics Department. In the summer of 2007, he worked as an intern at StataCorp. Much of the groundwork for this article was formulated there.

Simon Sheather is professor and head of the Department of Statistics at Texas A & M University. Simon's research interests are in the fields of flexible regression methods, and nonparametric and robust statistics. In 2001, Simon was named an honorary fellow of the American Statistical Association. Simon is currently listed on http://www.ISIHighlyCited.com among the top one-half of one percent of all mathematical scientists, in terms of citations of his published work.