Box-Cox Transformation

To illustrate the use of the Box-Cox power transformation, let's consider an artificial data set generated from the following model:

$$\log(y) = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 \cdot x_3 + \varepsilon \, .$$

```
n <- 100
x1 <- rnorm(n)
x2 <- runif(n)
x3 <- sample(c(0,1), n, replace=T)
y <- exp(1 + 2*x1 + 1.5*x2*x3 + rnorm(n, 0, .5))
dat <- data.frame(y=y, x1=x1, x2=x2, x3=x3)
dat

                y          x1          x2 x3
1   235.78240903  1.38795309 0.773861234  1
2     0.01889094 -2.71191803 0.975024951  0
3     0.07631980 -1.63615631 0.001274388  1
4    10.63959125  0.07934400 0.370508643  1
              .........
98    3.39144289  0.28762264 0.621960530  0
99    0.68726918 -1.12069670 0.295496882  0
100 108.38809753  1.83044089 0.189433787  0
```

To apply the Box-Cox transformation, click on the "**packages**" in the main menu. Then select the option "**Load Package**". A window containing all the available packages will prompt up. Choose the **MASS** package and press the **OK** button. Now all the build-in functions in the MASS library are available to use.

You may type `?boxcox` to get the help information on the function `boxcox`. The argument `plotit = TRUE` plots loglik *vs* λ and indicates a 95% confidence interval about the maximum observed value of λ.

```
> inter <- x2*x3
> boxcox(y ~ x1 + inter, data = dat,  plotit=T)
```

The above command gives a plot as shown in Figure 1. The vertical axis is actually the maximized log-likelihood for each value of λ, which is a function of SSE or R. The best transformation goes to $\lambda = 0$ as expected. Namely, a logarithm transformation is needed.

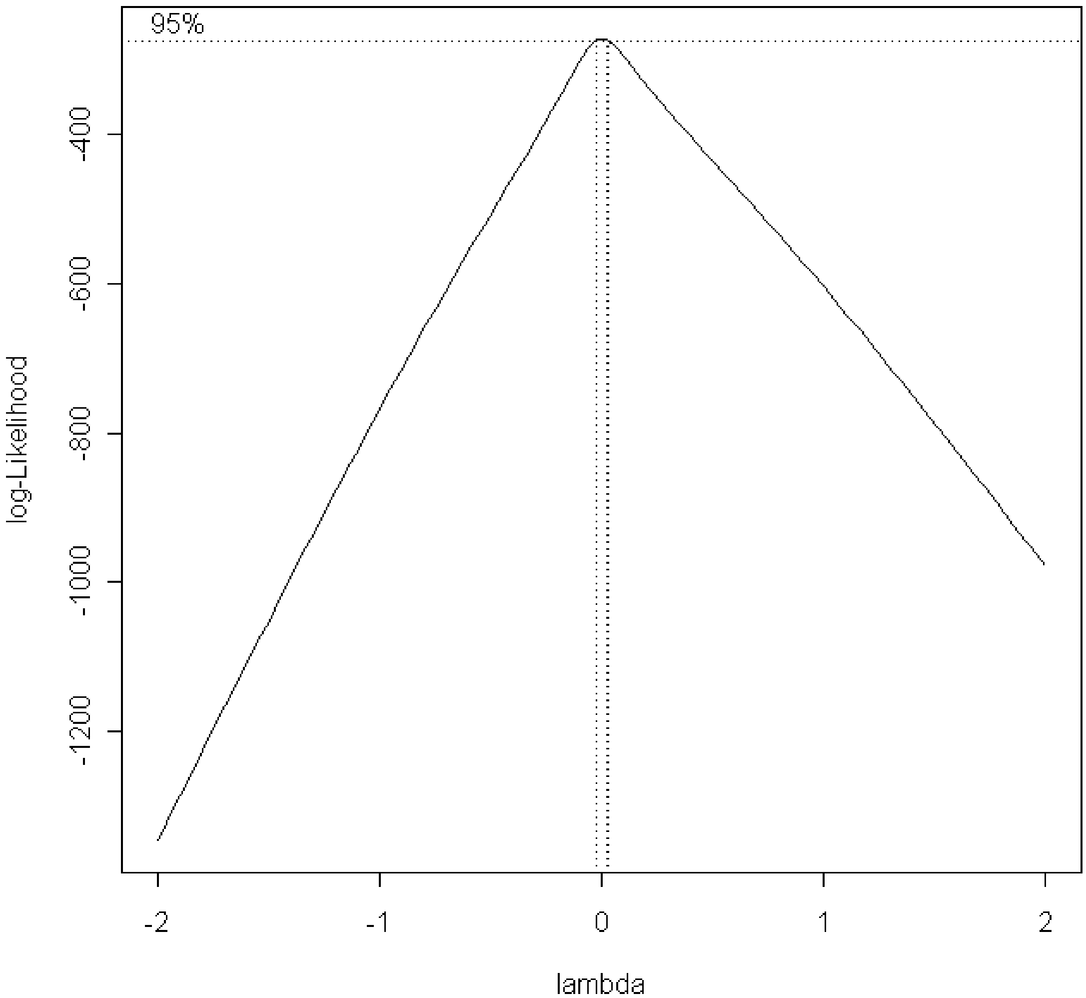

Figure 1: The maximized log-likelihood vs. different values of λ