A Computer Program for Estimating the Re-transformed Mean in Heteroscedastic Two-part Transformation Models

Xiao-Hua Zhou, Hao Cheng

August 1, 2005

Abstract

Welsh and Zhou (2005) proposed a two-part heteroscedastic model to deal with zero costs and handle re-transformation bias when the transformation on non-zero costs achieves linearity and additivity but not necessarily normality and homoscedasticity. They proposed two estimators of the mean response on the original scale, which have been proven to be consistent and asymptotically normally distributed.

In this paper, we develop a software program to implement the newly proposed two-part heterscedastic transformation model proposed by Welsh and Zhou (2005). Our program computes two estimates for the mean cost, their asymptotical standard deviations with the associated normal-based confidence intervals, and bootstrap based confidence intervals. Our program includes a user-friendly interactive mode and more efficient and flexible batch mode. It is written in free statistical computing language R and can be run on a wide variety of platforms.

1 Introduction

Welsh and Zhou [1] developed a two-part heteroscedastic transformation model to handle zero cost and right-skewed non-zero cost data and a method for correcting for re-transformation bias when the transformation achieves linearity and additivity but not necessarily normality and homoscedasticity. They proposed two estimators of the mean response on the original scale. They have proved that both estimators are consistent and asymptotically normally distributed, and have shown how to construct approximate confidence intervals for the mean response on the original scale.

In this paper, we develop a software implementation of Welsh and Zhou's method. We first briefly describe computational details of Welsh and Zhou's method in Section 2. Then we discuss software implementation and usage in Section 3. Finally, we illustrate the use of our computer program in a real application in Section 4.

2 Computational Method

2.1 A heteroscedastic two-part regression model

Let random variables $Y_1, \ldots,$ and Y_n be independent with a density function,

$$f^*(y_i, \pi_i, \phi_i) = \begin{cases} \pi_i & \text{if } y_i = 0, \\ (1-\pi_i) f(y_i, \phi_i) & \text{if } y_i > 0, \end{cases} \tag{1}$$

where $f(y_i, \phi_i)$ is a proper density function. Clearly, $\pi_i = Pr(Y_i = 0)$, and $f(y_i, \phi_i)$ is the conditional density of Y_i given that $Y_i > 0$.

Here π_i is related to known vectors of covariates x_i through a known link function l so that

$$l(\pi_i) = x_i^T \alpha_0, , \tag{2}$$

where α_0 is a vector of unknown parameters. A common choice of l is the logistic function $l(x) = \log(x/(1-x))$, but other choices are possible.

Since $f(y_i, \phi_i)$ is often asymmetric, we adopt a conditional transformation model to relate Y_i to vectors of covariates x_i, given that $Y_i > 0$. Specifically, given that $Y_i > 0$, we assume that there exists a known monotone transformation h so that

$$h^{-1}(Y_i) = x_i^T \beta_0 + g_i(\beta_0, \theta_0)\epsilon_i, \tag{3}$$

where β_0 and θ_0 are vectors of unknown parameters with dimensions q_1 and q_2, respectively, g_i is a known function allowing scaling and heteroscedasticity on the transformed scale, and $\{\epsilon_i\}$'s are independent and identically distributed random variables with the common density function f_ϵ with mean zero and variance one. In model (3), we only assume that the transformation h^{-1} makes the mean linear, leaving us to model any heteroscedasticity through the g_i and to account for possible non-normality of the $\{\epsilon_i\}$.

2.2 Regression parameter estimation

Let $\xi = (\beta^T, \theta^T)^T$ and define

$$e_i(\xi) = \frac{h^{-1}(Y_i) - x_i^T \beta}{g_i(\xi)}. \tag{4}$$

Here the dimension of ξ is $q = q_1 + q_2$. Then the log-likelihood for the model (1) is

$$\begin{aligned} \ell(\alpha, \xi) &= \sum_{i=1}^{n} \left\{ I\left(y_i = 0\right) \log\left(\frac{\pi_i}{1-\pi_i}\right) + \log(1-\pi_i) \right\} + \sum_{i=1}^{n} I(y_i > 0) \log f(y_i, \phi_i) \\ &= \ell_1(\alpha) + \ell_2(\xi). \end{aligned} \tag{5}$$

This factorization shows that the parameters α_0 and ξ_0 are orthogonal so, without any loss of efficiency, can be estimated separately.

Estimation of α_0 by maximizing $\ell_1(\alpha)$ is a standard binary regression problem. Similarly, we can maximize $\ell_2(\xi)$ to obtain an estimate of ξ_0. An alternative approach to estimating ξ_0 is to consider a wider class of estimators which satisfy estimating equations of the form,

$$\sum_{i=1}^{n} \Psi_i(Y_i, \xi) = 0. \tag{6}$$

Writing the derivatives of g as

$$g_i^{(1)}(\xi) = \frac{\partial g_i(\xi)}{\partial \beta} \text{ and } g_i^{(2)}(\xi) = \frac{\partial g_i(\xi)}{\partial \theta},$$

we can show that the maximum likelihood estimator satisfies (6) with $\Psi_i(Y_i, \xi) = I(Y_i > 0)(\psi(e_i(\xi))x_i^T/g_i(\xi)$ $+\chi(e_i(\xi))g_i^{(1)}(\xi)^T/g_i(\xi),\ \chi(e_i(\xi))g_i^{(2)}(\xi)^T/g_i(\xi))^T$, where $\psi(x) = -f_\epsilon'(x)/f_\epsilon(x)$ and $\chi(x) = x\psi(x) - 1$, derived under the assumed parametric distribution f_ϵ of ϵ_i.

When $\{\epsilon_i\}$'s are normally distributed, $\psi(x) = x$ and $\chi(x) = x^2 - 1$. The pseudo likelihood estimator satisfies (6) with $\Psi_i(Y_i, \xi) = I(Y_i > 0)(\psi(e_i(\xi))x_i^T,\ \chi(e_i(\xi))g_i^{(2)}(\xi)^T)^T$;

2.3 Estimating mean

We are interested in estimating the mean of the response on the original-scale using the estimated regression coefficients on the transformed scale. That is, given the covariate x, we want to estimate $u = E(Y \mid x)$, where Y is the response of the outcome on the patient with the covariate x. Define

$$\eta_i(\xi) = x^T\beta + g(\xi)e_i(\xi),$$

where $e_i(\xi)$ is defined by (4), and $g(\xi)$ is the value of $g_i(\xi)$ when $x_i = x$. For simplicity, we write $\eta_i(\xi_0) = \eta_i$. Since $e_i(\xi_0) = \epsilon_i$ and the $\{\epsilon_i\}$ are assumed to be independent and identically distributed, for fixed covariate x, the random variables $\{\eta_i\}$'s are independent and identically distributed. In this notation, we have $Eh(\eta_i) = Eh(\eta_1)$ and that

$$u = (1 - \pi)Eh(\eta_1),$$

where $\pi = l^{-1}(x^T\alpha_0)$.

We consider two different estimators of u. Both estimators are generalizations of the smearing estimator of Duan [2]. Put $\hat{\xi} = (\hat{\beta}^T, \hat{\theta}^T)^T$ and $\hat{\pi}_i = l^{-1}(z_i^T \hat{\alpha})$. Then we have the "externally" weighted estimator

$$\hat{u}^* = \frac{1 - \hat{\pi}}{1 - \bar{\hat{\pi}}} \hat{m}^*, \tag{7}$$

where $\bar{\hat{\pi}} = \frac{1}{n} \sum_{i=1}^n \hat{\pi}_i$ and $\hat{m}^* = \frac{1}{n} \sum_{i=1}^n I(y_i > 0) h(\eta_i(\hat{\xi}))$, and the "internally" weighted estimator

$$\hat{u} = (1 - \hat{\pi}) \hat{m}, \tag{8}$$

where $\hat{m} = \frac{1}{n} \sum_{i=1}^n \frac{I(y_i > 0)}{1 - \hat{\pi}_i} h(\eta_i(\hat{\xi}))$.

Welsh and Zhou [1] proved that, under mild conditions, these two nonparametric estimators were consistent, asymptotic normal. They also gave closed-form expressions for their asymptotic variance.

To state their asymptotic variances, we need some additional notations. Let us define $\mu_i(\xi) = x - \frac{g(\xi)}{g_i(\xi)} x_i$, $\nu_i(\xi) = g^{(1)}(\xi) - \frac{g(\xi)}{g_i(\xi)} g_i^{(1)}(\xi)$ and $\tau_i(\xi) = g^{(2)}(\xi) - \frac{g(\xi)}{g_i(\xi)} g_i^{(2)}(\xi)$, and write $\mu_i(\xi_0) = \mu_i$, $\nu_i(\xi_0) = \nu_i$ and $\tau_i(\xi_0) = \tau_i$ for simplicity. Also, put $\bar{\hat{\mu}}^* = \frac{1}{n} \sum_{i=1}^n (1 - \hat{\pi}_i) \mu_i(\hat{\xi})$, $\bar{\hat{\nu}}^* = \frac{1}{n} \sum_{i=1}^n (1 - \hat{\pi}_i) \nu_i(\hat{\xi})$ and $\bar{\hat{\tau}}^* = \frac{1}{n} \sum_{i=1}^n (1 - \hat{\pi}_i) \tau_i(\hat{\xi})$. Define $\hat{B}_\beta$ and $\hat{B}_\theta$ by

$$\begin{pmatrix} \hat{B}_\beta \\ \hat{B}_\theta \end{pmatrix} = \left\{ -\frac{1}{n} \sum_{i=1}^n E \frac{\partial}{\partial \xi} \Psi_i(Y_i, \xi) \Big|_{\xi = \hat{\xi}} \right\}^{-1}$$

and set

$$\hat{w}^* = \begin{pmatrix} 1 \\ \hat{B}_\beta^T \left\{ \left(\frac{1}{n} \sum_{i=1}^n \frac{I(Y_i > 0)}{1 - \hat{\pi}_i} h'(\eta_i(\hat{\xi})) \right) \bar{\hat{\mu}}^* + \left(\frac{1}{n} \sum_{i=1}^n \frac{I(Y_i > 0)}{1 - \hat{\pi}_i} e_i(\hat{\xi}) h'(\eta_i(\hat{\xi})) \right) \bar{\hat{\nu}}^* \right\} \\ \hat{B}_\theta^T \left\{ \left(\frac{1}{n} \sum_{i=1}^n \frac{I(Y_i > 0)}{1 - \hat{\pi}_i} e_i(\hat{\xi}) h'(\eta_i(\hat{\xi})) \right) \bar{\hat{\tau}}^* \right\} \end{pmatrix},$$

$$\hat{\Omega}_i^* = \begin{pmatrix} I(Y_i > 0) h(\eta_i(\hat{\xi})) - (1 - \hat{\pi}_i) \frac{1}{n} \sum_{j=1}^n \frac{I(Y_j > 0)}{1 - \hat{\pi}_j} h(\eta_j(\hat{\xi})) \\ \Psi_i(Y_i, \hat{\xi}) \\ \Psi_i(Y_i, \hat{\xi}) \end{pmatrix},$$

$$\hat{\Sigma}^* = \frac{1}{n}\sum_{i=1}^{n} \hat{\Omega}_i^* \hat{\Omega}_i^{*T},$$

$$\hat{d}^* = \left\{ \frac{1}{n}\sum_{i=1}^{n} \frac{I(Y_i > 0)}{1-\hat{\pi}_i} h(\eta_i(\hat{\xi})) \right\} \left\{ \frac{x}{l'(\hat{\pi})} - \frac{1}{n}\frac{1-\hat{\pi}}{1-\bar{\hat{\pi}}} \sum_{i=1}^{n} \frac{x_i}{l'(\hat{\pi}_i)} \right\}$$

and

$$\hat{A} = \frac{1}{n}\sum_{i=1}^{n} x_i x_i^T \frac{1}{l'(\hat{\pi}_i)^2 \hat{\pi}_i (1-\hat{\pi}_i)}.$$

Then we can estimate the asymptotic variance of the externally weighted estimator $\hat{u}^*$ by

$$\hat{v}^* = \frac{1}{n}\left(\frac{1-\hat{\pi}}{1-\bar{\hat{\pi}}}\right)^2 \hat{w}^{*T} \hat{\Sigma}^* \hat{w}^* + \frac{1}{n}\hat{d}^{*T} \hat{A}^{-1} \hat{d}^*.$$

An approximate $100(1-\gamma)\%$ confidence interval for u is given by

$$\left[\hat{u}^* - \Phi^{-1}(1-\gamma/2)\sqrt{\hat{v}^*}, \quad \hat{u}^* + \Phi^{-1}(1-\gamma/2)\sqrt{\hat{v}^*}\right],$$

where Φ is the cumulative distribution function of the standard normal distribution.

Similarly, the asymptotic variance of the internally weighted estimator $\hat{u}$ can be estimated by

$$\hat{v} = \frac{1}{n}(1-\hat{\pi})^2 \hat{w}^T \hat{\Sigma} \hat{w} + \frac{1}{n}\hat{d}^T \hat{A}^{-1} \hat{d},$$

and an approximate $100(1-\gamma)\%$ confidence interval for u is given by

$$\left[\hat{u} - \Phi^{-1}(1-\gamma/2)\sqrt{\hat{v}}, \quad \hat{u} + \Phi^{-1}(1-\gamma/2)\sqrt{\hat{v}}\right],$$

where

$$\hat{w} = \begin{pmatrix} 1 \\ \hat{B}_\beta^T \left\{ \left(\frac{1}{n}\sum_{i=1}^n \frac{I(Y_i>0)}{1-\hat{\pi}_i} h'(\eta_i(\hat{\xi}))\right) \left(\frac{1}{n}\sum_{i=1}^n \mu_i(\hat{\xi})\right) + \left(\frac{1}{n}\sum_{i=1}^n \frac{I(Y_i>0)}{1-\hat{\pi}_i} e_i(\hat{\xi}) h'(\eta_i(\hat{\xi}))\right) \left(\frac{1}{n}\sum_{i=1}^n \nu_i(\hat{\xi})\right) \right\} \\ \hat{B}_\theta^T \left\{ \left(\frac{1}{n}\sum_{i=1}^n \frac{I(Y_i>0)}{1-\hat{\pi}_i} e_i(\hat{\xi}) h'(\eta_i(\hat{\xi}))\right) \left(\frac{1}{n}\sum_{i=1} \tau_i(\hat{\xi})\right) \right\} \end{pmatrix},$$

$$\hat{\Omega}_i = \begin{pmatrix} \frac{I(Y_i>0)}{1-\hat{\pi}_i} h(\eta_i(\hat{\xi})) - \frac{1}{n}\sum_{j=1}^n \frac{I(Y_j>0)}{1-\hat{\pi}_j} h(\eta_j(\hat{\xi})) \\ \Psi_i(Y_i, \hat{\xi}) \\ \Psi_i(Y_i, \hat{\xi}) \end{pmatrix},$$

$$\hat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} \hat{\Omega}_i \hat{\Omega}_i^T,$$

and

$$\hat{d} = \left\{ \frac{1}{n} \sum_{i=1}^{n} \frac{I(Y_i > 0)}{1 - \hat{\pi}_i} h(\eta_i(\hat{\xi})) \right\} \left\{ \frac{x}{l'(\hat{\pi})} - \frac{1}{n}(1 - \hat{\pi}) \sum_{i=1}^{n} \frac{x_i}{(1 - \hat{\pi}_i) l'(\hat{\pi}_i)} \right\}.$$

3 Computer Program

We develop a computer program in R, a language and environment for statistical computing and graphics [4]. Our program reads in patient data files in plain text format, estimates regression parameters $\hat{\alpha}$, $\hat{\beta}$, and $\hat{\theta}$, then computes both externally weighted estimator $\hat{u}^*$ and internally weighted estimators $\hat{u}$ of mean on the original scale. In addition to these two mean estimates, our program also outputs their statistics, including standard deviations, asymptotic confidence intervals, and an option for bootstrap confidence intervals.

Our program can run in both interactive mode and batch mode. In the interactive mode, the program guides users to specify computation settings step by step. In the batch mode, computation is activated by a single line of command. Users have more control and more flexibility, but may have to bear the trouble of writing R code. These two modes are discussed in detail in the following two subsections.

3.1 Interactive Mode

The interactive mode program is activated by command

```
source("interactive.r").
```

The program then guides users to specify computation settings step by step. In each step, a brief prompt is displayed on the screen. The program reads in users' answer, verifies that the answer is semantically correct, then moves to the next step. A part of the screen displaying an example is shown in Figure 1. The details of these settings are discussed as following:

- The program starts from prompting users to input a patient data file name, then opens the file and reads in the data. The input data file name is also used to name output files. Each output file name has two parts: the first part is the same as input file name; the second part is a proper postfix indicating the content of this output file.
- Users have flexibility to choose how to estimate regression parameters β and θ. The first choice is to directly maximize $\ell_2(\xi)$ as in (5). This is the default choice in this program, because the maximization is performed by using built-in R function `Optim`, which is numerically more stable. The second choice is to solve estimating equation (6). This option enables us to extend MLE estimator to a wide class of estimators such as pseudo likelihood estimators mentioned in Sec. 2.2.
- Users can give an initial guess of regression parameters α, β, and θ. Closer the guessed values are to the true values, faster convergence the process of estimating these parameters can achieve. If no initial guess is given to a parameter or part of a parameter, the program will set default values to be zero.
- Users can specify the confidence level. Default value is 95%.

When users decide to calculate bootstrap confidence interval, there are two extra settings:

- The first one is a bootstrap sample size. Default size is 100.
- The Bootstrap process needs to generate random numbers. Users can specify a seed for a random number generator. If no seed is given, then the program will pick system time as the seed.

Next, users are prompted to give a value for the vector of covariates x. This is an iterative loop. For each given x, the program calculates estimates for the mean cost and their statistics, then prompts for another covariate. In this pattern, users can compare results for multiply covariates in a single run. The example screen display is shown in Figure 2.

3.2 Batch Mode

Our program can also run in the batch mode by a command

```
source("batch.r").
```

In this mode, there has no interactions between users and the program. All computational settings have to be hard coded in the source file `barch.r`. However, users may gain a high degree of flexibility.

Regression models (2) and (3) are shown in the format that parameters are related with all covariates. However, in real applications, each parameter may only related to some components of covariates. Our program can handle this situation easily. Let us use parameter α as an example. There is an indicator vector `XalphaIndicator` and an integer `nAlpha` in the program. `XalphaIndicator` has the same length as the covariate vector with all components being either 0 or 1. When one covariate is used to estimate α, the corresponding component of `XalphaIndicator` is set to be one. Otherwise, it is set to be zero. The value of `nAlpha` is the total number of components of `XalphaIndicator` being one. For example, if users want to specify (2) as

$$l(\pi_i) = x_1\alpha_1 + x_3\alpha_3,$$

they only need to assign $\texttt{XalphaIndicator} = (1, 0, 1, 0, \ldots, 0)$ and $\texttt{nAlpha} = 2$. For parameters β and θ, the pairs $(\texttt{XbetaIndicator}, \texttt{nBeta})$ and $(\texttt{XgammaIndicator}, \texttt{nGamma})$ play the similar roles.

Our program also has flexibility to specify the transformation function h, mean function on the transformed scale (appearing as $x^T\beta$ in (3)), and variance g on the transformed scale. All related functions are listed in Table 1.

3.3 Input and Output

An input data file to our program is a ASCII text file containing a data table. An example of an input data file is shown in Figure 5. A single line of the file is a record. Columns are white-space (spaces and/or tabs) delimited. The first column is the observation of the response variable. Remaining columns are observations of covariates. Our program has no requirement on the number of data lines and the number of columns. They are determined automatically while the program reading in data. Hence, our program works for various applications with different numbers of covariates without any modification.

Our program produces two output files whose names are determined by an input data file name. Assume the input data file is named `example.dat`. Then the output files are `example_par.dat` and `example_result.dat`.

The output file `example_par.dat` shows estimations and standard deviations of regression parameters α, β, and θ. An example of `example_par.dat` is shown in Figure 4.

The output file `example_result.dat` shows mean estimators, their standard derivations, confidence intervals, and optional bootstrap confidence intervals, all on the original scale. If multiple covariates are provided, then statistics for all covariates are shown in this single file to make comparisons more convenient. Following the example shown in Figure 2, the content of the output file `example_result.dat` is illustrated in Figure 5.

4 Application Example

We illustrate the use of our computer program in a data set on hypertension patients from a prospective drug utilization review (DUR) study [3]. A goal of our analysis is to estimate the expected in-patient charge of a patient given his/her age, gender, race, and general health status on in-patient charges. Since in-patient charges are zero for some patients, we apply a two-stage heteroscedastic regression model to our data set.

Let Y_i be the in-patient charge of the ith patient, and corresponding covariates are defined as follows. X_{i1} is the age of the patient; X_{i2} represents the patient's race; X_{i3} represents the gender of the patient ($X_{i3} = 1$ for males and $X_{i3} = 0$ for females); X_{i4} is the score based on 100 representing ith patient's general health status. Then, for $i = 1, \ldots, n$, we model the probability of non-zero in-patient charges by the logistic regression model,

$$\log \frac{P(Y_i = 0 \mid X_{i1}, \ldots, X_{i4})}{P(Y_i > 0 \mid X_{i1}, \ldots, X_{i4})} = \alpha_0 + \alpha_1 X_{i1} + \ldots \alpha_4 X_{i4}, \tag{9}$$

and we model the conditional magnitude of the positive charges Y_i given $Y_i > 0$ by the log-transformed, heteroscedastic linear regression model,

$$\log Y_i = \beta_0 + \beta_1 X_{i1} + \ldots + \beta_4 X_{i4} + \exp\{(\theta_0 + \theta_1 X_{i1} + \ldots + \theta_4 X_{i4})/2\}\epsilon_i. \tag{10}$$

Our data set consists of 483 patients. Each data line is a patient record, starting from his/her in-patient charge Y_i and followed by covariates $X_{i1}, \ldots, X_{i4}$. A part of data set is illustrated in Figure 5. The estimates of parameters in (9) and (10) and their standard deviations are given in Figure 4. Using these parameter estimates, we can estimate the average charge of a patient with the given covariate values and the associated standard deviation. As shown in Figure 5, for a 55-year old white female patient with the general health score of 50, the estimated average charge is $\hat{u}_0^* = \$1156.67$ using the externally weighted estimator with 95% confidence interval of ($489.85,$1823.49) and $\hat{u}_0 = \$1160.98$ using the internally weighted estimator with 95% confidence interval of ($485.81,$1836.16).

5 Discussion

We developed a computer program for modelling skewed, heteroscedastic data with zero observations. The program is an implementation of the two-part regression model proposed by Welsh and Zhou (2005). It computes two nonparametric estimates of the mean cost, their asymptotic standard derivations, asymptotically normal-based confidence intervals, and optional bootstrap confidence intervals.

The program can be run in both the used friendly interactive mode and more efficient batch mode. It also provides flexibility for users to extend the program to a more general context. The two-part regression model can be generalized by re-coding some of modularized functions. For parameters in the two-part regression model, users can choose from different estimates and/or different numerical approaches.

Our program is written in free statistical computing language R and can be run on a wide variety of platforms. The program is available from *http://faculty.washington.edu/ azhou.*

ACKNOWLEDGEMENT

This work was supported in part by AHRQ grant R01HS013105. It does not necessarily represent those of VA HSR&D Service.

References

[1] A. H. Welsh and X. H. Zhou (2005). Estimating the retransformed mean in a heteroscedastic two-part model, Journal of Statistical Planning and Inference, in press.

[2] N. Duan (1983). Smearing estimate: a nonparametric retransformation method, Journal of the American Statistical Association 78, 605–610.

[3] D. Murray, L. E. Harris, M. Overhage, X. H. Zhou, G. J. Eckert, F. E. Smith, N. N. Buchanan, F. D. Wolinsky, C. J. McDonald, and W. M. Tierney (2005). Failure of Computerized Treatment Suggestions to Improve the Health Outcomes of Outpatients with Uncomplicated Hypertension: Results of a Randomized Controlled Trial Michael, Archives of Internal Medicine, in press.

[4] R Development Core Team (2004). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org.

```
Please specify the name of input data file: example.dat

Please choose method for estimating parameters beta and theta
  1: MLE estimator, maximize log-likelihood function (default)
  2: MLE estimator, solve estimating equation
1

Do you want to give initial guess of parameters (default: all zeros)? (Y/N)n
alpha.0 = 0 0 0 0 0
beta.0  = 0 0 0 0 0
gamma.0 = 0 0 0 0 0

Please specify (1-r) confidence interval (default r=0.05):
  r =
  r =  0.05

Do you want to calculate bootstrap confidence interval? (Y/N)y

Please specify bootstrap sample size (default 100): 100

Do you want to assign seed for random number generator? (Y/N)n
Seed not assigned.
```

Figure 1: Example of screen display in interactive mode: part I.

```
Please input covariates values:
1: 55 1 0 50
Covariate =  55 1 0 50

Do you want to see estimation results for another covariate? (Y/N) y

Please input covariates values:
1: 65 1 0 50
Covariate =  65 1 0 50

Do you want to see estimation results for another covariate? (Y/N) n
```

Figure 2: Example of screen display in interactive mode: part II.

```
0        56 0 0 32
0        64 1 1 25
1952.05 68 1 1 42
0        54 1 1 50
......
```

Figure 3: Example of input data file.

```
estimator of parameter alpha =
1.513531 -0.008025662 0.4702371 0.3740364 0.007434743

std of alpha estimator =
0.7334805 0.01321834 0.3182745 0.2856736 0.006093177

estimator of parameter beta =
9.538691 -0.004213263 -0.8231364 0.02954837 0.003566381

std of beta estimator =
0.701519 0.0126878 0.3566817 0.2814012 0.005726485

estimator of parameter theta =
-0.9736211 0.05375912 -1.048779 -0.5785688 -0.01572478

std of theta estimator =
0.9676426 0.01870368 0.4402539 0.3763115 0.00826068
```

Figure 4: Example of output file for regression parameters.

Function name	Notation	Description
fh	h	transformation function h in (3)
fhinv	h^{-1}	inverse function of h
dfh	h'	derivative of h
fm	m	mean function in transformed space, appearing as $x^T\beta$ in (3)
dfm.beta	$\partial m/\partial\beta$	first order partial derivative of m w.r.t. β
fg	g	variance function in transformed space
logfg	$\log(g)$	logarithm of g
dfg.beta	$\partial g/\partial\beta$	first order partial derivative of g w.r.t. β
dfg.gamma	$\partial g/\partial\theta$	first order partial derivative of g w.r.t. θ
d2fg.beta	$\partial^2 g/\partial\beta^2$	second order partial derivative of g w.r.t. β
d2fg.betagamma	$\partial^2 g/\partial\beta\partial\theta$	second order partial derivative of g w.r.t. β and θ
d2fg.gammabeta	$\partial^2 g/\partial\theta\partial\beta$	second order partial derivative of g w.r.t. θ and β
d2fg.gamma	$\partial^2 g/\partial\theta^2$	second order partial derivative of g w.r.t. θ

Table 1: List of functions in regression model.

```
Source data file:        example.dat
Number of observations:  483
Number of covariates:    4
----------------------------------------------------------

Covariate = 55 1 0 50

Externally weighted estimator:
                             mean = 1156.673
               standard deviation = 340.2204
          95% confidence interval = [489.8531, 1823.493]
95% bootstrap confidence interval = [278.9567, 2667.711]

Internally weighted estimator:
                             mean = 1160.983
               standard deviation = 344.4845
          95% confidence interval = [485.8061, 1836.160]
95% bootstrap confidence interval = [401.0650, 2736.815]
----------------------------------------------------------

Covariate = 65 1 0 50

Externally weighted estimator:
                             mean = 1544.924
               standard deviation = 539.8988
          95% confidence interval = [486.7414, 2603.106]
95% bootstrap confidence interval = [685.6556, 4059.045]

Internally weighted estimator:
                             mean = 1556.870
               standard deviation = 555.9018
          95% confidence interval = [467.3222, 2646.417]
95% bootstrap confidence interval = [722.824, 4266.309]
----------------------------------------------------------
```

Figure 5: Example of output file for summary statistics.