Box-Tidwell Transformation Realization with R language

Yi He ChengYu Huang JunJie Liu
JiaChen Tu DaiChen Yao

United International College - STAT

December 9, 2018

Overview

Intro

Procedure

Step by Step Procedure

Step Zero

$$y = \beta_0 + \beta_1 w_1 + \cdots + \beta_k w_k + \epsilon$$

$$w_j = f(x) = \begin{cases} x_j^{\alpha_j} \alpha \neq 0 \\ ln(x_j), \alpha = 0 \end{cases}$$

The method accommodates exponents on one or more of the regressor variables. $\alpha_1, \alpha_2, \ldots, \alpha_k$

Step by Step Procedure (Continued)

Step One (Inital Model)

- Do a multiple linear regression:

$$E(y_i) = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_k x_{ki}$$

  We denote the parameter estimates by $b_0, b_1, \ldots, b_k$

- We can get: $z_j = x_j ln(x_j)$ which x_j comes from the original model
- Do a regression of y on $x_1, x_2, \ldots, x_k, z_1, z_2, \ldots, z_k$. Thus estimate $\gamma_1, \gamma_2, \ldots, \gamma_k$, the coefficients of the z's. Denote the estimates by $\hat{\gamma_1}, \hat{\gamma_2}, \ldots, \hat{\gamma_k}$
- Estimate $\alpha_1, \alpha_2, \ldots, \alpha_k$ by:

$$\hat{\alpha}_j = \frac{\hat{\gamma}_j}{b_j} + 1 \quad (j = 1, 2, \ldots, k)$$

Step by Step Procedure (Continued)

Step Two (New Model and the Iteration)
The results of equation 6 may be viewd as an updated estimate of α_j. Often a one step computation is sufficient, in the example below, we do four iterations. Here is the procedure of the iteration model and the how to update the α:

- Use $w_1^* = x_1^{\hat{\alpha}}, w_2^* = x_2^{\hat{\alpha}} \ldots, w_k^* = x_k^{\hat{\alpha}_k}$ to fit the model

  $$E(y_i) = \beta_0 + \beta_1 w_1^* + \cdots + \beta w_k^*$$

  with estimates $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_k$
- Define $z_1^* = w_1^* ln(w_1^*),\ z_2 = w_2^* ln(w_2^*), \ldots, z_k = w_k^* ln(w_k^*)$
- Fit a regression of y on $w_1^*, w_2^*, \ldots, w_k^*, z_1^*, z_2^*, \ldots, z_k^*$ with a new coefficients of $z_1^*, z_2^*, \ldots, z_k^*$ denoted by $\hat{\gamma}_1, \hat{\gamma}_2, \ldots, \hat{\gamma}_k$

Step by Step Procedure (Continued)

Step Two (Update Term)
We compute the update $alp\hat{h}a_j$ with the formula:

$$\hat{\alpha}_j = (\frac{\hat{\gamma}_j}{\hat{\beta}_j} + 1) \times (\text{ Current Value of } \hat{\alpha}_j)$$

We can get the new α for the new iteration, after n times iterations ($|(\alpha_{k-1} - \alpha_{k-2}| \leq \epsilon$,$\epsilon$ is the tolrance value), we can stop the iteration and at that time, the α is what we need: the relative best fitting curve. In multiple regression it can also be a good use of this method, one more variable is to do a set of data loop.

R Code

Code One

```
get.Beta.hat <- function(y,x){
  y <- as.matrix(y)
  x <- cbind(constant = 1, as.matrix(x))
  beta.hat <- solve(t(x)%*%x) %*% t(x)%*%y # (x'x)^-1 * (x'y)

  # Compute standard errors
  s2 <- sum((y - x%*%beta.hat)^2)/(nrow(x) - ncol(x))
  VCV <- s2*solve(t(x)%*%x)
  SE <- sqrt(diag(VCV))

  # Compute t-values
  t <- beta.hat/SE

  # Compute p-values
  p <- 2*pt(abs(t),nrow(x) - ncol(x), lower.tail = FALSE)

  # Compute adjusted R-squared
  y_hat <- x%*%beta.hat
  SSr <- sum((y - y_hat)^2)
  SSt <- sum((y - mean(y))^2)
  R2 <- 1 - (SSr/SSt)
  adj.R2 <- 1 - ((1 - R2)*(nrow(x) - 1))/(nrow(x) - ncol(x[,-1]) - 1)


  Table <- as.data.frame(round(cbind(beta.hat,SE,t,p, R2), digits = 4))
  names(Table)[1:5] <- c("Estimate","Standard Error","t-value","p-value", "R^2")
  paste("adj R square is ", adj.R2)
  return(list("Table" = Table, "coefficients" = beta.hat, "R2" = R2))
}
```

Figure

```
boxtidwell <- function(y,x,tol = .001, max.iter = 25){
  iter <- 1 # the initial iteration
  y <- as.matrix(y)
  x <- as.matrix(x)
  w <-x
  alpha <- rep(1, ncol(x)) # initial alpha

  repeat{
    y <- as.matrix(y)
    x <- as.matrix(x)


    w.log.w <- w*log(w) # alpha
    mod.1 <- get.Beta.hat(y ,  w)
    mod.2 <- get.Beta.hat(y , cbind(w,w.log.w)) # new model
    alpha <- (mod.2$coefficients[-c(1,2)] / mod.1$coefficients[-1] + 1) * alpha
    print(iter)
    print(alpha)
    if(alpha <= tol || iter >= 4){
      break
    }
    iter <- iter + 1
    for (i in 1:ncol(x)){
      for (j in 1:nrow(x)){
        if(alpha[i] != 0){
          w[j] <- x[j]^alpha
        }
        if(alpha[i] == 0){
          w[i] <- log(x)
        }
      }
    }
  }
  return(list("iter" = iter, "alpha" = alpha))
}
```

Conclusion

References

- Raymond H.Myers (2005)
- Professor.John Fox (2010)

Box-Tidwell Transformation Realization with R language

Yi He ChengYu Huang JunJie Liu
JiaChen Tu DaiChen Yao

United International College - STAT

December 9, 2018