Transforming to Linearity

James H. Steiger

Department of Psychology and Human Development
Vanderbilt University

Transforming to Linearity

Introduction

- You've gathered your data on X and Y, plotted them, and although the points seem to follow a functional rule, it is questionable whether the functional rule is linear.
- Suppose for now that the relationship appears to be monotonic but nonlinear.
- Some advantages would accrue if we could monotonically transform X (and possibly Y) so that the graph has been "straightened out" to be linear.
  1. Ordinary linear regression can be used to derive an equation representing the relationship between X and Y.
  2. Departures from linearity are easy to spot.
  3. The transformation to linearity may also move the data in the direction of constant variances around the regression line.

Introduction

- We know that, if the data are nonlinear, we will need to apply a nonlinear transformation to X and/or Y in order to straighten out the plot.
- But how should we proceed?
- Some of the most famous statisticians of the 20th century thought a lot about this problem, and we'll review some of their key findings in the next sections.

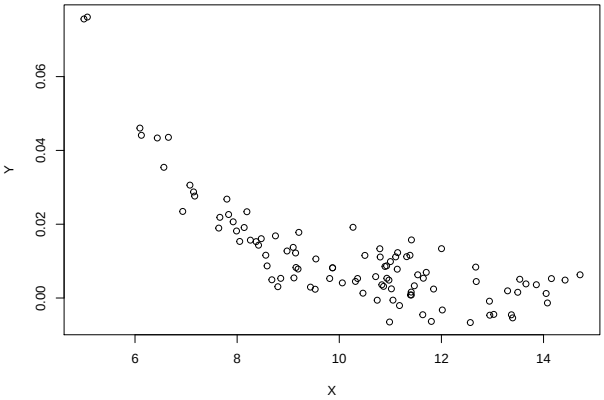
Power Transforms

- Power transforms can be very effective when the relationship between X and Y is "simple monotone," that is, either strictly increasing or strictly decreasing with no inflection point.
- If there is an "inflection point" at which the second derivative (direction of change of slope) changes sign, then a power transform alone will not achieve linearity.
- In that case, we may need to switch to a more complex model, as discussed in other lecture modules.

Power Transforms

Simple Monotonic

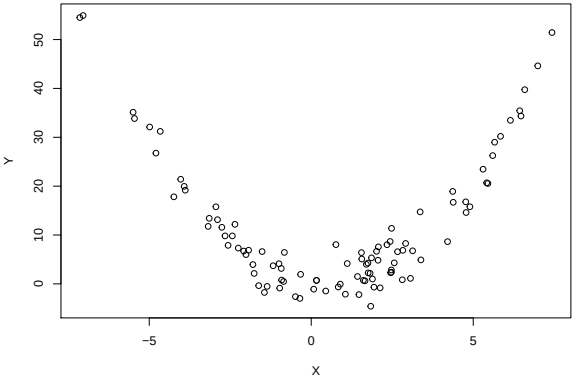
A power transformation will achieve linearity for these data.



Power Transforms

Non-Monotonic

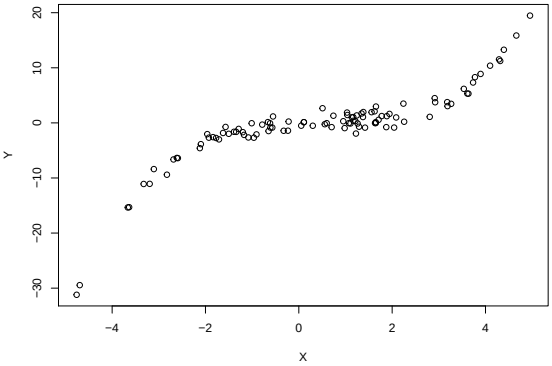
The relationship is not monotonic. A power transformation will be unsuccessful transforming these data to linearity.



Power Transforms

Monotonic with an Inflection Point

Here the relationship is monotonic, but has an inflection point, and a power transform will not achieve linearity.



Power Transforms

- We seek a transformation so if X is the transformed predictor and Y is the transformed response, then the mean function in the transformed scale is

$$E(Y|X = x) \approx \beta_0 + \beta_1 x \tag{1}$$

- We use "$\approx$" rather than "$=$" to recognize that whatever model we end up employing is almost certainly not literally true, but is only an approximation.

Power Transforms

- A transformation family is a collection of transformations that are indexed by one or a few parameters that the analyst can select conveniently.
- The family that is used most often is called the *power family*.
- For a strictly positive variable U the power family is of the form

$$\psi(U,\lambda) = \begin{cases} \log(U) & \text{for } \lambda = 0 \\ U^{\lambda} & \text{otherwise} \end{cases} \tag{2}$$

- If U is not strictly positive, then the transformation can be applied to $U + c$, where c is a constant so that the minimum value of $U + c$ is a positive number, perhaps 1.
- Some statisticians refer to $\log(U + c)$ as "started logs," and $(U + c)^{\lambda}$ as "started powers."

The Box-Cox Transform

- In applying the power simple power transform, there is a technical problem: there is a discontinuity at $\lambda = 0$.
- The scaled power transformation family, discussed by Box and Cox (1964), and often referred to as the *Box-Cox transformation*, is defined as

$$\psi_S(X, \lambda) = \begin{cases} (X^\lambda - 1)/\lambda & \text{if } \lambda \neq 0 \\ \log(X) & \text{if } \lambda = 0 \end{cases} \tag{3}$$

- This function family is actually continuous in λ, since $\lim_{\lambda \to 0} \psi_S(X, \lambda) = \log_e(X)$.
- Also, scaled power transformations preserve the direction of association, in the sense that if X and Y are positively related, then $\psi_S(X, \lambda)$ and Y are positively associated for all values of λ. (With basic power transformations, the direction of association changes when $\lambda < 0$.)
- When transforming X, we shall use the Box-Cox transformation.

The Box-Cox Transform

- When transforming Y and trying to find the "best" value of λ in the Box-Cox transformation, there is an additional technical problem.
- After transforming the Y variable, the scores are no longer in their original metric.
- Consequently the residual sum of squares no longer has the same statistical meaning as it did prior to transformation.
- As a result, one cannot find the best λ by comparing the residual variance (or residual sum of squares) for several competing values of λ.
- A solution to this problem is straightforward: Use the normalized Box-Cox transformation discussed on the next slide when transforming the dependent variable Y.

The Normalized Box-Cox Transform

- The normalized Box-Cox family $\psi_M(Y, \lambda_y)$ for a strictly positive variable Y is a simple modification of the Box-Cox power transformation family, in which an additional multiplier based on the geometric mean is employed to simplify the derivation of a maximum-likelihood method:

$$\psi_M(Y, \lambda_y) = \begin{cases} \text{gm}(Y)^{1-\lambda_y} \times (Y^{\lambda_y} - 1)/\lambda & \text{for } \lambda_y \neq 0 \\ \text{gm}(Y) \times \log y & \text{for } \lambda_y = 0 \end{cases}$$

where the *geometric mean* gm is defined as

$$\text{gm}(Y) = \exp((\sum_i \log y_i)/N)$$

.

Choosing a Transform

The Mosteller-Tukey Bulging Rule

- The power transformation bends the scatter plot in a predictable way.
- Mosteller and Tukey(1977), in their classic text *Data Analysis and Regression*, spoke of going up or down a "ladder of re-expression" as λ is increased or decreased.
- Mosteller and Tukey quickly add a serious caution: When a nonlinear transformation is performed strictly on an empirical basis, extrapolation beyond the range of the data is extremely dangerous. On the other hand, if guided by strong theory, such extrapolation might be reasonable.
- To keep computational effort reasonable, Mosteller and Tukey suggested powers λ on the ladder of values $-3, -2, -1, -1/2, \#, 1/2, 1, 2, 3$, with $\#$ referring to a log transformation. On page 84, they present a graphical representation of their "bulging rule."

Choosing a Transform

The Mosteller-Tukey Bulging Rule

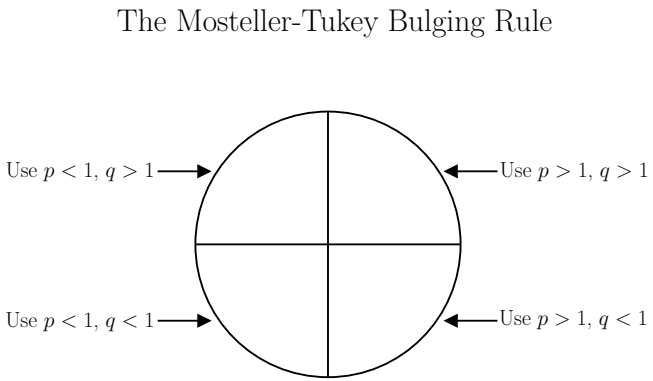
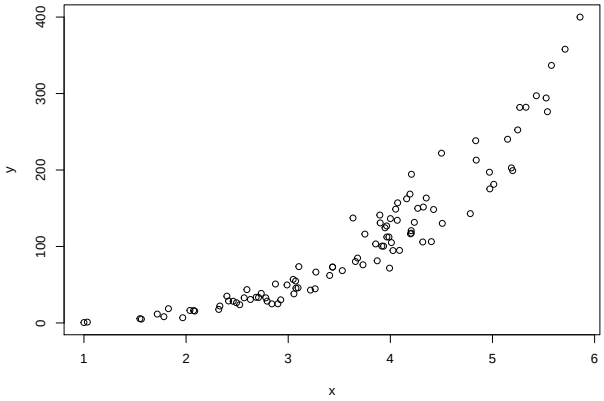


Figure 1: *The Mosteller-Tukey Bulging Rule.* After adding a constant to X and Y if necessary so that both variables are positive, apply a power transformation X^p and/or Y^q. Choose p and q according to which quadrant of the above diagram looks most like the original plot of X and Y.

Choosing a Transform

The Mosteller-Tukey Bulging Rule

Here is an artifical data set that demonstrates substantial nonlinearity.



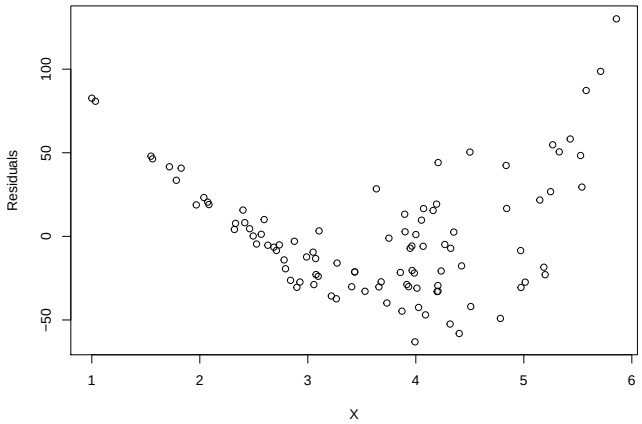
Choosing a Transform

The Mosteller-Tukey Bulging Rule

- Which transformations should we consider?
- The plot looks very much like the lower right quadrant of the Bulging Rule diagram, which suggests that X should be raised to a power greater than 1, and/or Y to a power less than 1.
- But which variables whould we transform?
- Looking at the residual plot may help.

Choosing a Transform

The Mosteller-Tukey Bulging Rule



Choosing a Transform

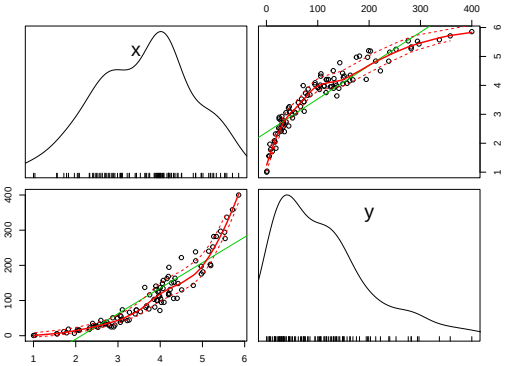
The Mosteller-Tukey Bulging Rule

- Notice that the residuals grow larger from left to right in the plot.
- Will transforming X alone help this? No, it will not.
- We will either have to transform Y or try some other method to deal with a nonconstant residual variance that violates the assumption of the simple linear regression model.
- While we're at it, let's use the `scatterplotMatrix` function to look at the individual distributions of X and Y along with their scatterplot.

Choosing a Transform

The Mosteller-Tukey Bulging Rule

```
> scatterplotMatrix(cbind(x,y))
```



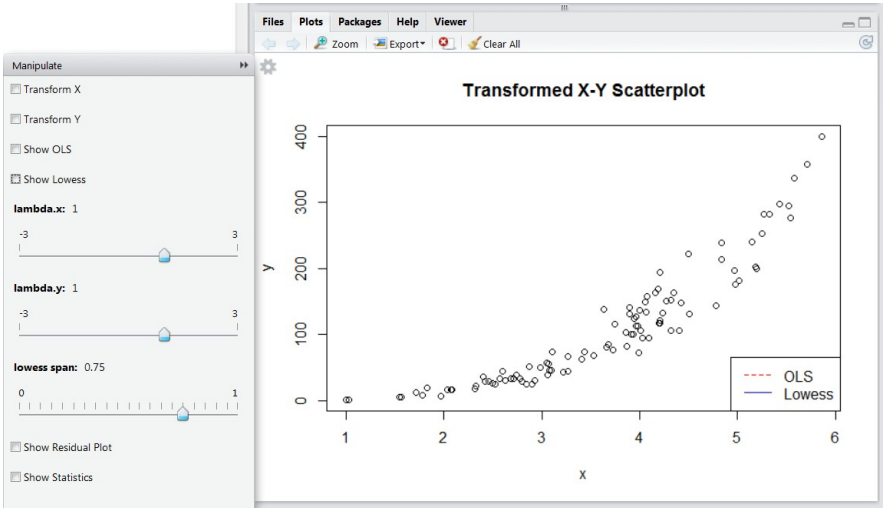
Manually Transforming Y to Linearity

- X is already close to normal in shape.
- A nonlinear power transform will create non-normality.
- Let's try transforming Y, using a visual transforming application I wrote in R.
- You'll need to run this in RStudio with the `manipulate` library loaded.
- Start up RStudio, go to the R console, and type

```
> source("http://www.statpower.net/R2101/TransformDemo.R")
```

- You should see the data open in a manipulate window with sliders, as shown on the next slide.
- Note: If the *Manipulate* window with its controls is not open, click on the cog icon in the upper left of the plot.

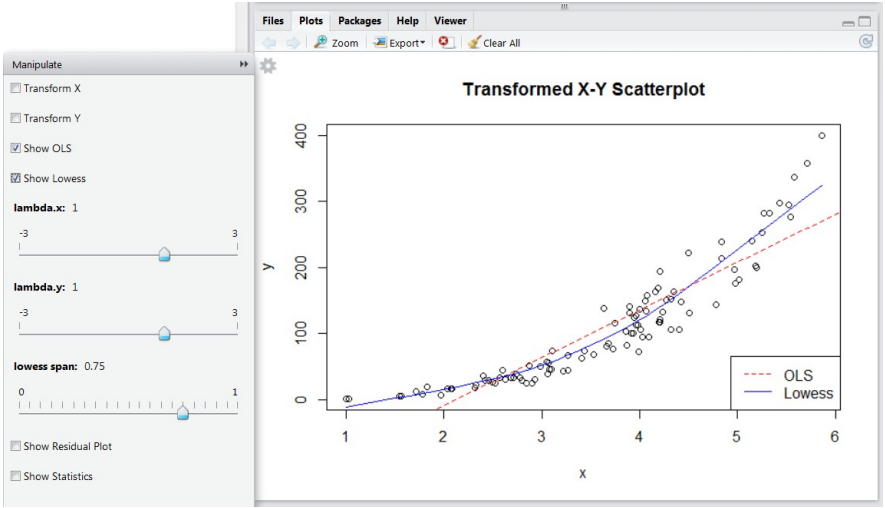
Manually Transforming Y to Linearity



Manually Transforming Y to Linearity

- Notice that λ_x and λ_y are both initialized at 1, so you begin by viewing the X and Y variables in their original, untransformed form.
- There are two checkboxes at the top-left of the plot for display the OLS line of best linear fit and the Lowess smoothed line, which of course will be curvilinear.
- Click on both checkboxes to display the fit lines.
- The disparity between the linear fit line and the Lowess line is one way of demonstrating the essential nonlinearity of the relationship.

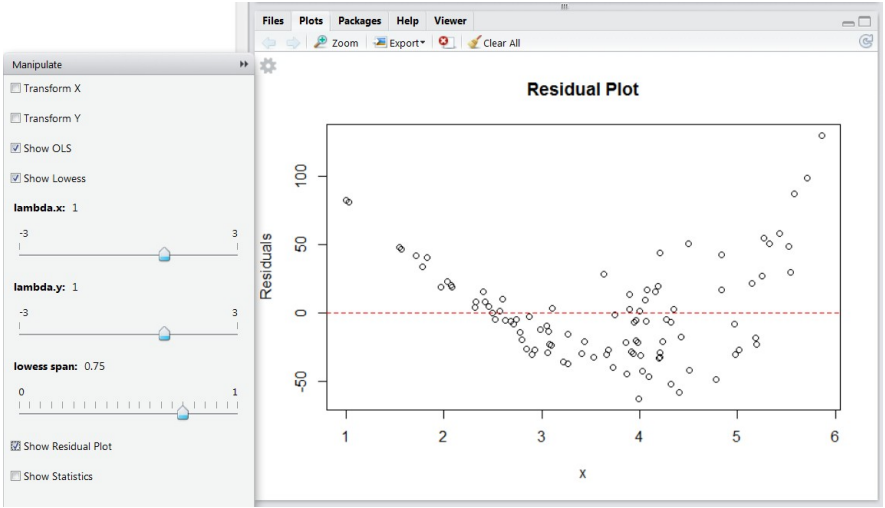
Manually Transforming Y to Linearity



Manually Transforming Y to Linearity

- Another way to examine nonlinearity visually is to display a plot of residuals versus X.
- Click on the *Show Residual Plot* checkbox to display a plot of the residuals.
- You can clearly see from the curved shape of the plot that linearity is not satisfied.

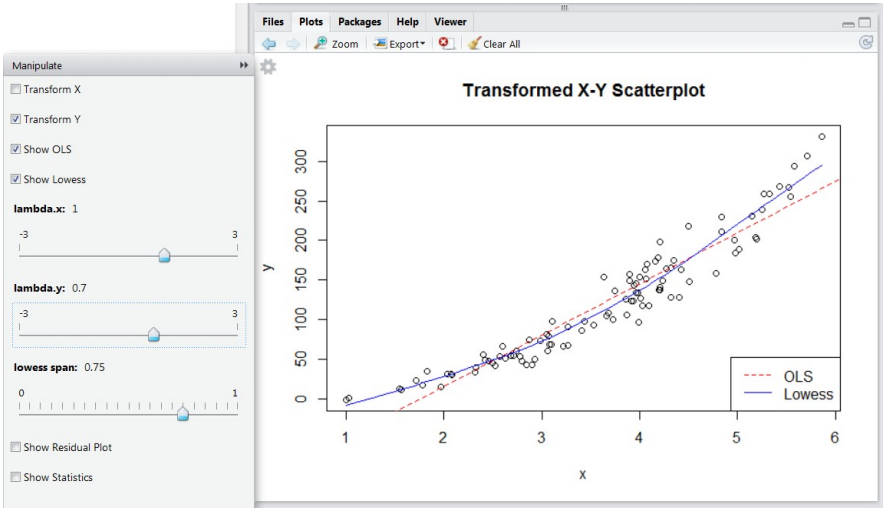
Manually Transforming Y to Linearity



Manually Transforming Y to Linearity

- Uncheck the the *Show Residual Plot* checkbox to show the X-Y scatterplot again.
- Now we are going to transform Y.
- Using the mouse, drag the *lambda-y* slider to the left until the value changes from 1.0 to about 0.7.
- Release the slider, and you will note that the relationship between X and the transformed Y has become substantially more linear.

Manually Transforming Y to Linearity



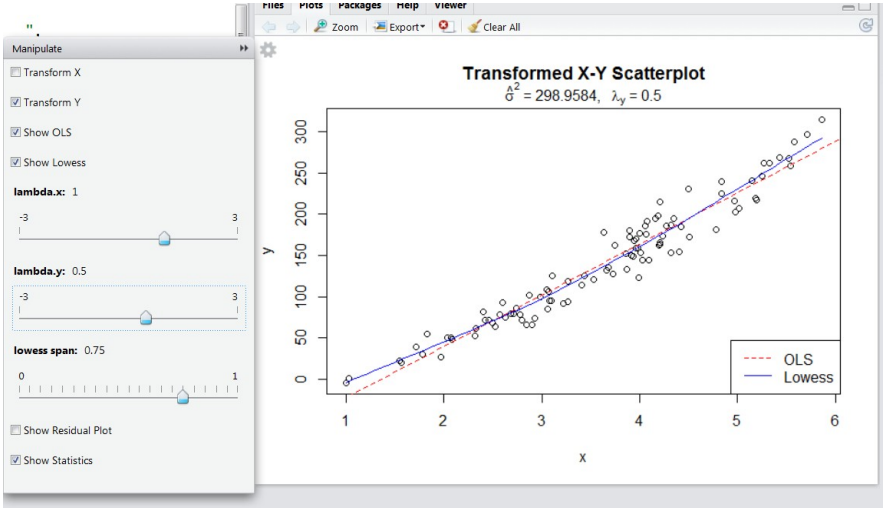
Manually Transforming Y to Linearity

- Our transformation appears to be moving in the correct direction.
- To get finer control of the transformation process, use the left and right arrow keys on your keyboard.
- Tap the left arrow key several times to reduce λ_y to 0.50. Note how the plot changes and becomes increasingly linear.
- We are getting close to an optimal Box-Cox transform, as evidenced by the fact that the Lowess and linear lines are now very close together.

Manually Transforming Y to Linearity

- In the final stages, we'll use the estimated residual variance to decide when we have optimized our transformation.
- Click on the *Show Statistics* checkbox in order to display current values of $\hat{\sigma}^2$, λ_x, and λ_y in the plot subtitle.

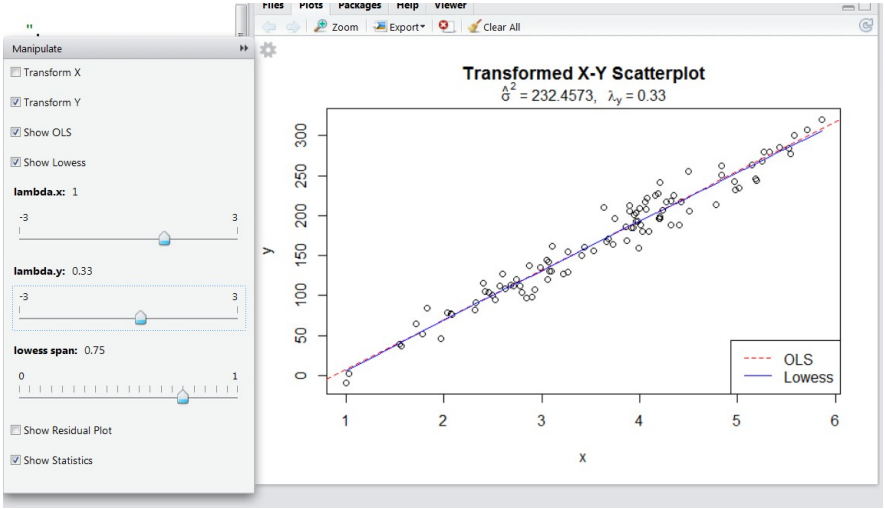
Manually Transforming Y to Linearity



Manually Transforming Y to Linearity

- The current residual variance is around 307.
- Start tapping on the left arrow key again, watching the value of $\hat{\sigma}^2$ carefully.
- You will see the plot become increasingly linear, to the point where the Lowess and OLS lines are virtually identical.
- You should also see the value of $\hat{\sigma}^2$ get increasingly smaller until it bottoms out at $\lambda_y = 0.32$.
- Changing λ_y to be less than 0.32 causes the value of $\hat{\sigma}^2$ to start increasing, and the plot to become less linear.
- Evidently, $\lambda_y = 0.32$ gives the optimal Box-Cox transformation.

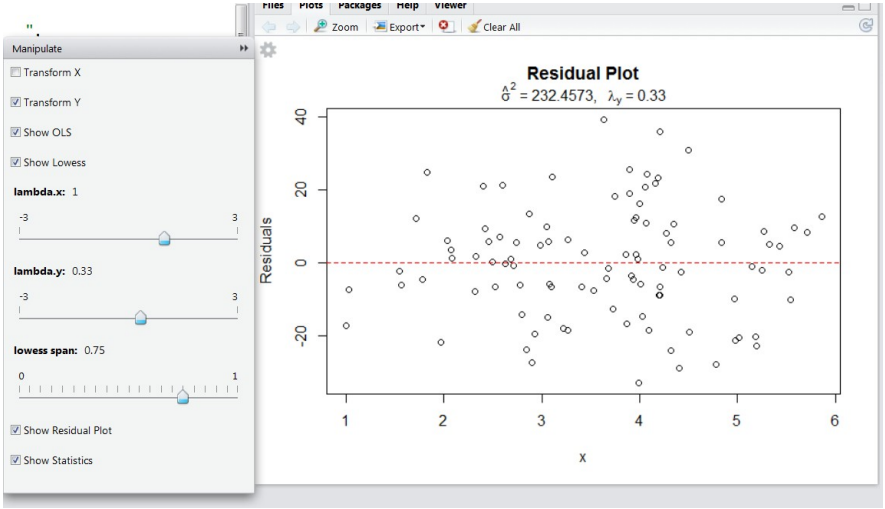
Manually Transforming Y to Linearity



Manually Transforming Y to Linearity

- Click on the *Show Residual Plot* checkbox again.
- Note how the residuals are now essentially randomly distributed around 0.

Manually Transforming Y to Linearity

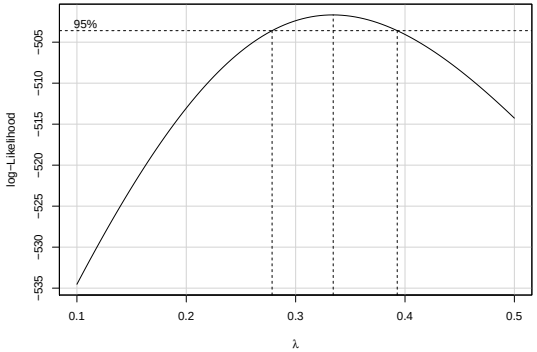


Manually Transforming Y to Linearity

- Our estimate of the best transformation for Y is $\lambda_y = 0.33$, essentially the cube root of Y, which suggests that the original Y was a cubic function of X.
- An approximate likelihood-based confidence interval can be established for this estimate, using the `boxCox` function in the verb—car— library.
- After approximating the value of λ, one zooms in on the graph to establish the confidence limits, as shown on the next slide.

Manually Transforming Y to Linearity

```
> boxCox(y~x, lambda=seq(.1,.5,.01))
```



Manually Transforming Y to Linearity

- The confidence limits appear to run from .28 to .39.
- By setting `plotit=FALSE`, one may obtain a table of values for the log likelihood and establish the confidence interval to a higher level of precision.
- However, two decimal places are usually sufficient.

The Log Rule and the Range Rule

Weisberg cites two rules that would predict the usefulness of the log transformation:

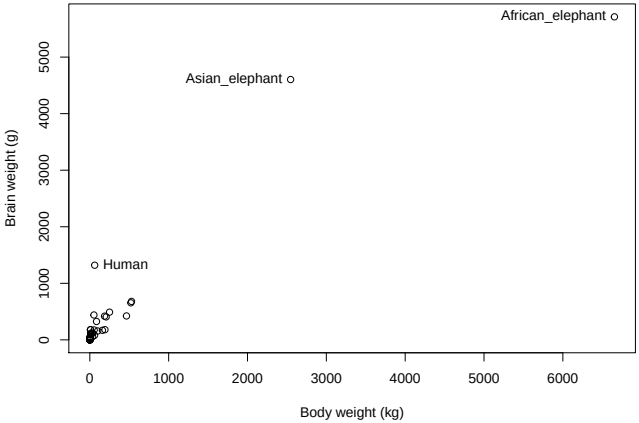
- *The Log Rule.* If the values of a variable range over more than one order of magnitude and the variable is strictly positive, then replacing the variable by its logarithm is likely to be helpful.
- *The Range Rule.* If the range of a variable is considerably less than one order of magnitude, then any transformation of that variable is unlikely to be helpful.

The Log Rule and the Range Rule

- The next slide contains a plot of body weight (*BodyWt*) in kilograms and brain weight (*BrainWt*) in grams for 62 mammalian species in the file `brains.txt`.
- Apart from the three separated points for two species of elephants and for humans, the uneven distribution of points hides any useful visual information about the mean of *BrainWt*, given *BodyWt*.
- It seems, in any case, that the data are decidedly nonlinear.
- Both *BodyWt* and *BrainWt* easily satisfy the Log Rule.

Transforming X and Y

Brainweight vs. Bodyweight



Transforming X and Y

Brainweight vs. Bodyweight

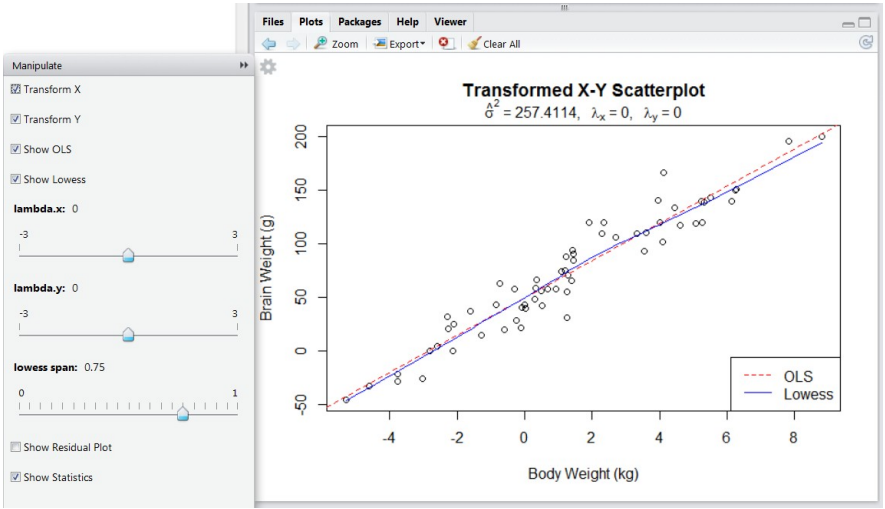
- Both variables seem to satisfy the Log Rule, so let's open our transformation utility in RStudio and examine the data.
- Enter the following code in the R console window of RStudio.

```
> library(alr4)
> attach(brains)
> TransformSetup(BodyWt,BrainWt,xlab="Body Weight (kg)"
+                ,ylab="Brain Weight (g)")
```

- Next, transform both variables with $\lambda_x = \lambda_y = 0$.
- Turn on the OLS and Lowess fit lines, and display statistics.
- Your figure should essentially match the next slide.

Transforming X and Y

Brainweight vs. Bodyweight



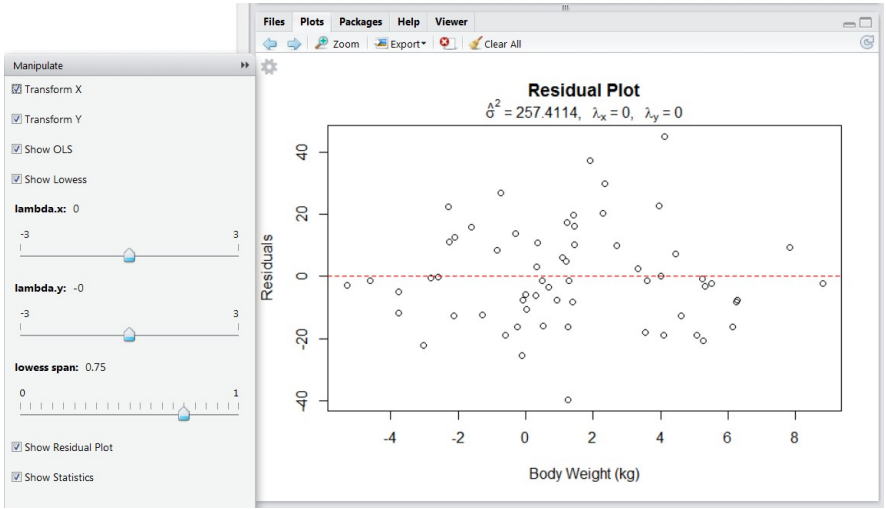
Transforming X and Y

Brainweight vs. Bodyweight

- If you experiment with values of λ_y and/or λ_x from here, you'll find only very small improvements om residual variance can be achieved by moving slightly away from the log transforms.
- Examining the residuals reveals that they show no visually obvious non-random trend.

Transforming X and Y

Brainweight vs. Bodyweight



Interpretation of Log-Transformed Regressions

- When we deal with the simple regression $\hat{Y} = \beta_1 X + \beta_0$, interpretation of β_1 is easy: increase of 1 unit in X is associated with an increase of β_1 units in the expected value of Y.
- Once variables are log-transformed, things become more complicated.

Interpretation of Log-Transformed Regressions

Regression of $\log(Y)$ on X

- For a model $\log(Y) = \beta_0 + \beta_1 X + e$, a 1 unit increase in X results in a β_1 unit increase in log(Y), which is equivalent to multiplying Y by $\exp(\beta_1)$.
- Specifically, if $\log(f(X)) = \beta_0 + \beta_1 X$, then $f(X) = \exp(\beta_0 + \beta_1 X) = \exp(\beta_0)\exp(\beta_1 X)$. Suppose we increase X by 1 unit. Then $f(X+1) = \exp(\beta_0)\exp(\beta_1(X+1)) = \exp(\beta_0)\exp(\beta_1 X)\exp(\beta_1) = f(X)\exp(\beta_1)$.
- For small values of β_1, $exp(\beta_1) \approx 1 + \beta_1$, so an increase of 1 unit in X would result in a percentage change in Y of $(100 \times \beta_1)\%$.
- For example, if $\beta_1 = 0.02$, a 1 unit increase in X would result in approximately a 2% increase in Y.

Interpretation of Log-Transformed Regressions

Regression of Y on $\log(X)$

- Suppose $f(X) = \beta_0 + \beta_1 \log(X)$.
- Suppose we increase X by 1%. What will happen to $f(X)$?
- Consider the change in $f(X)$. $f(X_2) - f(X_1) = \beta_1 \log(X_2) - \beta_1 \log(X_1) = \beta_1 \log(X_1/X_1)$.
- For small c, $\log(1 + c) \approx c$, and so a 1% change in X corresponds to a change of slightly less than $\beta_1/100$ in Y.
- For example, if $X_1/X_1 = 1.01$, $f(X)$ is multiplied by $\beta_1 \log(1.01) = 0.00995\beta_1$, slightly less than $\beta_1/100$.

Interpretation of Log-Transformed Regressions

Regression of $\log(Y)$ on $\log(X)$

- Suppose we increase X by exactly 1%. What will be the proportional change in Y? $\log(Y) = \beta_1 \log(X) + \beta_0$. Now, suppose we exponentiate both sides. We get $\exp(\log(Y)) = \exp(\beta_1 \log(X) + \beta_0)$, or, using the laws of exponents,

$$Y = X^{\beta_1} \exp(\beta_0)$$

.

Interpretation of Log-Transformed Regressions

Regression of $\log(Y)$ on $\log(X)$

- With a little algebra, we arrive at

$$\begin{aligned} X_2 &= 1.01X_1 \\ Y_2/Y_1 &= \frac{(1.01X_1)^{\beta_1}\exp(\beta_0)}{X_1^{\beta_1}\exp(\beta_0)} \\ &= 1.01^{\beta_1} \end{aligned} \tag{4}$$

The first few terms of the Taylor Series approximation of 1.01^{β_1} are $1 + 0.00995033\beta_1 + 0.0000495045\beta_1^2$. This is very close to $1 + .01\beta_1$. In other words, a proportional change of 1% in X will result in a multiple of $1 + .01\beta_1$, which approximately a *proportional* change of β_1% in Y.

Variance Stabilizing Transformations

Transforming Proportions

- In many cases the variance of the dependent variable will be a predictable function of its expected value.
- One well-known example is a sample proportion that is an estimate of a probability.
- Since probabilities are bounded by 0 and 1, simple linear regression cannot directly model probabilities as a function of X.
- Moreover, many quantities we encounter in practice are "disguised proportions,", i.e., proportions that are multiplied (or divided) by some constant.
- Examples include number correct on an exam and infant mortality rates per 1000 births.
- Proportion data represents a special challenge that is usually dealt with by a special kind of regression called logistic regression, which we'll deal with in a separate module.

Variance Stabilizing Transformations

Transforming Proportions

- ALR4 also discusses some transforms that have been used to eliminate certain kinds of dependencies between means and variances.
- The problem is, these transforms may introduce nonlinearity.
- Consequently, with count data (where means and variances are related) and proportion data, Poisson regression and logistic regression have supplanted the use of variance-stabilizing transforms

Variance Stabilizing Transformations

Transforming Proportions

Table 7.5 Common Variance Stabilizing Transformations

Y_T	Comments
$\sqrt{Y}$	Used when $\text{Var}(Y\|X) \propto \text{E}(Y\|X)$, as for Poisson distributed data. $Y_T = \sqrt{Y} + \sqrt{Y+1}$ can be used if many of the counts are small (Freeman and Tukey, 1950).
$\log(Y)$	Use if $\text{Var}(Y\|X) \propto [\text{E}(Y\|X)]^2$. In this case, the errors behave like a percentage of the response, ±10%, rather than an absolute deviation, ±10 units.
$1/Y$	The inverse transformation stabilizes variance when $\text{Var}(Y\|X) \propto [\text{E}(Y\|X)]^4$. It can be appropriate when responses are mostly close to 0, but occasional large values occur.
$\sin^{-1}(\sqrt{Y})$	The *arcsine square-root* transformation is used if Y is a proportion between 0 and 1, but it can be used more generally if y has a limited range by first transforming Y to the range $(0, 1)$, and then applying the transformation.