MODEL VALIDATION AND DIAGNOSTIS IN RIGHT CENSORED REGRESSION

A Dissertation
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Tatjana Miljkovic

In Partial Fulfillment
for the Degree of
DOCTOR OF PHILOSOPHY

Major Department:
Statistics

May 2013

Fargo, North Dakota

North Dakota State University
Graduate School

Title

MODEL VALIDATION AND DIAGNOSTICS IN RIGHT CENSORED REGRESSION

By

Tatjana Miljkovic

The Supervisory Committee certifies that this ***disquisition*** complies with North Dakota State University's regulations and meets the accepted standards for the degree of

DOCTOR OF PHILOSOPHY

SUPERVISORY COMMITTEE:

Dr. Megan Orr
Chair

Dr. Rhonda Magel

Dr. Seung Won Hyun

Dr. Nikita Barabanov

Approved:

May 31, 2013.
Date

Dr. Rhonda Magel
Department Chair

ABSTRACT

When censored data are present in the linear regression setting, the Expectation-Maximization (EM) algorithm and the Buckley and James (BJ) method are two algorithms that can be implemented to fit the regression model. We focus our study on the EM algorithm because it is easier to implement than the BJ algorithm and it uses common assumptions in regression theory, such as normally distributed errors. The BJ algorithm, however, is used for comparison purposes in benchmarking the EM parameter estimates, their variability, and model selection.

In this dissertation, validation and influence diagnostic tools are proposed for right censored regression using the EM algorithm. These tools include a reconstructed coefficient of determination, a test for outliers based on the reconstructed Jackknife residual, and influence diagnostics with one-step deletion.

To validate the proposed methods, extensive simulation studies are performed to compare the performances of the EM and BJ algorithms in parameter estimation for data with different error distributions, the proportion of censored data, and sample sizes. Sensitivity analysis for the reconstructed coefficient of determination is developed to show how the EM algorithm can be used in model validation for different amounts of censoring and locations of the censored data. Additional simulation studies show the capability of the EM algorithm to detect outliers for different types of outliers (uncensored and censored), proportions of censored data, and the locations of outliers. The proposed formula for the one-step deletion method is validated with an example and a simulation study.

Additionally, this research proposes a novel application of the EM algorithm for modeling right censored regression in the area of actuarial science. Both the EM and BJ algorithms are utilized in modeling health benefit data provided by the North Dakota Department of Veterans Affairs (ND DVA). Proposed model validation and diagnostic tools are applied using the EM algorithm. Results of this study can be of great benefit to government policy makers and pricing actuaries.

ACKNOWLEDGMENTS

This research could not be finished without the support and contributions of the following people to whom I am very grateful:

Kelly Schmidt, State Treasurer of North Dakota, and Lonnie Wangen, Commissioner of North Dakota Department of Veterans Affairs for providing the data used in this dissertation.

Committee members starting with my advisor: Dr. Megan Orr, Dr. Nikita Barabanov, Dr. Rhonda Magel, and Dr. Seung W. Hyun for their valuable guidance and feedback.

Dr. Volodymyr Melnykov for teaching me about the EM algorithm and reviewing the initial programing of this project in R.

Most of all my husband Dragan, son Petar, and daughter Kristina for their patience, encouragement, and support in pursuing Ph.D degree.

DEDICATION

Dedicated to my parents Zdravka and Radisav Jevtic

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1. INRODUCTION

Censoring has been extensively discussed as part of survival analysis and a large volume of literature is generated in this area. Good information on these topics can be found in books by Klein and Moeschberger (2003) and Lee (1997). An observation is right censored at a censoring point if when it is above the censoring point, it is recorded as being equal to the censoring point, but when it is below the censoring point, it is recorded as its observed value. In medical statistics, right censoring is analyzed from the data of patients who are still alive at the end of the study and those who terminated the study due to surrender (Miller 1976). In the insurance industry, some policies are structured in such a way that the policy limits serve as a restricted amount of payment on a given loss. For a loss below or equal to the policy limit, payment is made in the amount equal to the loss. If the loss exceeds the policy limit, payment is imputed at the policy limit (Guiahi 2001).

Linear regression models are commonly used in many applications to analyze the functional relationship between a response variable and other explanatory variables that are perceived to be related to the response variable. Typically, a normal distribution is assumed for the underlying assumption of the error structure. However, these models have limitations when the response variable is right censored since they may yield fitted values of the variable of interest to exceed its upper or lower bound when the censoring is ignored. The most popular semi-parametric and non-parametric models for right censored regression are the Cox (1972) and Buckley-James (BJ) (1979) models, respectively. These two models are available commercially. Currently, the normal model for right censored regression based on the Expectation-Maximization (EM) method introduced by Dempster, Laird, and Rubin (1977) is not implemented in any statistical software. Diagnostic tools for this model have not been

sufficiently developed. This is surprising because the right censored regression based on the EM algorithm is easy to implement and uses the same assumptions that are used in regression theory.

One of the goals of a pricing actuary, when dealing with censored losses, is to develop the best fitting model based on the historical data reported by claims. Data available to an actuary includes: individual losses, the information about the coverage limits, and axillary policy information on rating variables. Currently, actuaries group losses by loss size and their risk attributes when they deal with censored losses. For each group they develop a separate loss distribution. Developing a right censored regression model for losses in the presence of multiple rating variables using the EM algorithm would be of a great benefit to a pricing actuary because this would allow for modeling individual losses. Model validation and diagnostics based on the EM algorithm would allow for better understanding of the model fit, quantifying the influence of the individual observations on the parameter estimates, and detecting of outliers.

The organization of this dissertation is as follows. Chapter 2 reviews literature in the area of right censored regression. Chapter 3 introduces notation used in the subsequent chapters. Chapter 4 describes the EM method. The proposed EM model, parameter estimates, and variability assessment for right censored regression are developed in Chapter 5. The BJ method is summarized in Chapter 6 including the proposed model, parameter estimates, variability assessment, and model selection. Chapter 7 provides theoretical development of proposed validation and diagnostic tools for the EM algorithm. Extensive simulations studies are discussed and the results are reported in Chapter 8. Chapter 9 includes an analysis of data provided by the North Dakota Department of veterans Affairs. Finally, the conclusion is presented in Chapter 10.

CHAPTER 2. REVIEW OF LITERATURE

The Cox Model (1972), the most popular model in survival analysis, has been used extensively to link censored failure times of the variable of interest to a set of related explanatory variables (covariates). The hazard function (age-specific failure rate) models the response variable "non-parametrically" or "parametrically" as a function of time, while the set of covariates form a regression model in which these variables are modeled "parametrically". Some applications of the model are considered in non-life insurance, such as the occurrence of claims (Keiding 1998) or censored payments from property losses that can be explained by some individual's characteristics (Klugman, Panjer, and Willmot 2004).

Non-parametric regression models with right censored responses were originally studied by Miller (1976), Buckley and James (1979), and Koul, Susarla, and Ryzin (1981). Miller developed a Kaplan-Meier Least Squares estimator which minimizes the weighted sum of squares of the residuals. The weights are obtained using the Kaplan-Meier (1958) estimator, well known in survival analysis. Buckley and James developed an estimator known as the BJ estimator which is based on the normal equations. This method is a special case of the quasi-likelihood method incorporated in a framework of Generalized Linear Models (McCullagh and Nelder 1983). Kaplan-Meier estimates replace the censored observations, and the inferences about the parameters of the model are made using quasi-likelihood, which requires assumptions on the first two moments of the data (Wedderburn 1974) and (Yu, Yu, and Liu 2009).

While the estimators of Miller and Buckley and James both use an iterative procedure, Koul, Susarla, and Ruzin (1981) proposed an estimator which is obtained without an iterative procedure. However, this estimator is based on the assumption that the distribution of the

censored variable does not depend on the covariates. In practice, the dependent variable may be sensitive to this assumption. All of these models were mostly applied in area of survival analysis with applications in biostatistics and medical research.

Early studies on parametric methods for right censored regression were dated in the 1970s. Dempster, Laird, and Rubin (1977) proposed an iterative procedure known as the EM Algorithm. The EM algorithm has been extensively used for missing data or data containing missing values. Good information on the EM methodology and the applications can be found in McLachlan and Krishnan (2007).

Schmee and Hahn (1979) analyzed right censored regression data on electrical insulation in 40 motorettes tested at four different temperature settings. They recorded the time until failure in hours of each motorette. Observations were right censored if the motorettes were still on test without failure at the indicated time. An iterative least square (ILS) method was used to estimate the parameters of the simple linear regression model where the response variable was right censored and errors were assumed to be normally distributed. Parameter estimates in each step were computed by least squares using the uncensored observations and the conditional expectations of the censored observations. They also showed that the ILS estimates performed as well as those obtained with maximum likelihood estimation studied by Hahn and Nelson (1974), for the same data. The ILS method was prized for computational simplicity and attractiveness to non-statisticians as it is easier to explain.

Aitkin (1981) showed that the parameter estimates for the same data (40 motorettes) can be obtained by maximum likelihood using the EM algorithm. In the E-step, censored observations were replaced with their conditional expectations given the observed data and the

current parameter estimates. Then in the M-step, the new parameter estimates were computed by the maximum likelihood method based on the complete data.

In 1977, Cook developed a measure based on confidence ellipsoids, which is useful in assessing the influence of the *i*th data point on the estimated regression coefficient. This measure is known as "Cook's Distance". Cook's Distance measures the distance between the estimated regression coefficient and that obtained when the *i*th point is deleted from the sample. Cook's Distance is based on the confidence ellipsoid formula for the unknown vector of coefficients, which follows the *F* distribution. The observations with Cook's distance at or above the 50^{th} percentile of the *F* distribution are considered influential. Various approaches have been studied by Chatterjee and Hadi (1986) to check model validity and influential observations in standard linear regression. Aziz and Wang (2009) developed "Cook's Distance" for the BJ model, which extended the application of this model to validation and influence diagnostics.

Although, the EM algorithm for right censored regression has been studied, it is rarely used largely due to the fact that it is not readily available in statistical software packages. Additionally, diagnostic tools for the EM algorithm have not been developed extensively. Weissfeld and Schneider (1990) proposed models for assessing the influence of a single observation on the estimation of coefficients in the normal model with censored data. Their methods include the empirical influence function and one-step deletion methods based on the Newton- Raphson and EM algorithms. However, their formula developed for one-step deletion method in the case of the EM algorithm is different than the one proposed in this research.

CHAPTER 3. NOTATION

The notations that are used consistently throughout this dissertation are defined here. They follow the same convention used in the book *Linear Models in Statistics* by Rencher and Schaalje (2008). Other notations that are less frequently used will be defined later when they are introduced for the first time. In terms of script, font, and style, the following is defined:

- Parameters - not bold, lower-case Greek letter
- Parameter space - not bold, upper-case Greek letter
- Sample space - not bold, upper-case Greek letter
- Random variable – not-bold, upper-case italic Roman letter
- Vectors – bold and lower-case
- Observation – not bold, lower-case italic Roman letter
- Matrices - bold, upper-case Roman letter
- Functions - not bold, lower-case italic Roman letter with parenthesis ( )
- Scalars – not bold, lower-case italic Roman letter

Mathematical symbols for vectors, functions, matrices, and sizes of datasets are consistent with common statistical practices. They are defined as follows:

n – the total number of observations

m- the number of censored observations

p – the number of explanatory variables in the regression model

$\boldsymbol{y} = (y_1, \ldots, y_n)'$ - the observed-data vector

$\boldsymbol{z} = (z_1, \ldots, z_n)'$ - the unobservable or incomplete-data vector

$\boldsymbol{\Psi} = (\Psi_1, \ldots, \Psi_d)'$ - d-dimensional parameter vector

$L(\boldsymbol{\Psi})$ – the likelihood function for $\boldsymbol{\Psi}$ formed based on observed data $y_1, \ldots, y_n$

$L_c(\boldsymbol{\Psi})$ – the complete-data likelihood function for $\boldsymbol{\Psi}$ formed based on observed data $y_1, \ldots, y_n$

$l_c(\boldsymbol{\Psi})$ – the complete-data log likelihood function for $\boldsymbol{\Psi}$ formed based on observed data $y_1, \ldots, y_n$

Q-function- conditional expectation of log $L_c(\boldsymbol{\Psi})$ given the observed-data

J- objective function

$\mathbf{X}$ – $n \times (p+1)$- dimensional matrix of explanatory variables in multiple regression

$\mathbf{X}_1$ – *(n-m)* x $(p+1)$- dimensional partition matrix of explanatory variables in multiple regression containing the observed data

$\mathbf{X}_2$ – *m* x $(p+1)$- dimensional partition matrix of explanatory variables in multiple regression containing the observed data

$\boldsymbol{\varepsilon}$ – random error vector

$\boldsymbol{\beta}$ – vector of coefficients in the linear model

$\widehat{\boldsymbol{\beta}}$ – maximum likelihood estimates of $\boldsymbol{\beta}$

$\varphi(x)$ – probability density function of $X \sim N(0,1)$

$\Phi(\mathrm{x})$ – cumulative distribution function of $X \sim N(0,1)$

CHAPTER 4. EXPECTATION MAXIMIZATION METHOD

The EM Method is broadly introduced by Dempster, Laird, and Rubin (1977). They call this the "EM Algorithm". This method is an iterative climbing approach for computing maximum likelihood estimates in the presence of incomplete-data (e.g. missing observations, truncated or censored data). The EM methodology is often confused with the "EM Algorithm" when in fact there are many examples of EM algorithms built on the idea of the EM method.

Several sample spaces exist in the presence of incomplete data. The observed data $\boldsymbol{y}_{\mathrm{o}}$ are realizations of the random variable Y_0 having sample space $\mathcal{Y}o$. We have only incomplete information about the remaining data, $\boldsymbol{y}_u$, which is presented by the observation vector $\mathbf{z}$. That is, for some function f_u we have $\boldsymbol{z} = f_u(\boldsymbol{y}_u)$ **.** Denote by Y_{o}, Y_u, and Z the corresponding random variables. Next, we introduce the total data vector $\boldsymbol{y}_t = (\boldsymbol{y}_o, \boldsymbol{y}_u)$. Now define

$Y_{\mathrm{t}} = (Y_{\mathrm{o}}, Y_u)$. The sample space of corresponding random variable Y_t we denote by $\mathcal{L}$.

Obviously, $\mathcal{L} = \mathcal{Y}o \times \mathcal{Y}_u$, where $\mathcal{Y}_u$ is the sample space of the random variable Y_u**.** We also introduce a vector $\boldsymbol{y}_c = (\boldsymbol{y}_{\mathrm{o}}, \mathbf{z})$, which we call the complete observation vector, and corresponding random variable Y_{c}. Denote by $\mathcal{Z}$ the sample space of $\mathbf{z}$. Denote by $\mathcal{L}(\boldsymbol{z})$ the set of all values $\boldsymbol{y}_u \in \mathcal{Y}_{\mathrm{u}}$ such that $f_u(\boldsymbol{y}_u) = \mathbf{z}$. Assume $\boldsymbol{\Psi}$ contains all parameters of the p.d.f. f_t of the random variable Y_t**;** that is $f_t = f_t(\boldsymbol{y}_o, \boldsymbol{y}_u, \boldsymbol{\Psi})$. Then for every $\boldsymbol{y}_c = (\boldsymbol{y}_o, \mathbf{z})$,

$$f(\boldsymbol{y}_c, \boldsymbol{\Psi}) = \int_{\boldsymbol{y}_u} f_t(\boldsymbol{y}_o, \boldsymbol{y}_u, \boldsymbol{\Psi})\, dy_u$$

is a p.d.f. of the random variable Y_{c} with sample space $\mathcal{Y}\mathrm{o} \times \mathcal{Z}$ and vector $\boldsymbol{\Psi}$ of parameters. The vector $\boldsymbol{\Psi}$ is not known. The goal of the EM method is to find a reasonable approximation of this vector.

The EM method consists of two steps that are applied iteratively: the E-step and M-step. During the E-step, we find the conditional expectation, or Q-function, of the complete-data log likelihood, L_c given the observed data $\boldsymbol{y}$, based on the current fit for $\boldsymbol{\Psi}$. Let $\boldsymbol{\Psi}^{(\mathbf{0})}$ be an initial value for $\boldsymbol{\Psi}$. The E-step consists of computing $Q = \boldsymbol{E}(\log L_{\boldsymbol{c}} | \boldsymbol{y}_c, \boldsymbol{\Psi})$, the conditional expectation of $\log L_{\boldsymbol{c}}$ given the observed data $\boldsymbol{y}$ with respect to f_t with parameters $\boldsymbol{\Psi}$.

Denote by $G_u(\boldsymbol{\Psi})$ a function of parameters of the p.d.f. of $\boldsymbol{y}_u$ such that, for some function P, we have $Q = P\big(\boldsymbol{y}_c, \boldsymbol{\Psi}, G_u(\boldsymbol{\Psi})\big)$. In this dissertation, $G_u(\boldsymbol{\Psi})$ represents the first and second moments of the censored observation $\boldsymbol{y}_u$ given $\boldsymbol{y}_u > z$.

During the E-step of the first iteration, we compute $G_u\big(\boldsymbol{\Psi}^{(0)}\big)$. During the M-step, we compute the maximum of $P\left(\boldsymbol{y}_c, \boldsymbol{\Psi}, G_{\boldsymbol{u}}\big(\boldsymbol{\Psi}^{(\mathbf{0})}\big)\right)$ with respect to $\boldsymbol{\Psi}$. The value $\boldsymbol{\Psi}^{(\mathbf{1})}$ of the argument of the maximum is considered as a new approximation of the set of unknown parameters $\boldsymbol{\Psi}$. On the next iteration, during the E-step, we compute again the value of $G\big(\boldsymbol{\Psi}^{(\mathbf{1})}\big)$. During the M-step, we maximize $P\left(\boldsymbol{y}_c, \boldsymbol{\Psi}, G_u\big(\boldsymbol{\Psi}^{(\mathbf{1})}\big)\right)$ with respect to $\boldsymbol{\Psi}$. On the kth iteration, during the E-step, we compute the value of $G_u\big(\boldsymbol{\Psi}^{(\boldsymbol{k-1})}\big)$. During M-step, we find the argument $\boldsymbol{\Psi}^{(\boldsymbol{k})}$ which maximizes $P\left(\boldsymbol{y}_c, \boldsymbol{\Psi}, G_u\big(\boldsymbol{\Psi}^{(\boldsymbol{k-1})}\big)\right)$.

The E-step and M-step are repeated until the relative difference between the values of the log likelihood function between two sequential iterations changes by a sufficiently small number.

CHAPTER 5. EM ALGORITHAM FOR RIGHT CENSORED REGRESSION

5.1. Proposed Model

Consider the traditional form of the multiple regression model; that is,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

where $\boldsymbol{\varepsilon} \sim N(0, \mathbf{I}\sigma^2)$ and $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots\ldots \beta_p)'$ is a vector of unknown parameters.

The matrix, referred to as the design matrix, $\mathbf{X}$ is of size $n \times (p+1)$ and is assumed to have rank equal to $p+1$ (full column rank). The goal of traditional multiple regression is to estimate the parameter vector $\boldsymbol{\psi} = (\beta_0, \beta_1, \ldots\ldots \beta_p, \sigma^2)'$. This can be accomplished by Least-Squares method, which minimizes the sum of squares of deviations for the n observed responses, y_i, from their fitted values, $\widehat{y_i}$.

Now, consider the censored linear regression model,

$$\mathbf{y}^* = \mathbf{X}^*\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

where $\boldsymbol{\varepsilon} \sim N(0, \mathbf{I}\sigma^2)$, $\mathbf{X} = \begin{pmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{pmatrix}$ and $\mathbf{y}^* = \begin{pmatrix} \mathbf{y} \\ \mathbf{z} \end{pmatrix}$**.** The uncensored and censored partition design matrices are denoted by $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively. The uncensored and censored responses are **y** and **z,** respectively.

The true value of the censored response is unknown and is estimated using the conditional expectation during the E-step of the EM algorithm. This estimated value is referred to as the *reconstructed* or *renovated* response value. The EM Algorithm is employed to obtain parameter estimates as well as the reconstructed values of the censored observations.

5.2. Parameter Estimates

The complete likelihood function, based on the complete information for censored regression, is defined as follows:

$$L_c(\boldsymbol{\Psi}, \mathbf{y}, \mathbf{z}) = (2\pi)^{\frac{-n}{2}}(\sigma^2)^{\frac{-n}{2}} e^{-\frac{[(\mathbf{y}-\mathbf{X}_1\boldsymbol{\beta})'(\mathbf{y}-\mathbf{X}_1\boldsymbol{\beta})+(\mathbf{z}-\mathbf{X}_2\boldsymbol{\beta})'(\mathbf{z}-\mathbf{X}_2\boldsymbol{\beta})]}{2\sigma^2}}.$$

The logarithm of this function is defined as l_c. The complete log likelihood is

$$l_c(\boldsymbol{\Psi}, \mathbf{y}, \mathbf{z}) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma^2)$$
$$-\frac{1}{2\sigma^2}[\mathbf{y}'\mathbf{y} - 2\boldsymbol{\beta}'\mathbf{X}_1'\mathbf{y} + \boldsymbol{\beta}'\mathbf{X}_1'\mathbf{X}_1\boldsymbol{\beta} + \mathbf{z}'\mathbf{z} - 2\boldsymbol{\beta}'\mathbf{X}_2'\mathbf{z} + \boldsymbol{\beta}'\mathbf{X}_2'\mathbf{X}_2\boldsymbol{\beta}].$$

Next, we introduce the Q-function:

$$Q(\boldsymbol{\Psi}, \mathbf{z}) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma^2)$$
$$-\frac{1}{2\sigma^2}[\mathbf{y}'\mathbf{y} - 2\boldsymbol{\beta}'\mathbf{X}_1'\mathbf{y} + \boldsymbol{\beta}'\mathbf{X}_1'\mathbf{X}_1\boldsymbol{\beta} + \boldsymbol{B} - 2\boldsymbol{\beta}'\mathbf{X}_2'\boldsymbol{A} + \boldsymbol{\beta}'\mathbf{X}_2'\mathbf{X}_2\boldsymbol{\beta}]. \quad (5.1)$$

Here, $\boldsymbol{A}$ and $\boldsymbol{B}$ are calculated in the E-step as the first and second moments of the conditional expectation for censored observations, given that their values are above the censoring point. It is straightforward to show that

$$\boldsymbol{A} = E(\boldsymbol{z}|\boldsymbol{z} > \boldsymbol{z}, \boldsymbol{\beta}, \sigma^2) = \mathbf{X}_2'\boldsymbol{\beta} + \sigma f\left(\frac{\mathbf{z}-\mathbf{X}_2'\boldsymbol{\beta}}{\sigma}\right) \quad (5.2)$$

$$\boldsymbol{B} = E(\boldsymbol{z}'\boldsymbol{z}|\boldsymbol{z} > \boldsymbol{z}, \boldsymbol{\beta}, \sigma^2) = (\boldsymbol{X}_2'\boldsymbol{\beta})^2 + \sigma(\mathbf{X}_2'\boldsymbol{\beta} + \boldsymbol{z})f\left(\frac{\mathbf{z}-\mathbf{X}_2'\boldsymbol{\beta}}{\sigma}\right) + m\sigma^2, \quad (5.3)$$

where $f(x) = \frac{\varphi(x)}{\Phi(-x)}$, $\varphi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$, and $\Phi(-x) = \int_{-\infty}^{-x} \varphi(s)ds$.

The E-step consists of computing $\boldsymbol{A}$ and $\boldsymbol{B}$. During the next step, the M-step, we maximize the Q-function with respect to parameters $\boldsymbol{\beta}$ and σ using the values $\boldsymbol{A}$ and $\boldsymbol{B}$.

The maximized value of the Q-function will lead to the maximum likelihood estimates (MLEs) for the model. Finding the maximum amounts to finding the solutions to the following equations:

$$\frac{\partial Q}{\partial \boldsymbol{\beta}} = 0 \text{ and } \frac{\partial Q}{\partial(\sigma^2)} = 0.$$

From this, we have

$$\frac{\partial Q(\boldsymbol{\Psi},\boldsymbol{z})}{\partial \boldsymbol{\beta}} = \frac{[\mathbf{X}_1'\mathbf{y} - \mathbf{X}_1'\mathbf{X}_1\boldsymbol{\beta} + \mathbf{X}_2'\boldsymbol{A} - \mathbf{X}_2'\mathbf{X}_2\boldsymbol{\beta}]}{\sigma^2},$$

and therefore

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}_1'\mathbf{y} + \mathbf{X}_2'\boldsymbol{A}) \,. \tag{5.4}$$

Similarly,

$$\frac{\partial Q(\boldsymbol{\Psi}, \mathbf{z})}{\partial(\sigma^2)} = \frac{[\mathbf{y}'\mathbf{y} - 2\boldsymbol{\beta}'\mathbf{X}_1\mathbf{y} + \boldsymbol{\beta}'\mathbf{X}_1'\mathbf{X}_1\boldsymbol{\beta} + \mathbf{B} - 2\boldsymbol{\beta}'\mathbf{X}_2'\mathbf{A} + \boldsymbol{\beta}'\mathbf{X}_2'\mathbf{X}_2\boldsymbol{\beta}]}{\sigma^2} - n$$

and

$$\widehat{\sigma}^2 = \frac{1}{\mathrm{n}}\left[\mathbf{y}'\mathbf{y} + \mathbf{B} + \widehat{\boldsymbol{\beta}}'(\mathbf{X}_1'\mathbf{X}_1 + \mathbf{X}_2'\mathbf{X}_2)\widehat{\boldsymbol{\beta}} - 2\widehat{\boldsymbol{\beta}}'(\mathbf{X}_1'\mathbf{y} + \mathbf{X}_2'\boldsymbol{A})\right]. \tag{5.5}$$

Here $\widehat{\boldsymbol{\beta}}$ and $\hat{\sigma}^2$ are MLEs of parameters $\boldsymbol{\beta}$ and σ^2, respectively. Using norms notation, the equation above (5) can be expressed as

$$\hat{\sigma}^2 = \frac{1}{\mathrm{n}}\left[\left\|\mathbf{y} - \mathbf{X}_1'\widehat{\boldsymbol{\beta}}\right\| + \left\|\mathbf{X}_2'\widehat{\boldsymbol{\beta}} - \boldsymbol{A}\right\|^2 + \boldsymbol{B} - \|\boldsymbol{A}\|^2\right].$$

Calculation of parameter estimates $\widehat{\boldsymbol{\beta}}^{(k+1)}$ and $\sigma^{(k+1)^2}$ in each (k + 1) step can be obtained as follows:

$$\widehat{\boldsymbol{\beta}}^{(k+1)} = (\mathbf{X}'\mathbf{X})^{-1}\left(\mathbf{X}_1'\mathbf{y} + \mathbf{X}_2'\mathbf{X}_2\widehat{\boldsymbol{\beta}}^{(k)} + \hat{\sigma}^{(k)}\mathbf{X}_2'\boldsymbol{A}^{(\mathrm{k}+1)}\right)$$

$$\sigma^{2^{(k+1)}} = \frac{1}{n}(\left\|\mathbf{y} - \mathbf{X}_1'\widehat{\boldsymbol{\beta}}^{(k)}\right\| + \sigma^{(k)}\left(\mathbf{z} - \mathbf{X}_2'\widehat{\boldsymbol{\beta}}\right)f(.) + m\sigma^{(k)^2}).$$

5.3. Variability Assessment

McLachlan and Peel (2000) defined an approach that can be employed for the variability assessment of all parameter estimates. The empirical observed information matrix serves as an estimate of the corresponding observed information matrix and is obtained by

$$I_e\left(\widehat{\boldsymbol{\psi}}\right) = \sum_{i=1}^{n} \nabla q_i\,(\widehat{\boldsymbol{\psi}})\nabla q_i(\widehat{\boldsymbol{\psi}})'\,),$$

where $\widehat{\boldsymbol{\psi}} = (\widehat{\boldsymbol{\beta}},\ \widehat{\sigma}^2)$ represents the vector of parameter estimates, or MLEs, and $\nabla \mathrm{q_i}\left(\widehat{\boldsymbol{\psi}}\right)$ is the gradient vector of the conditional expectation of the complete data log likelihood function constructed on the i^{th} observation and evaluated at $\widehat{\boldsymbol{\psi}}$. Note that: $\mathrm{Q}\left(\widehat{\boldsymbol{\psi}}\right) = \sum_{i=1}^{n} q_i\,(\widehat{\boldsymbol{\psi}})$**.** For each i, $\nabla q_i\left(\widehat{\boldsymbol{\psi}}\right)$ is a vector of length (p + 2) defined by

$$\nabla q_i\left(\widehat{\boldsymbol{\psi}}\right) = \left(\left(\frac{\partial q_i(\boldsymbol{\psi})}{\partial \boldsymbol{\beta}}\right)', \left(\frac{\partial q_i(\boldsymbol{\psi})}{\partial \sigma}\right)\right)'.$$

Consider a vector d = (1,..0,..1,..0) of length n, where a 1 represents a censored observation and a 0 represents an uncensored observation. It follows that

$$\frac{\partial q_i(\boldsymbol{\psi})}{\partial \boldsymbol{\beta}} = \frac{[\mathbf{x}_{1i}'y_i(1-d_i) - \mathbf{x}_{1i}'\mathbf{x}_{1i}(1-d_i)\boldsymbol{\beta} + \mathbf{x}_{2i}'d_iE(z_i) - \mathbf{x}_{2i}'\mathbf{x}_{2i}d_i\boldsymbol{\beta}]}{\sigma^2} \tag{5.6}$$

and

$$\frac{\partial q_i(\boldsymbol{\psi})}{\partial \sigma} = -\frac{1}{\sigma} + \frac{[y_i^2(1-d_i) - 2\boldsymbol{\beta}'\mathbf{x}_{1i}'y_i(1-d_i) + \boldsymbol{\beta}'\mathbf{x}_{1i}'\mathbf{x}_{1i}(1-d_i) + E(z_i^2)d_i - 2\boldsymbol{\beta}'\mathbf{x}_{2i}'E(z_i)d_i + \boldsymbol{\beta}'\mathbf{x}_{2i}'\mathbf{x}_{2i}\boldsymbol{\beta}d_i]}{\sigma^3}.$$

These partial derivatives will be used to assemble the covariance matrix. This covariance matrix of the MLEs, which is obtained by taking the inverse of $I_e(\widehat{\boldsymbol{\psi}})$ can be directly employed for testing various hypotheses and finding confidence intervals for the parameters of the model.

5.4. Model Selection

The Akaike Information Criterion (AIC) is a popular model selection procedure proposed by Akaike (1974). The AIC considers the negative log-likelihood plus a penalty term that reflects the number of free parameters (p) in the model. The form of the AIC is given by

$$AIC = -2l(\widehat{\boldsymbol{\psi}}) + 2p, \tag{5.7}$$

where $l(\widehat{\boldsymbol{\psi}})$ is defined as follows:

$$l(\widehat{\boldsymbol{\Psi}}) = -\frac{(n-m)}{2}\log(2\pi) - \frac{(n-m)}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}(\mathbf{y}-\mathbf{X}_1\boldsymbol{\beta})'(\mathbf{y}-\mathbf{X}_1\boldsymbol{\beta}) + \sum_{i=n-m+1}^{n} logP(y_i > z_i)$$

$$= -\frac{(n-m)}{2}\log(2\pi) - \frac{(n-m)}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}(\mathbf{y}-\mathbf{X}_1\boldsymbol{\beta})'(\mathbf{y}-\mathbf{X}_1\boldsymbol{\beta}) + \sum_{i=n-m+1}^{n} \log[\,1-\Phi(\frac{z_i-\mathbf{x}_{2i}\boldsymbol{\beta}}{\sigma})].$$

The model with the minimum AIC is selected as the best model to fit the data.

Another commonly used method in model selection was proposed by Schwarz (1978) and is known as Bayesian Information Criterion (BIC). Similar to AIC, the BIC approach adjusts the log-likelihood $l(\widehat{\boldsymbol{\psi}})$ by a penalty term which considers the number of observations (n) in the sample in addition to the number of parameters in the model:

$$BIC = -2l(\widehat{\boldsymbol{\psi}}) + p\log(n) \tag{5.8}$$

The model with the minimum BIC is chosen as the best model to fit the data.

CHAPTER 6. BUCKLY AND JAMES METHOD

6.1. Proposed Model

Buckley and James (1979) proposed a method that modifies the least squares normal equations in order to accommodate for right censoring. Consider the following model:

$$\boldsymbol{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \qquad \boldsymbol{\varepsilon} \sim \boldsymbol{F},$$

where $\boldsymbol{y}$ is an $n \times 1$ vector of right censored responses; $\mathbf{X}$ is a design matrix of size $n \times (p + 1)$ with p covariates; $\boldsymbol{\beta}$ is a $(p + 1) \times 1$ parameter vector estimated by

$\widehat{\boldsymbol{\beta}}^{\mathrm{T}} = (\hat{\beta}_0, \hat{\beta}_1, \ldots \ldots \ldots, \hat{\beta}_p)$; and $\boldsymbol{\varepsilon}$ is an $n \times 1$ error vector with independent and identically distributed realizations from an unspecified distribution $\boldsymbol{F}$ having mean zero and finite variance.

6.2. Parameter Estimates

The Buckley-James method replaces the censored responses with their estimated expected conditional values, called renovated values, $\mathbf{y}^*(\widehat{\boldsymbol{\beta}})$, using the following equation:

$$\mathbf{y}^*(\widehat{\boldsymbol{\beta}}) = \mathbf{X}\widehat{\boldsymbol{\beta}} + \mathrm{W}(\widehat{\boldsymbol{\beta}})(\boldsymbol{z} - \mathbf{X}\widehat{\boldsymbol{\beta}}),$$

where $\boldsymbol{z} - \mathbf{X}\widehat{\boldsymbol{\beta}} = e(\widehat{\boldsymbol{\beta}})$ is a vector of observed residuals $e(\widehat{\boldsymbol{\beta}}) = (e_1(\hat{\beta}), e_2(\hat{\beta}), \ldots e_n(\hat{\beta}))^T$. The matrix of weights is the upper triangular Renovation Weight Matrix defined as follows:

$$\mathrm{W}(\widehat{\boldsymbol{\beta}}) = \mathrm{diag}(\boldsymbol{\delta}) + \{\mathrm{w}_{ik}(\widehat{\boldsymbol{\beta}})\}$$

$$= \begin{bmatrix} \delta_1 & w_{12}(\widehat{\boldsymbol{\beta}}) & w_{13}(\widehat{\boldsymbol{\beta}}) & \ldots & & w_{14}(\widehat{\boldsymbol{\beta}}) \\ 0 & \delta_2 & w_{23}(\widehat{\boldsymbol{\beta}}) & \ldots & & w_{2n}(\widehat{\boldsymbol{\beta}}) \\ \vdots & \vdots & \ddots & \ddots & & \vdots \\ 0 & 0 & 0 & \ldots & & w_{(n-1)n}(\widehat{\boldsymbol{\beta}}) \\ 0 & 0 & 0 & \ldots & \ldots & \delta_n \end{bmatrix},$$

where $\delta_i = 1$ if the observation is uncensored and $\delta_i = 0$ if the observation is censored, and the weights $w_{ik}(\widehat{\boldsymbol{\beta}})$ are defined as:

$$w_{ik}(\widehat{\boldsymbol{\beta}}) = \begin{cases} \frac{d\hat{F}(e_k(\widehat{\boldsymbol{\beta}})\delta_k(1-\delta_i)}{\hat{s}(e_i(\widehat{\boldsymbol{\beta}}))} & \text{if } e_k(\widehat{\boldsymbol{\beta}}) > e_i(\widehat{\boldsymbol{\beta}}), \\ 0 & \text{otherwise} \end{cases}$$

with $d\hat{F}$ being a probability mass assigned to the uncensored residuals using the Kaplan-Mayer product limit estimator :

$$\hat{F} = 1 - \prod_{i;e(i)\leq \varepsilon} \left(\frac{n-i}{n-i+1}\right)^{\delta_i}.$$

The process of finding the parameter estimates resembles an EM algorithm. The initial value of $\widehat{\boldsymbol{\beta}}^{(0)}$ is proposed. Then a new estimated value on the $(m+1)$ iteration is calculated as

$$\widehat{\boldsymbol{\beta}}^{(m+1)} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}^*(\widehat{\boldsymbol{\beta}}^{(m)}). \tag{6.1}$$

6.3. Variability Assessment

Variance estimation by the Buckley-James method has been studied by many researchers considering that the original paper by Buckley and James proposed a heuristic variance estimator based on uncensored observations only. Buckley and James substantiated the adequacy of the variance estimator through simulation testing. They showed that even in cases when censoring is not uniformly distributed along the line, the variance calculation is adequate. For multiple linear regression the covariance matrix is estimated as:

$$\Sigma = \hat{\sigma}^2(\mathbf{X}^{\mathrm{T}}\Delta\,\mathbf{X})^{-1}, \tag{6.2}$$

where

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^{n} \delta_i [e_i(\hat{\boldsymbol{\beta}}) - \frac{1}{n_u} \sum_{j=1}^{n} (\delta_j e_j(\hat{\boldsymbol{\beta}})]^2}{n_u - p},$$

n_u is the number of uncensored observations, $\Delta = diag(\boldsymbol{\delta})$, and p is the length of the parameter vector $\boldsymbol{\beta}$ in the regression model. The implementation of this method is currently available in the R library (rms).

6.4. Model Selection

A measure of explained variation was proposed by Hocking (2003) based on the square of the Pearson correlation coefficient between the response and predicted response. Using a similar path, Glasson (2007) proposed a measure of explained variation in the BJ model, similar to the Pearson correlation coefficient, using the uncensored observations only. Let s_{Y_u} and $s_{\hat{Y}_u}$ represent the sample standard deviation for the actual Y_u and the predicted response $\hat{Y}_u$ based on the uncensored data only. The measure of explained variation for the uncensored data based on the number of uncensored observations n_u is calculated as

$$r_u^2 = [r(Y_u, \hat{Y}_u)]^2,$$

where

$$r_u = \frac{\sum_{i=1}^{n} \delta_i (Y_i - \bar{Y}_i)(\hat{Y}_i - \bar{\hat{Y}}_i)}{(n_u - 1) s_{Y_u} s_{\hat{Y}_u}}. \quad (6.3)$$

It follows that $0 \leq r_u^2 \leq 1$. The biggest disadvantage of r_u^2 is that it does not take into account censored data. However, Glasson (2007) pointed out that r_u^2 in practice produces the most realistic results for the BJ model and is an adequate measure for assessing its predictive power.

CHAPTER 7. VALIDATION AND DIAGNOSTIC TOOLS FOR THE EM ALGORITHM

7.1. Introduction

Model validation and diagnostics are well developed procedures in ordinary regression theory for checking correct specifications of the model. In this chapter, we propose formulas for computing the coefficient of determination, outlier detection, and influence diagnostics in right censored regression based on the EM algorithm. Availability of these tools should promote use of the EM algorithm in modeling right censored regression. Users of the EM model will be equipped to measure the utility of the model and detect any outliers that may lead to possible model misspecifications. Also, influence diagnostics are helpful in detecting influential observations that may allow users to better understand the nature of the data.

7.2. Reconstructed Coefficient of Determination

It is a standard approach for modeling multiple regression to consider the coefficient of determination (R^2) as a useful measure of how well the model fits the data. The R^2 is defined as the proportion of total response variation that is explained by the model. The R^2 is also used as a tool in model selection, with higher R^2 indicating better model fit. However, R^2 alone does not indicate whether the model is appropriate.

The R^2 for ordinary regression is defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} = 1 - \frac{SSE}{TSS},$$

where SSE represents the sum of squares for error and TSS is the total sum of squares. The TSS measures the variability in the model relative to the horizontal line $\bar{y}$. The SSE measures the

variability in the response **y** from the fitted line $\hat{\boldsymbol{y}}$. For ordinary regression, the best fitted model is defined based on the principle of least squares which minimizes the sum of squares of errors (SSE).

For right-censored regression using the EM algorithm, there is no a comparable measure developed by researchers. The least squares method cannot be applied due to the presence of censored data. The following proposed R^2 calculation is based on the idea of maximizing the Q-function given optimal values of the parameters relative to the maximization of the same function assuming the intercept term only.

Assume *p* is the number of independent variables in the model. Define the following objective function as

$$J(\beta_0, \beta_1, \ldots., \beta_p) = \|\boldsymbol{y} - \mathbf{X}_1\boldsymbol{\beta}\|^2 + \|\boldsymbol{A} - \mathbf{X}_2\boldsymbol{\beta}\|^2 + \boldsymbol{B} - \|\boldsymbol{A}\|^2.$$

Next, define

$$J_{lin}(p) = \min_{\beta_0,\beta_1,\ldots\ldots\beta_p} J(\beta_0, \beta_1 \ldots, \beta_p),$$

$$J_{const} = \min_{\beta_0} J(\beta_0, 0 \ldots, 0) = J_{lin}(0).$$

J_{lin} is the optimal value of the objective function J if we use the whole design partition matrix **X**. J_{const} is the optimal value of the same function if we use only the first column of **X.**

The proposed R-squared is defined as the *reconstructed coefficient of determination:*

$$R^2_{EM}(p) = 1 - \frac{J_{lin}(p)}{J_{const}}. \qquad (7.1)$$

The $R^2_{EM}(p)$ does not have a closed form solution compared to an ordinary coefficient of determination.

THEOREM: The following statements about $R^2_{EM}(p)$ are true:

(i) $0 \leq R^2_{EM}(p) \leq 1$

(ii) Function $R^2_{EM}(p)$ is non-decreasing with respect to p.

PROOF:

(i) By definition of J_{lin} and J_{const}, we have $0 \leq J_{lin} \leq J_{const}$.

Therefore $0 \leq R^2_{EM}(p) \leq 1$

(ii) By definition, $J_{lin}(p)$ is non-increasing in p. Therefore $R^2_{EM}(p)$ is non- decreasing with respect to p.

7.3. Reconstructed Jackknife Residuals and Outliers

A Jackknife residual is commonly used in regression diagnostics to denote a difference between the actual response y_i and the predicted response $\hat{y}_{(i)}$ for observation *i* when y_i is deleted from the analysis. That is,

$$\hat{d}_i = y_i - \hat{y}_{(i)}.$$

For right censored regression based on the EM algorithm, the *reconstructed Jackknife residual* is proposed to accommodate censored observations and is defined as

$$\hat{d}_i^* = y_i - \hat{y}_{(i)}^*,$$

where $\hat{y}^*_{(i)} = \hat{y}_{(i)}$ if the *i*th observation is uncensored and $\hat{y}^*_{(i)} = \mathbf{x_2^T}\,\widehat{\boldsymbol{\beta}}_{(i)}$ if the *i*th observation is censored. The parameter estimate $\widehat{\boldsymbol{\beta}}_{(i)}$ is obtained when the EM algorithm is implemented on the data with the *i*th observation removed. The actual value of y_i is known for deleted uncensored observations. However, for the *i*th censored observation the actual value is unknown and can be estimated by $y_i = \boldsymbol{A}_i$ using equation (5.2).

There are many procedures used to identify outliers. One procedure involves simply comparing the residuals for all observations. If one residual is much larger in absolute value than the others, then the observation corresponding to this residual is declared an outlier. Another procedure for identifying outliers involves comparing a residual to a critical value based on a probability distribution. Both of these procedures are discussed below.

The variance of a standard residual $\hat{\varepsilon}_i = y_i - \hat{y}_i$ is not constant. It is estimated by

$$var(\varepsilon_i) = \sigma^2(1 - h_{ii}),$$

where h_{ii} is *i*th diagonal element of the matrix $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. Scaling the residual by its variance and replacing σ by the sample standard deviation *s*, we obtain the standardized residual $r_i = \frac{\hat{\varepsilon}_i}{s\sqrt{1-h_{ii}}} \sim N(0,1)$. The residual r_i is known as *studentized residual*.

The method of scaling Jackknife residuals is based on statistic $t_i = \frac{\hat{d}_{(i)}}{\sqrt{var(\hat{d}_{(i)})}}$ where $t_i \sim t_{\alpha/2n}(n-p-1)$. Simple computations show that $var(\hat{d}_{(i)}) = \sigma^2/(1 - h_{ii})$. If an outlier comes from a distribution with a different mean, then the model can be expressed as $E(y_i) = \boldsymbol{x_i^T}\boldsymbol{\beta} + \theta$. The test for an outlier is the same as the test of the hypothesis H_0: $\theta = 0$.

If $|t_i| > t_{\alpha/2n}(n-p-1)$, then H_0 is rejected and the *i*th observation is declared an outlier. Since a test needs to be performed on all observations, the Bonferroni adjustment to the critical value $\frac{\alpha}{2n}$ can be used or observations with relatively large t_i compared to the other observations should be flagged.

A similar idea can be used for Jackknife residuals in right censored regression. Let $s^*_{(i)}$ denote the standard deviation of the model when the *i*th observation is deleted. If the deleted observation is uncensored, then $s^*_{(i)}$ coincides with the same estimate obtained from the EM algorithm. For the censored case, we propose that an approximation is used where $s^*_{(i)}$ is the standard deviation based on all uncensored observations and the *i*th censored observation.

For uncensored observations, we calculate a matrix similar to **H** as $\mathbf{H^u} = \mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'$, where the *i*th diagonal element of this matrix is denoted as h_{ii}^u, $i = 1, 2, \ldots, (n-m)$. In the case of censored observations, we propose to use $\mathbf{H^c} = \mathbf{C}(\mathbf{C}'\mathbf{C})^{-1}\mathbf{C}'$, where $\mathbf{C}$ has matrix dimension $(n-m+1)\times(p+1)$ that includes the partition matrix $\mathbf{X}_1$ with the *i*th row added from the partition matrix $\mathbf{X}_2$. Let the *i*th diagonal element of $\mathbf{C}$ be denoted by c_{ii}. A test statistic for detecting outliers based on the reconstructed Jackknife residuals is proposed as

$$t_i^* = \frac{\hat{d}^*_{(i)}}{\sqrt{var(\hat{d}^*_{(i)})}}\ , \qquad (7.2)$$

where $var(\hat{d}^*_{(i)}) = {s^*_{(i)}}^2/(1-h_{ii}^u)$ applies when the *i*th observation is uncensored and ${var(\hat{d}^*_{(i)}) = s^*_{(i)}}^2/(1-c_{ii})$ applies when *i*th observation is censored. Similar to regression with no censoring, we test the hypothesis $H_0: \theta = 0$. If $|t_i^*| > t_{\alpha/2n}(n-p-1)$, H_0 is rejected the

observation is declared an outlier. Since *n* tests should be performed, a Bonferroni adjustment to the critical value of the t-distribution is appropriate to use, or the observations with relatively large values of t_i^* should be flagged as outliers.

7.4. Influence Diagnostics: One-Step Deletion Method

For assessing the influence of a single observation on the parameter estimates in censored regression, the most popular methods include the empirical influence curve and one-step deletion method. They are described briefly as follows.

Assume $\Lambda(\boldsymbol{\theta}, \mathbf{y})$ is a likelihood function depending on parameters $\boldsymbol{\theta}$ and observations $\mathbf{y}$. Assume $F(\mathbf{y})$ is the c.d.f. of random variable Y. The "average likelihood function" is defined as

$$J(\boldsymbol{\theta}) = \int \Lambda(\boldsymbol{\theta}, \mathbf{y}) dF(\mathbf{y}) .$$

The point $\widehat{\boldsymbol{\theta}}$ that maximizes J is considered the best approximation of parameters $\boldsymbol{\theta}$. Assuming all functions are sufficiently smooth, we have $\frac{dJ(\widehat{\boldsymbol{\theta}})}{d(\boldsymbol{\theta})} = 0$, or

$$\int \frac{\partial \Lambda(\widehat{\boldsymbol{\theta}}, \mathbf{y})}{\partial \boldsymbol{\theta}} dF(\mathbf{y}) = 0 .$$

Now consider the dependence of $\widehat{\boldsymbol{\theta}}$ on F. Denote by $\Delta_{\bar{y}}$ the step function at $\bar{y}$, and define

$F_{\epsilon,\bar{y}}(\mathbf{y}) = (1 - \epsilon)F(\mathbf{y}) + \epsilon\Delta_{\bar{y}}$. Also denote $\widehat{\boldsymbol{\theta}}_{\epsilon,\bar{y}}$ as the point that maximizes function

$$J_{\epsilon,\bar{y}}(\boldsymbol{\theta}) = \int \Lambda(\boldsymbol{\theta}, \mathbf{y}) dF_{\epsilon,\bar{y}}(\mathbf{y}).$$

Then, $lim_{\epsilon\to 0}\frac{\widehat{\boldsymbol{\theta}}_{\epsilon,\bar{y}}-\overline{\boldsymbol{\theta}}}{\epsilon}=\left.\frac{d\widehat{\boldsymbol{\theta}}_{\epsilon,\bar{y}}}{d\epsilon}\right|_{\epsilon=0}$ may be considered as a sensitivity coefficient of the optimal likelihood estimator $\overline{\boldsymbol{\theta}}$ with respect to an observation at point $\hat{\boldsymbol{y}}$. Denote

$q(\boldsymbol{\theta},\mathbf{y})=\frac{\partial\Lambda(\boldsymbol{\theta},\mathbf{y})}{\partial\boldsymbol{\theta}}$, and $I(\boldsymbol{\theta})=-\int\frac{\partial^2\Lambda(\boldsymbol{\theta},\mathbf{y})}{\partial\boldsymbol{\theta}^2}\,dF(\mathbf{y})$. Then

$$\left.\frac{d\widehat{\boldsymbol{\theta}}_{\epsilon,\bar{y}}}{d\epsilon}\right|_{\epsilon=0}=I\left(\widehat{\boldsymbol{\theta}}\right)^{-1}q\left(\widehat{\boldsymbol{\theta}},\bar{y}\right).$$

The one-step deletion method measures the change in parameter estimates when the *i*th data point is deleted from the sample. For the EM Algorithm, the formula produced by Weissfeld and Schneider (1990) is

$$\Delta EM=\widehat{\boldsymbol{\theta}}-\widehat{\boldsymbol{\theta}}_{(i)}=$$

$$=\frac{(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{x}_i^{\mathrm{T}}\left[y_i\delta_i-\mathbf{x}_i^{\mathrm{T}}\widehat{\boldsymbol{\beta}}+(1-\delta_i)\left(\mathbf{x}_i^{\mathrm{T}}\widehat{\boldsymbol{\beta}}+\hat{\sigma}h(\hat{u}_i)\right)\right]}{1-\mathbf{x}_i^{\mathrm{T}}(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{x}_i},\qquad(7.3)$$

where $\hat{u}_i=(y_i-\mathbf{x}_i^{\mathrm{T}}\widehat{\boldsymbol{\beta}})/\hat{\sigma}$ and $h(\hat{y}_i)=\phi(\hat{u}_i)/(1-\phi(\hat{u}_i))$.

This formula is not consistent with formula (7.6), which we derive as follows. The optimal values of the parameters in β are given by the following well known formula:

$$\widehat{\boldsymbol{\beta}}=(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}^{*},$$

where $\mathbf{y}^{*}=\binom{\mathbf{y}}{\boldsymbol{A}}$ is a vector of uncensored and reconstructed censored observations. Now, assume that the *i*th observation is omitted. Then, instead of using matrix **X** we have to use matrix $\mathbf{X}_{(i)}$ which is the matrix **X** with the *i*th row omitted. For this problem, the optimal model has optimal parameters $\boldsymbol{\beta}_{(i)}$ which can be found using a similar formula:

$$\widehat{\boldsymbol{\beta}}_{(i)} = \left(\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)}\right)^{-1}\mathbf{X}_{(i)}\tilde{\mathbf{y}}_{(i)}^{*}.$$

Here, $\tilde{\mathbf{y}}_{(i)}^{*}$ is the vector of uncensored and reconstructed censored observations based on all available observations except for *i*th observation. Denote the *i*th component of the vector of observations, y_i^*, based on all available observations (including the *i*th). Denote by $\mathbf{x}_i$ the *i*th row of the matrix $\mathbf{X}$ which is omitted in $\mathbf{X}_{(i)}$. Then, $\mathbf{X}^{\mathrm{T}}\mathbf{X} = \mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)} + \mathbf{x}_i^{\mathrm{T}}\mathbf{x}_i$ and

$\mathbf{X}^{\mathrm{T}}\mathbf{y}^{*} = \mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{y}_{(i)}^{*} + \mathbf{x}_i^{\mathrm{T}}\mathrm{y}_i^{*}$. Thus,

$$(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\left(\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)} + \mathbf{x}_i^{\mathrm{T}}\mathbf{x}_i\right) = \mathbf{I}\,.$$

Multiplying this equation by $(\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)})^{-1}$ we obtain:

$$(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\left(\mathbf{I} + \mathbf{x}_i^{\mathrm{T}}\mathbf{x}_i(\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)})^{-1}\right) = (\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)})^{-1}. \qquad (7.4)$$

Next, if we multiply each side of equation (7.4) by $\mathbf{x}_i$ from the left and regroup the terms, we have

$$\mathbf{x}_i\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1} + \mathbf{x}_i\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_i(\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)})^{-1} = \mathbf{x}_i(\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)})^{-1}$$

$$\mathbf{x}_i\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1} = \left(1 - \mathbf{x}_i\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{x}_i^{\mathrm{T}}\right)\mathbf{x}_i(\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)})^{-1}$$

$$\mathbf{x}_i(\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)})^{-1} = \left(1 - \mathbf{x}_i\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{x}_i^{\mathrm{T}}\right)^{-1}\mathbf{x}_i\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}. \qquad (7.5)$$

Substituting (7.5) into the appropriate part of (7.4) we get

$$(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\left(\mathbf{I} + \frac{\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_i(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}}{(\mathbf{1} - \mathbf{x}_i(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{x}_i^{\mathrm{T}})}\right) = (\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)})^{-1}.$$

Then, we have

$$\Delta\boldsymbol{\beta}^{EM} = \widehat{\boldsymbol{\beta}} - \widehat{\boldsymbol{\beta}}_{(i)} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}^{*} - \left(\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{X}_{(i)}\right)^{-1}\mathbf{X}^{\mathrm{T}}{}_{(i)}\tilde{\mathbf{y}}_{(i)}^{*} =$$

$$= (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\left[\mathbf{X}_{(i)}^{\mathrm{T}}\mathbf{y}_{(i)}^{*} + \mathbf{x}_i^{\mathrm{T}}\mathrm{y}_i^{*} - \left(\mathbf{I} + \frac{\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_i(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}}{\mathbf{1} - \mathbf{x}_i(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{x}_i^{\mathrm{T}}}\right)\mathbf{X}_{(i)}^{\mathrm{T}}\tilde{\mathbf{y}}_{(i)}^{*}\right] =$$

$$= \frac{(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}}{\mathbf{1} - \mathbf{x}_i(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{x}_i^{\mathrm{T}}}\left[\left(\mathbf{1} - \mathbf{x}_i(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{x}_i^{\mathrm{T}}\right)\mathbf{X}_{(i)}^{\mathrm{T}}(\mathbf{y}_{(i)}^{*} - \tilde{\mathbf{y}}_{(i)}^{*})\right.$$

$$\left. + \mathbf{x}_i^{\mathrm{T}}\mathbf{y}_{(i)}^{*}\left(\mathbf{1} - \mathbf{x}_i(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{x}_i^{\mathrm{T}}\right) - \mathbf{x}_i^{\mathrm{T}}\mathbf{x}_i(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}_{(i)}^{\mathrm{T}}\tilde{\mathbf{y}}_{(i)}^{*}\right].$$

Finally, we obtain $\Delta\boldsymbol{\beta}^{\mathrm{EM}}$ as

$$\Delta\boldsymbol{\beta}^{\mathrm{EM}} = \frac{(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}}{\mathbf{1} - \mathbf{x}_i(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{x}_i^{\mathrm{T}}}\{\left[\left(1 - \mathbf{x}_i(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{x}_i^{\mathrm{T}}\right)\mathbf{I} + \mathbf{x}_i^{\mathrm{T}}\mathbf{x}_i(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\right]\mathbf{X}_{(i)}^{\mathrm{T}}\left(\mathbf{y}_{(i)}^{*} - \tilde{\mathbf{y}}_{(i)}^{*}\right) +$$

$$+ \mathbf{x}_i^{\mathrm{T}}(y_i^{*} - \mathbf{x}_i\widehat{\boldsymbol{\beta}})\}. \qquad (7.6)$$

By comparing the new formula (7.6) to (7.3), we observe that they are different and coincide if $\mathbf{y}_{(i)}^{*} - \tilde{\mathbf{y}}_{(i)}^{*} = 0$. However, if the difference $\mathbf{y}_{(i)}^{*} - \tilde{\mathbf{y}}_{(i)}^{*}$ is not equal to zero, then in general, the formulas produce different results.

In order to eliminate the influence of an observation due to its position on the interval of *x* values, the vector $\Delta\boldsymbol{\beta}^{\mathrm{EM}}$ in the formula (7.6) can be divided by the vector $(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{x}_i$ component-vise providing a valuable measure of the change in the coefficients of the linear model. Thus the normalized version of the formula (7.6) is defined as

$$\left[\Delta\boldsymbol{\beta}_{\mathrm{NOR}}^{\mathrm{EM}}\right]_j = \frac{\left[\Delta\boldsymbol{\beta}^{\mathrm{EM}}\right]_{\mathrm{j}}}{\left[(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{x}_i\right]_j} \qquad j = 1,2\ldots\mathrm{p}. \qquad (7.7)$$

Relatively large values of this statistic indicate the most influential observations on the coefficient estimates of the model. It is important to note that Weissfeld and Schneider (1990) did not address this issue when calculating their formula for the one-step deletion based on the EM algorithm.

7.5. Examples

An example is presented in this section to illustrate how the one-step deletion method performs under different settings. Two scenarios are considered:

Scenario 1: the influential observations are closer to the end of the interval, and

Scenario 2: the influential observations are in the middle of the interval.

A fire insurance data set published by Mendenhall and Sincich (2012) was modeled using simple linear regression. All 15 observations reported in Table 7.1 were considered uncensored. Distance from the fire station (in miles) represents the independent variable (x) while Fire Damage in thousands of dollars represents the dependent variable (y).

Scatterplots of data presented in Table 7.1 are shown in Figure 7.1. While uncensored observations for Scenario 1 and Scenario 2 are represented by circles, the censored observations are represented by solid filled black circles. Influential observations stand out compared to all other observations and are easily recognizable on scatterplots for Scenario 1 and Scenario 2.

In Scenario 1, we assume that five randomly selected observations (numbered: 2, 4, 9, 13, and 15) were censored, with observation number 13 being censored at a very high level (e.g. 100) and located toward the upper end of the interval. Table 7.1 shows these censored observations in bold.

Table 7.1. Fire Insurance Data Used for Influence Diagnostics

No.	x	y	Scenario 1	Scenario 2
1	0.7	14.1	14.1	14.1
2	**1.1**	**17.3**	**17.3**	**17.3**
3	1.8	17.8	100	17.8
4	**2.1**	**24.0**	**24.0**	**24.0**
5	2.3	23.1	23.1	23.1
6	2.6	19.6	19.6	100
7	3.0	22.3	22.3	22.3
8	3.1	27.5	27.5	27.5
9	**3.4**	**26.1**	**26.1**	**100**
10	3.8	26.2	26.2	26.2
11	4.3	31.3	31.3	31.3
12	4.6	31.3	31.3	31.3
13	**4.8**	**36.4**	**100**	**36.4**
14	5.5	36.0	36.0	36.0
15	**6.1**	**43.2**	**43.2**	**43.2**

Observation number 3 was uncensored at a very high level (e.g. 100) and located close to the lower end of the interval. Thus, two highly influential observations were created in the data and both of them were close to the end of the interval.

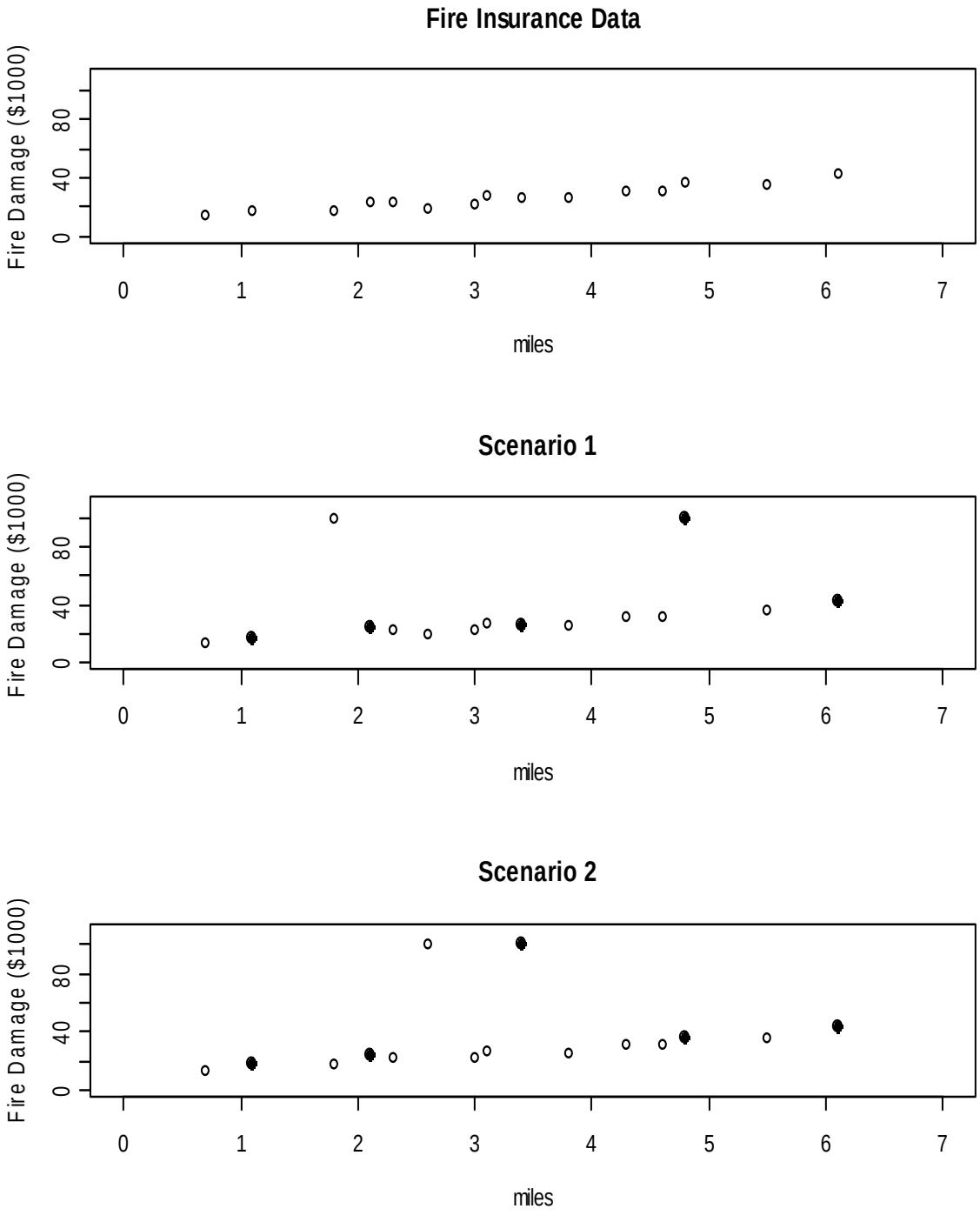

Figure 7.1. Scatterplots of Data Presented in Table 7.1.

If the influence on the parameter estimates is calculated without normalization for points close to the end of the interval, the results will always recognize these as influential points in the data. Influence on the slope is greater than the influence on the intercept for this case. Figure 7.2 shows the results of the one-step deletion method using formula (7.6) for Case 1. In figure 7.3, formula (7.7) is used to evaluate influence, and the results are much improved in a sense that influential points are more easily recognizable for the slope coefficients. We can observe that somewhat more uniform weights are given to other influential observations using formula (7.7) compared to the results of formula (7.6). Formula (7.6) gives very little weight to the influence of the observations in the middle of the interval. While in Case 1, we can still recognize the influential points for the slope coefficient even when we use formula (7.6), they are much more distinct and easily recognizable when formula (7.7) is used instead.

The influence for each observation before and after normalization for Case 1 is summarized in Table 7.2. As we expected, it can be easily observed that observations number 3 and 13 are the most influential. These observations were assigned very high values. These results confirm the validity of the new method proposed for identifying influential points using one-step deletion.

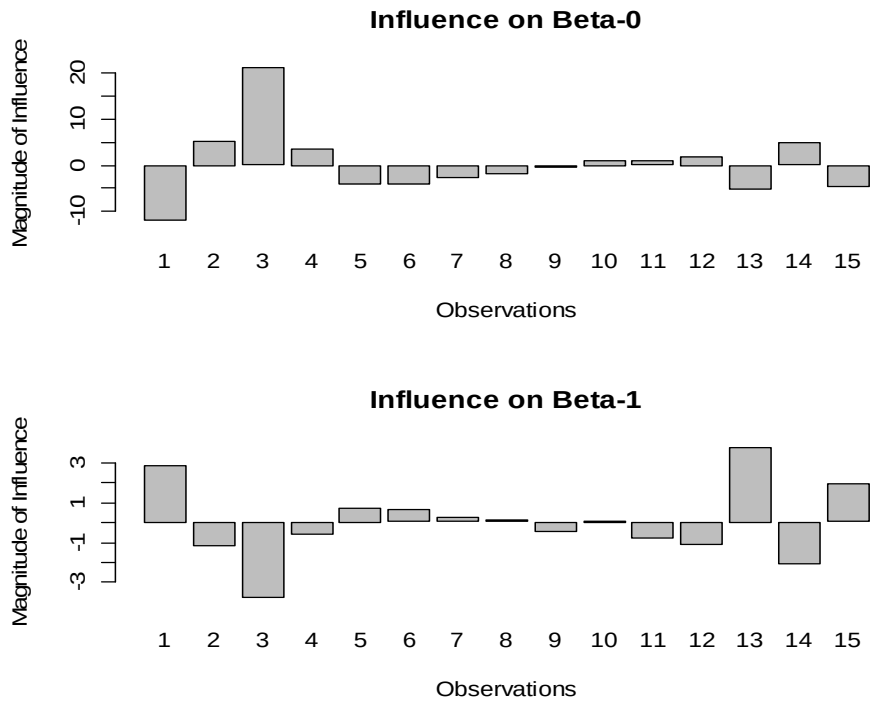

Figure 7.2. Influence Diagnostics for Scenario 1 Before Normalization

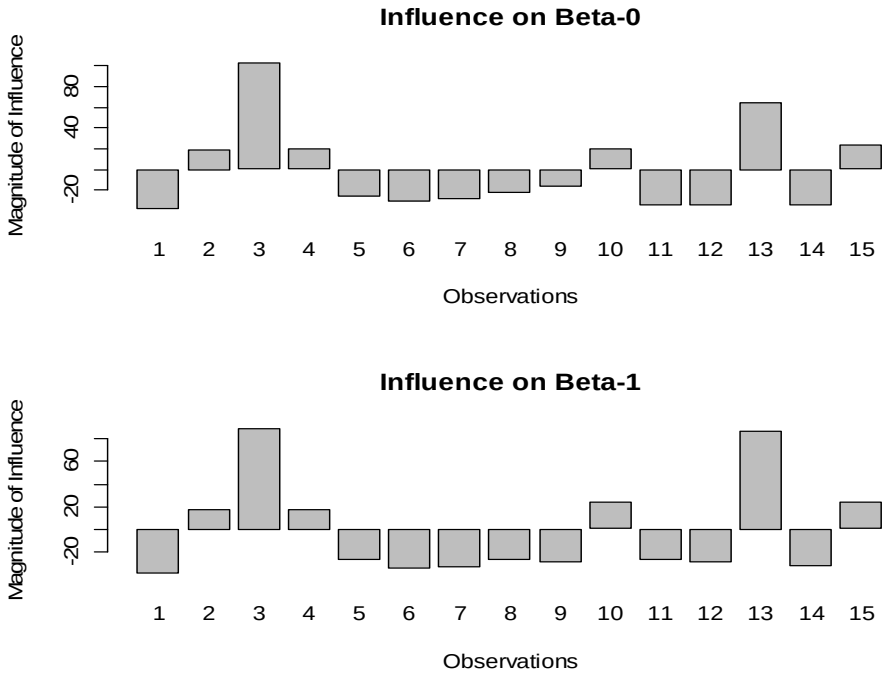

Figure 7.3. Influence Diagnostics for Scenario 1 After Normalization

Table 7.2. Influence Diagnostics Results for Scenario 1

Scenario 1 (Influential Points are Close to the end of the Interval)		
No.	**Influence Before Normalization**	**Influence After Normalization**
1	(-11.6529, 2.8758)	(-37.5961, -38.7724)
2	(5.2114, -1.1379)	(19.1433, 18.1577)
3	(21.1148, -3.7902)	(102.3876, 89.0805)
4	(3.5491, -0.6169)	(19.9461, 18.1852)
5	(-3.9668, 0.7501)	(-24.9365, -26.6245)
6	(-3.9455, 0.6542)	(-30.1672, -33.4652)
7	(-2.5754, 0.2653)	(-27.6726, -32.9662)
8	(-1.8204, 0.1350)	(-21.7653, -26.0957)
9	(-0.2829, -0.4275)	(-16.0450, -28.6000)
10	(1.0801, 0.0827)	(19.5145, 23.9726)
11	(0.9784, -0.7654)	(-33.1519, -26.1035)
12	(1.9150, -1.0945)	(-33.1302, -28.8424)
13	(-4.9586, 3.8028)	(64.6800, 87.0243)
14	(4.8640, -2.0568)	(-34.0926, -32.2269)
15	(-4.6082, 1.9613)	(23.1282, 24.1923)

In Case 2, two influential observations were selected in the middle of the interval. These are uncensored observation number 6 and censored observation number 9. Their values are set at 100. Figures 7.4 and 7.5 show the influence diagnostics for each observation on both coefficients of the model before and after normalization.

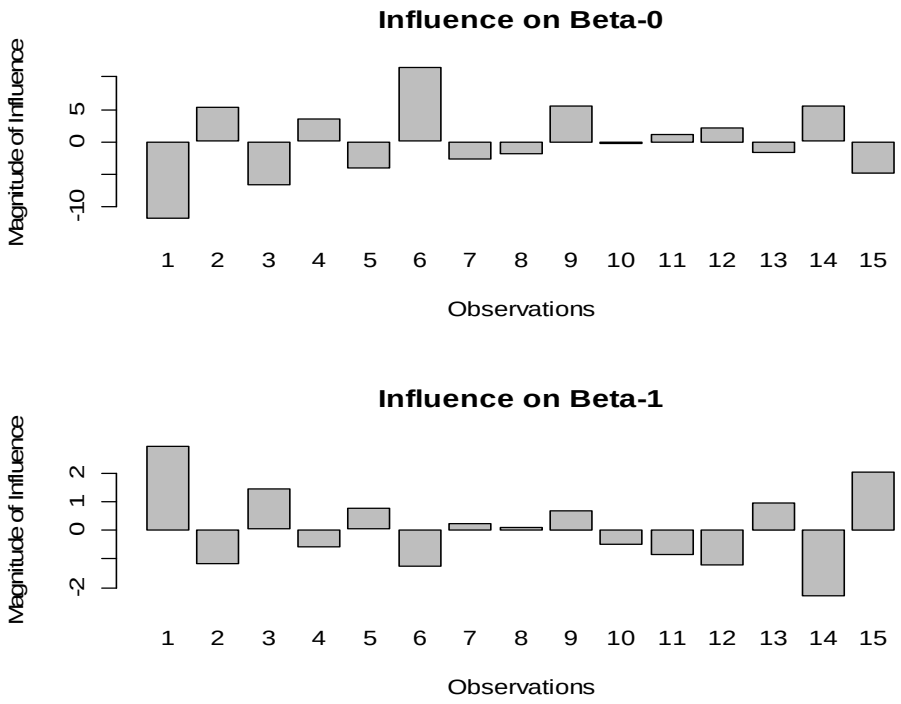

Figure 7.4. Influence Diagnostics for Scenario 2 Before Normalization

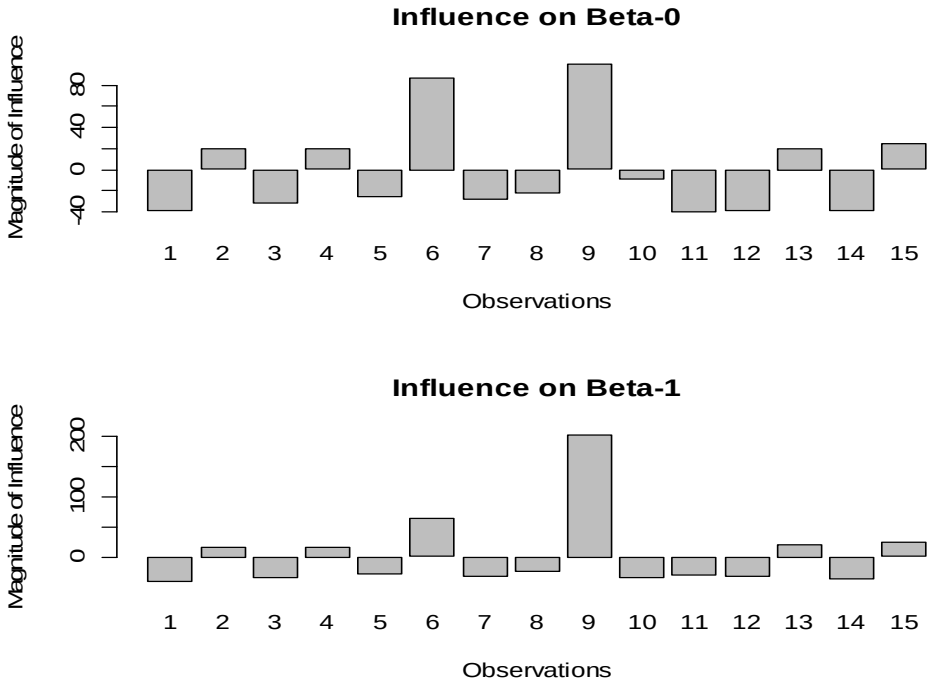

Figure 7.5. Influence Diagnostics for Scenario 2 After Normalization

Table 7.3. Influence diagnostics results for Scenario 2

Scenario 2 (Influential Points are in the Middle of the Interval)		
No.	**Influence Before Normalization**	**Influence After Normalization**
1	(-11.7775, 2.9510)	(-37.9981, -39.7872)
2	(5.2472, -1.1588)	(19.2750, 18.4909)
3	(-6.4686, 1.4292)	(-31.3668, -33.5909)
4	(3.4941, -0.6053)	(19.6373, 17.8458)
5	(-3.9956, 0.7628)	(-25.1178, -27.0756)
6	(11.4242, -1.2712)	(87.3493, 65.0263)
7	(-2.5405, 0.2497)	(-27.2977, -31.0275)
8	(-1.7918, 0.1180)	(-21.4235, -22.8198)
9	(5.5175, 0.6999)	(99.6834, 202.8967)
10	(-0.1501, -0.4876)	(-8.5167, -32.6227)
11	(1.1813, -0.8564)	(-40.0262, -29.2067)
12	(2.1993, -1.2184)	(-38.0483, -32.1057)
13	(-1.5339, 0.9642)	(20.0083, 22.0661)
14	(5.4936, -2.3171)	(-38.5054, -36.3058)
15	(-4.8448, 2.0557)	(24.3154, 25.3571)

When the influential points are located in the middle of the interval they may not be detected using formula (7.6). Figure 7.4 shows that influence is still driven by points located toward the end of the interval. Influence of the points in the middle of the interval is significantly worse for the slope than for the intercept.

When formula (7.7) is applied in Case 2, the results are much improved. From figure 7.5, we can easily observe observations number 6 and 9 as the most influential. Table 7.3 summarizes the magnitude of influence by observation for both coefficients in the model. It can also be observed that censored points in the middle of the interval have a greater impact on the slope than on the intercept of the model compared to those uncensored observations in the middle of the interval. In conclusion, formula (7.7) provides the most useful way of identifying influential points based on the EM algorithm with the one-step deletion method.

CHAPTER 8. SIMULATION STUDIES

8.1. Introduction

In this chapter a number of simulation studies were performed in order to assess the performance of right censored regression using the EM algorithm with application in actuarial science and insurance. Insurance data typically carry incomplete information. For example, if property losses are capped by their respective policy limits, then the sample data is right censored. Auto policies include liability limits, which is the maximum amount paid by the insurance company in case of a liability loss. Some health insurance policies are capped based on the type of coverage provided by the insurance company. Limits on insurance policies can vary from company to company, by line of business, type of policy, characteristics of the policyholder, etc.

The most important objective in the pricing of insurance products is to find the best model to fit the loss distribution in the presence of rating variables. For example, fire property losses are related to several rating variables such as building code, type of siding, location of the property, distance from the closest fire station, etc. Most of the methodologies for fitting the loss distribution in non-life insurance are based on grouped data which requires grouping losses by loss size. This approach is different than the proposed method. Ideally, an insurance company would like to determine prices on an individual basis using losses that are not grouped. In this case a right censored regression model can be used to link losses to a set of rating variables used in pricing. The objective of the following simulation studies is to help a pricing actuary evaluate the performance of the EM algorithm under different assumptions.

8.2. Performance of the EM Algorithm

The first simulation study was carried out to assess the performance of the EM algorithm for different amounts of censoring and different error assumptions. Since the EM algorithm is based on the assumption of normally distributed errors, the results are compared to those generated by the BJ algorithm which does not specify the type of error distribution to be used.

Data were simulated from the linear regression model:

$$y = \beta_0 + \beta_1 x + \varepsilon,$$

with $\beta_0 = 1$ and $\beta_1 = 2$.

The independent variable was designed such that $x_i = \frac{i}{n}$, $i = 1, 2, \ldots . n,$ where n is the sample size. Different simulation settings were created by manipulating the sample size ($n = 150, 60$), the percentage of points censored (10%, 30%, 50%), and the random distribution of the error terms (N(0, 0.182), U(-0.5,0.5), Exp(1)-1). Once the data were generated, censored points were selected at random on the entire interval. Their values were compared to a censoring level randomly drawn from U (1, 3). If a value of a selected data point was above the censoring level, it was trimmed at the censoring level, otherwise it remained uncensored. This procedure was repeated until the desired censoring level was achieved. The results of the 150 runs for each setting of simulation are summarized in Table 8.1, which shows the average parameter estimates and their corresponding mean square errors (MSE) from both the EM and BJ algorithms.

Overall the results of the EM algorithm are consistent with those produced by the BJ algorithm even when the distributional assumptions are violated. In general, the observed MSEs are similar for the EM and BJ algorithms. Both algorithms perform best when the errors are

normally distributed, as illustrated by the small bias and MSE for all censoring levels and sample sizes.

Table 8.1. Simulation Summary of EM and BJ Algorithms

Parameter	**Error Distribution**	**Sample Size**	**Censoring (%)**	**EM** $\widehat{\boldsymbol{\beta}}_{\mathbf{EM}}$ $(\mathbf{MSE}_{\mathbf{EM}})$	**BJ** $\widehat{\boldsymbol{\beta}}_{\mathbf{BJ}}(\mathbf{MSE}_{\mathbf{BJ}})$
$\boldsymbol{\beta_0}$	**N(0, 0.182)**	**150**	**10**	0.9993 (0.0007)	0.9993 (0.0007)
			30	1.0019 (0.0012)	1.0018 (0.0012)
			50	0.9994 (0.0010)	0.9989 (0.0010)
		60	**10**	0.9906 (0.0020)	0.9904 (0.0020)
			30	1.0036 (0.0026)	1.0035 (0.0026)
			50	0.9932 (0.0029)	0.9917 (0.0029)
$\boldsymbol{\beta_1}$		**150**	**10**	1.9970 (0.0025)	1.9970 (0.0025)
			30	2.0090 (0.0049)	2.0093 (0.0049)
			50	2.0173 (0.0048)	2.0184 (0.0049)
		60	**10**	2.0150 (0.0071)	2.0155 (0.0071)
			30	1.9974 (0.0090)	1.9976 (0.0091)
			50	2.0304 (0.0123)	2.0343 (0.0126)
$\boldsymbol{\beta_0}$	**U(-0.5, 0.5)**	**150**	**10**	1.0056 (0.0023)	1.0057 (0.0023)
			30	0.9990 (0.0026)	0.9993 (0.0026)
			50	0.9929 (0.0036)	0.9929 (0.0036)
		60	**10**	0.9913 (0.0055)	0.9912 (0.0055)
			30	1.0022 (0.0057)	1.0024 (0.0057)
			50	1.0041 (0.0080)	1.0036 (0.0079)
$\boldsymbol{\beta_1}$		**150**	**10**	1.9905 (0.0073)	1.9906 (0.0073)
			30	2.0110 (0.0086)	2.0117 (0.0086)
			50	2.0553 (0.0169)	2.0574 (0.0176)
		60	**10**	2.0094 (0.0145)	2.0098 (0.0146)
			30	2.0083 (0.0199)	2.0087 (0.0203)
			50	2.0511 (0.0319)	2.0533 (0.0325)
$\boldsymbol{\beta_0}$	**Exp(1) - 1**	**150**	**10**	0.9823 (0.0275)	1.0027 (0.0286)
			30	0.9488 (0.0334)	1.0147 (0.0383)
			50	0.8830 (0.0408)	1.0602 (0.0564)
		60	**10**	0.9807 (0.0542)	1.0015 (0.0573)
			30	0.9724 (0.0504)	1.0208 (0.0581)
			50	0.8418 (0.0954)	0.9478 (0.1098)
$\boldsymbol{\beta_1}$		**150**	**10**	2.0217 (0.0798)	2.0028 (0.0812)
			30	2.0949 (0.0939)	2.0852 (0.1058)
			50	2.4651 (0.3133)	2.4899 (0.3615)
		60	**10**	2.0270 (0.1582)	2.0117 (0.1639)
			30	2.0853 (0.1420)	2.0844 (0.1570)
			50	2.5147 (0.5209)	2.5406 (0.5925)

Additionally, both algorithms perform better with the large sample size than with the small sample size. Based on the censoring level, the algorithms perform better when a small percentage of the data are censored; higher censoring level increases the bias and MSE. In the case of uniformly distributed errors, the performances of the EM and BJ algorithms are comparable to that of a normal model when the sample size is large. Settings in which the errors are exponentially distributed show the worst performance for both EM and BJ algorithms, especially when the sample size is small and the censoring level is high. This is partly due to σ^2 being much higher for these simulations than the normal or uniform simulations.

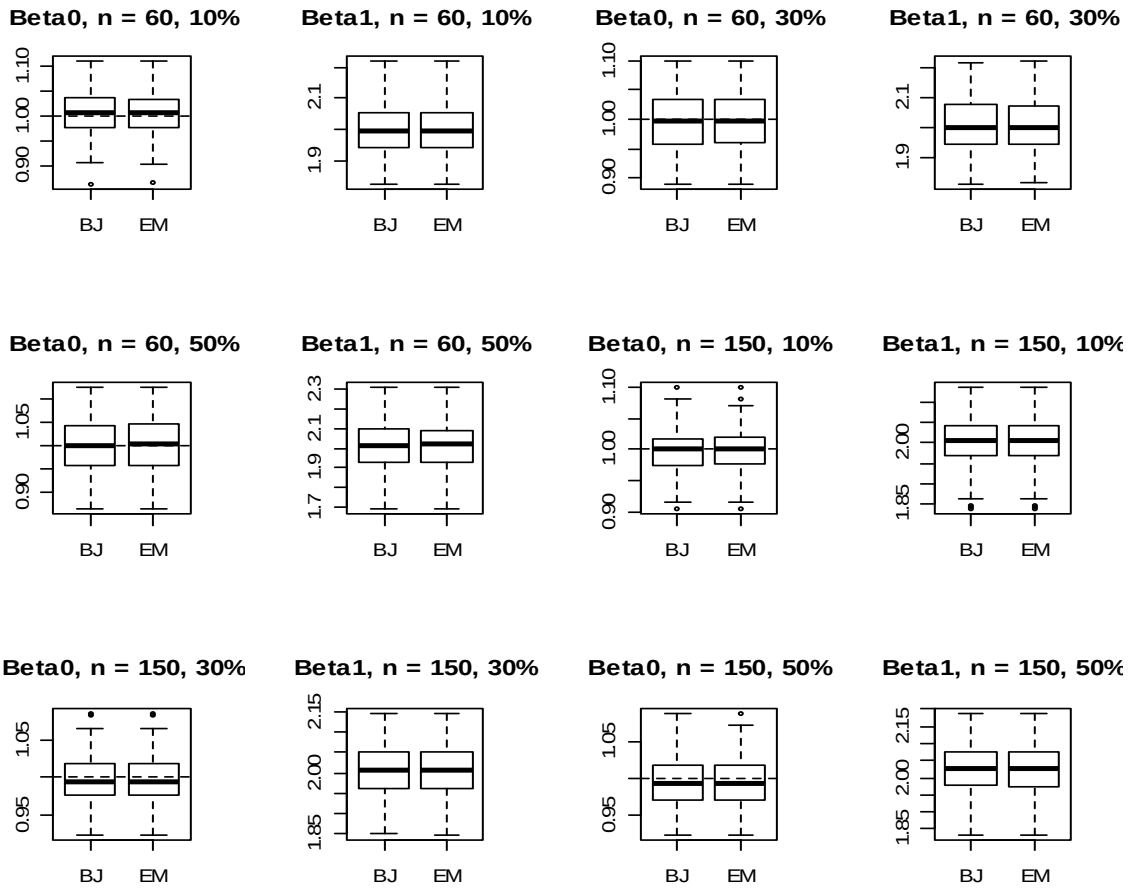

Figure 8.1. Box Plots of Parameter Estimates Produced by EM and BJ Algorithms for Different Sample Sizes and Censoring Levels in the Normal Model

It does not take a large number of simulations to verify the robustness of the EM algorithm and its good performance in the case of right censored regression for different models, sample sizes, and censoring amounts. Figure 8.2 shows box plots of the parameter estimates produced by the EM and BJ algorithms constructed side by side for different sample sizes and censoring levels assuming normally distributed errors.

Because of the similar performances of the EM and BJ algorithms in the estimation of the parameters, even when the assumptions underlying the EM algorithm are violated, we will use only the EM algorithm with normally distributed errors in the further simulation studies.

8.3. Sensitivity Analysis for Reconstructed R-squared

Several simulations were performed to evaluate the sensitivity of R-squared and reconstructed R-squared under different simulation settings. Three different scenarios were considered in regard to the location of the censoring points relative to the fitted line: censored points fall above and below the line, all censored points fall above the line, and all censored points fall below the line.

Data were simulated from the linear regression model:

$$y = \beta_0 + \beta_1 x + \varepsilon,$$

with $\beta_0 = 1$ and $\beta_1 = 2$.

The independent variable was designed such that $x_i = \frac{i}{n}$, $i = 1, 2, \ldots . n,$ where n is the sample size. Different simulation settings were created by manipulating the sample size ($n = 150, 60$),

the percentage of points censored (10%, 20%, 30%), and the random distribution of the error terms (N(0, 0.182), N(0, 0.5)). The censoring levels were modified from the first simulation study to accommodate distribution of points above and below the line (e.g. there is no guaranty that 50% of the generated data is located below the line).

Table 8.2. Summary of R-squares using N(0, 0.182)

Location of Censored Points	**Sample Size**	**Censoring (%)**	$\mathbf{R^2}$	$\mathbf{R_c^2}$	$\mathbf{R_{EM}^2(1)}$	$\frac{\Delta(\hat{\beta} - \hat{\beta}_{EM})}{\hat{\beta}}$
above and below the line	**150**	**10**	91.04%	75.41%	91.07%	(0.2762, -0.3629)
		20	91.12%	63.42%	91.11%	(0.2408, -0.5040)
		30	90.97%	54.35%	90.98%	(0.2132, -0.5616)
	60	**10**	91.21%	76.36%	91.23%	(0.2801, -0.4063)
		20	91.19%	63.82%	91.19%	(0.2838, -0.4409)
		30	90.96%	54.30%	91.07%	(0.2323, -0.5451)
above the line	**150**	**10**	91.04%	91.87%	90.97%	(0.3332, -0.3416)
		20	90.93%	92.71%	90.67%	(0.3119, -0.3524)
		30	91.03%	93.56%	89.94%	(0.3118, -0.3592)
	60	**10**	91.14%	91.92%	91.04%	(0.3299, -0.3400)
		20	91.15%	92.83%	90.79%	(0.3244, -0.3531)
		30	91.23%	93.67%	90.16%	(0.3227, -0.3682)
below the line	**150**	**10**	91.00%	80.47%	91.17%	(0.3025, -0.3300)
		20	91.20%	72.37%	91.64%	(0.2617, -0.3992)
		30	90.95%	63.93%	92.21%	(0.2313, -0.4625)
	60	**10**	91.14%	80.73%	91.27%	(0.2731, -0.4286)
		20	91.15%	72.14%	91.51%	(0.2889, -0.4058)
		30	90.99%	63.93%	92.29%	(0.2006, -0.4756)

The results of the 150 runs for each setting of simulation are summarized in Tables 8.2 and 8.3, which show average R-squared based on the original data (R^2), the average R-squared after censoring (R_c^2) treating censored values as uncensored, the average reconstructed R-squared ($R_{EM}^2(1)$**)** calculated using equation (7.1), and the average relative changes in the parameter estimates before and after the EM algorithm was run.

When the censored points are located below the fitted line, their reconstructed values are very close to the fitted line. The similarities of R^2 and $R^2_{EM}(1)$ show that the EM performs well in retrieving the original information in the data. The EM algorithm estimates the reconstructed value of a censored point as a conditional expectation given the point is above a censoring level. Suppose the censoring level is negative infinity. Then the conditional expectation is the same as unconditional expectation and the reconstructed value is the mean value.

Table 8.3. Summary of R-square using N(0, 0.5)

Location of Censored Points	Sample Size	Censoring (%)	R^2	R^2_c	$R^2_{EM}(1)$	$\frac{\Delta(\hat{\beta} - \hat{\beta}_{EM})}{\hat{\beta}}$
above and below the line	150	10	57.72%	49.81%	57.86%	(0.3397, -0.4365)
		20	57.24%	42.51%	57.54%	(0.2697, -0.5320)
		30	56.61%	36.82%	56.93%	(0.2455, -0.6254)
	60	10	58.32%	50.78%	58.32%	(0.2716, -0.3516)
		20	57.94%	43.92%	58.34%	(0.3353, -0.5078)
		30	57.13%	37.96%	57.14%	(0.1373, -0.7103)
above the line	150	10	57.15%	58.99%	57.07%	(0.3305, -0.3434)
		20	57.89%	62.48%	57.37%	(0.3357, -0.4226)
		30	57.88%	65.58%	56.18%	(0.2719, -0.4547)
	60	10	57.55%	59.46%	57.41%	(0.3671, -0.3610)
		20	58.22%	62.62%	57.63%	(0.3619, -0.3554)
		30	58.29%	65.31%	55.82%	(0.3213, -0.2982)
below the line	150	10	57.45%	53.09%	57.65%	(0.2289, -0.3189)
		20	57.86%	49.98%	58.71%	(0.2203, -0.3294)
		30	56.96%	44.41%	59.93%	(0.0429, -0.3610)
	60	10	58.29%	54.73%	58.95%	(0.2729, -0.4019)
		20	57.99%	50.53%	59.26%	(0.2614, -0.2409)
		30	57.95%	45.80%	61.67%	(0.1729, -0.5699)

This can also be observed from equation (5.2). If the censoring level is negative infinity the second term in the equation (5.2) goes to zero resulting in the conditional expectation being equal to its unconditional expectation or the mean. If the censoring level is low but larger than negative infinity, then the conditional expectation will be little bit bigger than the unconditional

expectation, but the reconstructed value will still end up being calculated approximately close to the average values and as such will be placed very close to the fitted line.

When the censored points are located above the line their reconstructed values are even higher. The $R^2_{EM}(1)$ results are very similar to the original R^2 results. Thus the EM performs very well in retrieving the original information in the data.

In cases where the censored points fall both above and below the line and when the censored points fall only below the line, $R^2_{EM}(1)$ is similar to the original R^2. A higher level of censoring combined with increased variability decreases R^2_c and $R^2_{EM}(1)$, resulting in a poorer model fit in general. Missing information in the censored data recovered by the EM algorithm adds credibility to the modeling compared to the modeling when censoring is ignored.

It is also observed that the reconstructed values for the censoring points change the parameter estimates in such a way that the line rotates slightly from the fitted line of full data resulting in an increase in slope and a decrease in the intercept. On average, censoring impacts the slope coefficient more than the intercept coefficient. The estimated slope will increase when a higher level of censoring is present in the data. This is impacted by the reconstructed values of the censored observations being always at or above the censoring level.

The importance of these findings is significant in insurance applications involving pricing based on the fitting loss models through the individual data. Censoring in insurance will depend on the type of policy and coverage provided to the policyholder. For example, limits on "Jewelry" coverage may be $2,500, $5,000, or $10,000 which is relatively low compared to "Homeowner's" policy limits in the range of $100,000-$500,000. However, this is relatively

lower than "Medical Liability" limits that can go from $1-$5 million. Thus censoring relative to the fitted line can occur with any of the three scenarios presented earlier.

Overall, we conclude that the EM algorithm generally restores the R-squared value well when all censored values fall below the line or censored values fall both below and above the line for censoring level up to 30%. Thus, $R^2_{EM}(1)$, which is the same as $R^2_{EM}(p)$ proposed by formula (7.1), can be used as a valuable tool in model validation.

8.4. Outlier Detection via EM Algorithm

Several simulations were run in order to assess to what extent the EM algorithm is capable of detecting outliers. Three different scenarios were considered in regard to location of the censoring points relative to the fitted line: censored points fall above and below the line, all censored points fall above the line, and all censored points fall below the line.

Data were simulated from the linear regression model:

$$y = \beta_0 + \beta_1 x + \varepsilon,$$

with $\beta_0 = 1$ and $\beta_1 = 2$.

The independent variable was designed such that $x_i = \frac{i}{n}$, $i = 1, 2, \ldots . n,$ where n is the sample size. Different simulation settings were created by manipulating the sample size ($n = 150, 60$), the percentage of points censored (10%, 20%, 30%), and the random distribution of the error terms (N(0, 0.182), N(0, 0.5)).

Once the data was generated, 4 and 2 artificial uncensored outliers are added to the data with sample sizes 150 and 60, respectively. These outliers were chosen randomly along the

interval and had values between 4 and 6 standard deviations from the fitted line. At this point the total number of outliers was counted, where an observation was to be an outlier if its value was larger than 2 standard deviations from the fitted line. Next, the censoring was applied. Censored points were selected at random (excluding the artificial outliers) above and below the line on the entire interval. Their values were compared to the censoring level randomly drown from U (1, 3). If a value of a selected data point was above the censoring level, it was trimmed at the censoring level, otherwise it remained uncensored. This procedure continued until the desired censoring level was achieved. A similar trimming approach was considered when censored points fell above the line or when censored points fell below the line. The EM algorithm was run, and the formulas proposed in Section 7.2 were used to determine which observations were outliers. The number of outliers detected was then recorded. The total number of outliers over all 150 runs for each setting of simulation is summarized in Tables 8.4 and 8.5. The number of observations declared to be outliers was counted (before censoring), after censoring, and after the EM algorithm was run. These counts correspond to the last three columns (A, B, and C) of Tables 8.4 and 8.5. Generally, the EM algorithm returned an outlier count similar to that generated with the original data, which confirms a good performance of the EM algorithm in outlier detection using the proposed formulas from Chapter 7. All artificial outliers were recognized in each simulation. This is not surprising considering that their location is far from the fitted line.

When censored observations were above the line, the observed relationship for columns A, B, and C based on the outlier count is such that $B \subset A \subset C$. Column C has a higher number compared to column A due to the few censored observations that became outliers after their

reconstructed values were estimated. Generally, if the censored points were below the line, the observed relationship for columns A, B, and C is such that $A \subset C \subset B$.

Table 8.4. Summary of Outlier Detection Based on N(0, 0.182) when Artificial Outliers are Uncensored Observations

Location of Censored Points	Sample Size	Censoring (%)	No. Outliers in the Beginning (A)	No. Outliers After Trimming (B)	No. Outliers From EM (C)
above and below the line	150	10	607	616	611
		20	605	633	608
		30	607	645	609
	60	10	302	318	305
		20	300	320	302
		30	302	341	308
above the line	150	10	607	597	608
		20	603	590	603
		30	604	587	604
	60	10	301	296	305
		20	303	278	305
		30	302	283	303
below the line	150	10	608	693	607
		20	609	755	605
		30	608	727	606
	60	10	300	315	300
		20	301	316	302
		30	300	311	300

Some censored points below the line were reported as outliers with the original data, but their reconstructed values were on the fitted line so they were not counted as outliers in the final count. Column B has a higher count than column A due to the trimming of censored observations which values will fall lower than their actual values.

Table 8.5. Summary of Outlier Detection Based on N(0, 0.5) when Artificial Outliers are Uncensored Observations

Location of Censored Points	Sample Size	Censoring (%)	No. Outliers Before Trimming (A)	No. Outliers After Trimming (B)	No. Outliers From EM (C)
above and below the line	150	10	603	627	615
		20	604	668	613
		30	608	681	612
	60	10	301	323	305
		20	301	337	302
		30	302	341	306
above the line	150	10	610	603	614
		20	604	607	606
		30	602	600	608
	60	10	300	297	301
		20	302	291	302
		30	302	300	302
below the line	150	10	611	701	612
		20	611	735	610
		30	604	741	604
	60	10	302	311	303
		20	303	313	303
		30	301	310	301

Additional simulations were performed with assumptions that the artificial outliers represent censored observations within 4 to 6 standard deviations above the fitted line. Adding censored outliers below the fitted line does not make sense considering that the EM algorithm would calculate their reconstructed values to be close to the line and they would not be identified or counted as outliers. The results of these simulations are summarized in Tables 8.6 and 8.7 for each simulation setting. Again, all artificial outliers were detected which confirms that the performance of the EM algorithm is consistent with the previous findings.

Table 8.6. Summary of Outlier Detection Based on N(0, 0.182) when Artificial Outliers are Censored Observations

Location of Censored Points	Sample Size	Censoring (%)	No. Outliers Before Trimming (A)	No. Outliers After Trimming (B)	No. Outliers From EM (C)
above and below the line	150	10	610	617	610
		20	611	639	612
		30	605	640	606
	60	10	300	310	300
		20	300	315	301
		30	300	323	300
above the line	150	10	605	597	605
		20	609	591	610
		30	610	601	610
	60	10	300	285	300
		20	302	288	302
		30	300	293	300
below the line	150	10	604	697	604
		20	605	736	605
		30	607	740	607
	60	10	301	317	302
		20	301	325	302
		30	300	330	300

Table 8.7. Summary of Outlier Detection Based on N(0, 0.5) when Artificial Outliers are Censored Observations

Location of Censored Points	Sample Size	Censoring (%)	No. Outliers Before Trimming (A)	No. Outliers After Trimming (B)	No. Outliers From EM (C)
above and below the line	150	10	609	619	609
		20	609	636	610
		30	607	645	611
	60	10	300	311	302
		20	301	323	301
		30	302	342	304
above the line	150	10	609	600	609
		20	609	601	610
		30	610	600	611
	60	10	310	300	310
		20	309	301	311
		30	302	297	302
below the line	150	10	607	639	609
		20	609	641	609
		30	608	657	608
	60	10	300	310	300
		20	301	313	301
		30	300	325	300

8.5. Influence Diagnostic: One-step Deletion Method via EM Algorithm

In order to validate the formula (7.7) for assessing the influence of individual observations on the parameter estimates, several simulations were performed. This section presents the summary of these results. Simulations were designed as follows.

Data were simulated from the linear regression model:

$$y = \beta_0 + \beta_1 x + \varepsilon,$$

with $\beta_0 = 1$ and $\beta_1 = 2$.

The independent variable was designed such that $x_i = \frac{i}{n}$, $i = 1, 2, \ldots . n,$ where n is the sample size. Different simulation settings were created by manipulating the sample size ($n = 150, 60$), the percentage of points censored (10%, 20%, 30%), and the random distribution of the error terms N(0, 0.182).

Censored points were selected at random (excluding the influential points) above and below the line on the entire interval. Their values were compared to the censoring level randomly drown from U (1, 3). If the value of a selected data point was above the censoring level, it was trimmed at the censoring level, otherwise it remained uncensored. This procedure was repeated until the desired censoring level was achieved. This random censoring resulted in censored points being above and below the fitted line. Now, two scenarios were implemented in order to quantify the influence based on the one-step deletion method:

(1) The magnitude of influence for each individual observation was calculated based on the data generated with the stated assumptions above
(2) The magnitude of influence for each individual observation was calculated after two influential observations were created in the same data set in (1).

Two influential observations in (2) were randomly selected across the entire interval and their values were randomly generated from U(5, 15) to replace the values of the original observations. The estimates for each component of the parameter vector were compared before and after two influential observations were created in the data. Figure 8.2 shows a snapshot of one simulation with 60 observations and 10% censoring using N(0, 0.182) as the distribution of the error terms. The magnitude of the influences before influential observations were added is relatively small for each observation (graphs in the first column of Figure 8.2). After adding two influential

observations, their influences stand out compared to all other observations in the sample (graphs in the second column of Figure 8.2).

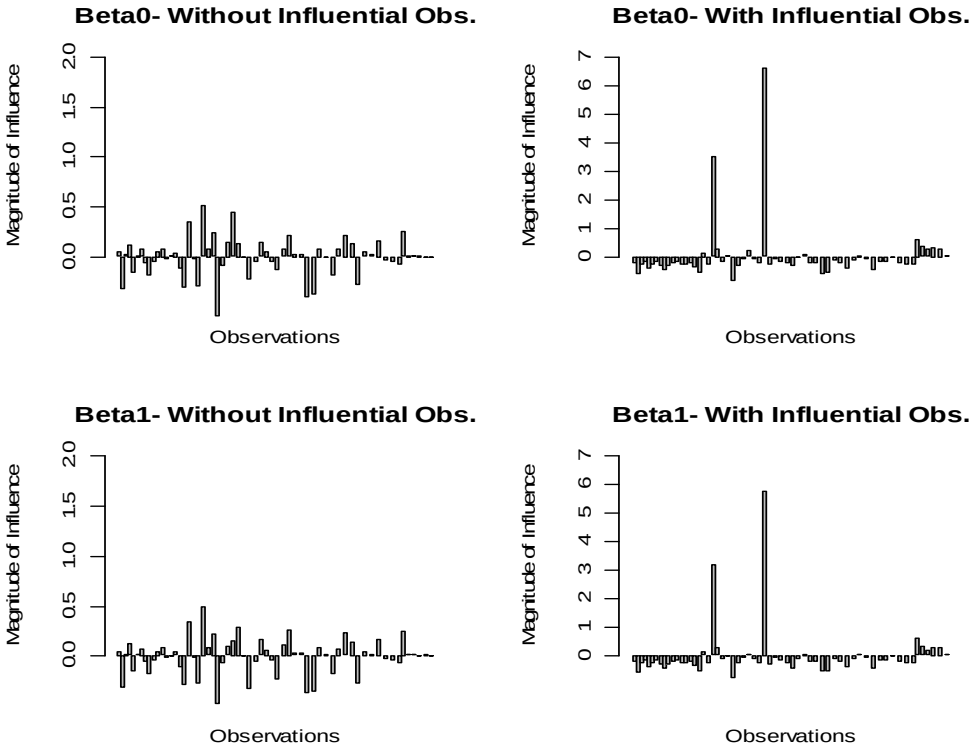

Figure 8.2. Influence Diagnostics: One Simulation Run With 60 Observations and 10% Censoring Using N(0, 0.182)

The two influential observations have a large influence on both the intercept and slope of the model compared to the other observations and are easily detected by visual inspection. A summary of all simulation runs for each setting is shown in Figure 8.3. Side-by-side box plots for each simulation setting are displayed for each parameter under two different scenarios using N(0, 0.182) as the distribution of the error terms. Data used to construct the box plots represent the influences from all observations over 150 runs for each simulation setting.

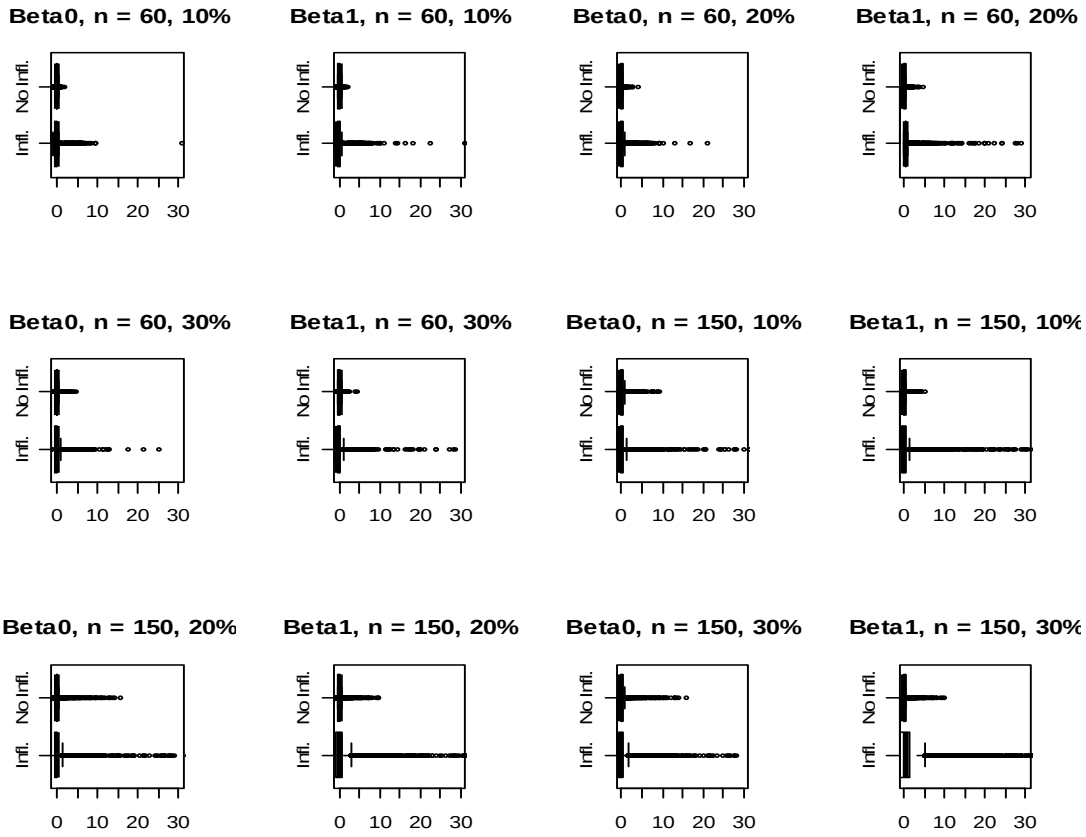

Figure 8.3. Summary of Influence Diagnostics by Simulation Setting for N(0, 0.182)

The box plots that were constructed using data without influential points reflect slightly skewed distributions with a longer right whisker and many outliers. The length of the right whisker depends on the censoring level and sample size. A higher percentage of censoring combined with a larger sample size results in a larger number of outliers. These distributions have small variance compared to data set without influential points as the width of boxes are very small (difference between upper and lower quartile) for all simulation settings. The box plots that were constructed using two influential points are severely skewed right with extremely

long right whiskers compared to the distributions developed for data sets without influential points.

Clearly, two different types of distributions are observable when influence diagnostics is measured between the two scenarios: data without influential points and data with influential points. These results confirm that the one-step deletion formula (7.7) based on the EM algorithm works and it is capable of identifying and quantifying influential points in a data set.

CHAPTER 9. APPLICATION TO ND DVA DATA

9.1. Introduction

There were 22.5 million living veterans in the United States as of 2010, representing 7.3% of the total population. Veterans are eligible for a number of federal and state benefit programs and services offered by the Department of Veterans Affairs (DVA). According to the U.S. Census Bureau, the uninsured rate of veterans decreased from 7.6% in 2000 to 7.2% in 2009. As federal and state medical health benefits are available to eligible veterans, the number of veterans 18 years and older using these programs increased from 50% in 2000 to 60% in 2009. The availability of these programs is critical for veterans who live below poverty level. Poverty rate among veterans, defined as income below 100% of poverty threshold, has increased over the past decade, and it was reported at 6.3% in 2009 compared to 5% in 2000. The Bureau of Labor Statistics reported that in 2007, 11.8% of North Dakota's population was living below the poverty level. The national average for the same period was 13%. Considering all of this, a researcher might want to answer the question "What are the health benefit needs of the veterans' population in North Dakota"?

State benefit programs for veterans vary from state to state. In the state of North Dakota, ND DVA, working under the supervision of the Administrative Committee of Veterans Affairs (ACOVA), administers various state benefit programs available to low income veterans and their families. The Hardship Grants Program provides aid to veterans for unmet medical needs and encompasses medical benefits for the following categories: dental, denture, hearing, optical, and special. The cost of this program is underwritten by the Veterans Post War Trust Fund (VPWTF). The State Treasurer is the trustee of this fund, as provided for in the state constitution.

This fund relies on its investments in the financial market in order to grow and generate annual income for use in grant programs that will benefit veterans. The ND DVA is responsible for the administration of these programs. The policy and guidelines of these programs are set by the ACOVA whose board is made up of veterans. In order to prudently manage the fund and budget Hardship Grants Program, it is important to evaluate the medical benefit needs of veterans in North Dakota so that appropriate decisions are made at the state level to generate sufficient funds to pay eligible veterans and their families in future years. This study provides statistical models and tools which can be applied in the financial assessment of the medical benefit needs for veterans in North Dakota and may be used in any other U. S. state where similar programs exist. Government and policy makers may also be interested in this study as they want to make decisions and provide sound investments for future public policies.

9.2. Data Set

Data used in this study were provided by ND DVA. Typically, categories of health benefits available to veterans are capped (right censored) or limited at certain level. The censoring points change over time, as they are subject to review and state approval, and they may vary across different categories. For any claim, if the expense exceeds the amount granted, it will be reimbursed at the value of granted amount.

Medical grants are subject to a limit and the annual amount of benefits is capped (right censored). The data provided consist of payment amounts granted to each applicant for years 2000 through 2010. Table 9.1 shows the variables provided and their descriptions.

Table 9.1. Summary of ND DVA Data

Variable	**Description**
VoucherDate	Day/Month/Year when the benefit payment is made
Gender	Male (0) or Female (1)
ApplicationDate	Day/Month/Year when the application was filed
ApprovedDate	Day/Month/Year when the application was approved
BirthDate	Birth date of each applicant
AmountGranted	Amount approved by the grant program
Category	Category of benefits (dental, denture, hearing, optical, and special)
ApplicantTB	Applicant's unique non-identifiable ID
MonthlyIncome	Applicant's income per month
Status	Status of a person receiving benefits (v- primary beneficiary (veteran), vs- spouse of a living veteran and w- widow/widower). These are codded as: 000-veteran; 010- spouse; 001- widow/er.
NoDependants	Family size including applicant (seven levels: 1, 2, 3, 4, 5, 6, and 7)
AmountPaid	Benefit amount paid
ZipCode	5-digit postal code of the applicant address
County	County code of the applicant address
CountyName	County name of the applicant address

About half of the variables listed above were of interest to our project. The difference between application year and birth year was used to determine the applicant's age. Year when the application was approved was extracted from the approved date. An applicant is given only 90 days to use the grant. In this case approved date and voucher date are only three months apart, and the data are available only for those applicants who actually used the grants. Dates for others who have not managed to use the grant were provided as cancellations and were ignored in this study. The amount of money granted as well as the amount of money given from 2000 to 2010 by ND DVA is adjusted for inflation using the Consumer Price Index (CPI) published by the Bureau of Labor Statistics, U.S. Department of Labor.

Historically, benefit categories carry different benefit caps (limits) on an annual basis. Dental benefits started with a $500 cap as of 12/2004, then increased to $750 as of 1/2006, and finally reached $1000 as of 11/2007. Dental services sometime require more than one

appointment; in this case applicants receive several payments during the year. Therefore, the data for dental category were aggregated by year and applicant. The data for dentures, hearing, optical, and special categories of benefits were excluded since they contained significantly lower number of records and as such they may not be reliable.

The ND DVA uses monthly income level and family size to determine if an applicant meets benefit eligibility criteria. Each income level corresponds to a certain family size. For example, a family of two earning less than $1400 per month, or a family of eight earning less than $2600 per month, would be eligible for benefits. Many records were missing family size but had income level provided. For this reason, we used income level only and ignored family size as these two variables seem to be correlated.

Dental records show that the applicants' age vary from 24 to 94 with 84% of the individuals being older than 50. Men represent 287 applicants compared to 81 women. Based on status, 26 applicants are spouses of living veterans and 33 applicants are widows or widowers of veterans. Living veterans represent 309 individuals or 84% of the sample. It is observed that 34 individuals or 9.2% of the sample reported zero income. The highest income reported is $2600 per month for a large family. Thus, most of these people live below the poverty level. The poverty guidelines are issued each year in the Federal Register by the Department of Health and Human Services (HHS). The 2008 income threshold by family size, reported by HHS, for the 48 contiguous states is summarized in Table 9.2.

Table 9.2. Department of Human Services –Poverty Guidelines

Household Size	1	2	3	4	5	6	7
Annual Income Level	$10,400	$14,000	$17,600	$21,200	$24,800	$28,400	$32,000

North Dakota had 11.8% of its total population living below the poverty level in 2007 compared to the national average of 13% reported for the same period. Poverty guidelines determined by ACOVA on the basis of national statistics are reported in Table 9.3.

Table 9.3. Eligibility Requirements Set by ACOVA

Household Size	1	2	3	4	5	6	7
Annual Income Level	$14,400	$16,800	$19,200	$21,600	$25,200	$28,800	$31,200

There were 575 annual aggregate applications for dental benefits used by 368 different individuals for years 2000-2010. We identified 274 (48%) applications with a paid amount in benefits equal to or higher than the amount granted. These policies represent right censored data. For uncensored data records, paid amount in benefits was greater than zero and less that the defined limit (cap or censoring point).

Finally, the following variables were selected for inclusion in the modeling of dental benefits: year, age, gender, amount granted adjusted for inflation, censored amount adjusted for inflation, income level, and applicant's status. Application year, age, gender, income level, and applicant's status represented explanatory variables while the amount paid (adjusted for inflation) was used as a response variable in the model.

9.3 Analysis

The EM and BJ algorithms were applied to illustrate the modeling of veterans' health benefits with a special focus at dental on the benefit category. First, the right censored regression model was considered with all explanatory variables. That is:

$$E(Benefit\ Paid) = \beta_0 + \beta_1(Application\ year) + \beta_2(Age) + \beta_3(Gender) + \beta_4(Income\ Level) + \beta_5(Spouse) + \beta_6(Widower)\ . \quad (9.1)$$

The EM algorithm, employed in modeling parameter estimates and variability assessments, indicated that gender, age, income level, and spouse were not significant predictors of the paid benefits. Application year and widow /er were significant predictors with the possibility of application year entering the model as a quadratic term. The parameter estimates (and their significance) of this model are shown in Table 9.4.

Table 9.4. Parameter Estimates for the Full EM Model. * Indicates ≤ 0.05 significance.

EM_Parameters	EM_Estimates	EM_95% CI	EM_p-value
Intercept	329.60	(116.36, 542.83)	0.0024*
Application year	58.37	(42.66, 74.07)	0.0000*
Age	-0.31	(-3.30, 2.68)	0.8391
Gender	88.19	(-34.30, 210.69)	0.1582
Income Level	0.05	(-0.03, 0.14)	0.2422
Spouse	-54.12	(-223.15, 114.91)	0.5303
Widower	-157.45	(-328.90, 13.99)	0.0718

Paid benefit amount is plotted against application year for the simple linear and quadratic models in Figure 9.1. Bold black points represent censored observations. For both the simple and quadratic models, dotted lines show the fitted trend ignoring the information from the censored observations. Bold dark lines are created from the estimated fits produced by the EM algorithm. A curvature trend is obvious in the data. Dotted lines show the trend based on the actual data. As expected, right censored observations will shift the trend upward due to the inclusion of the estimated values from the missing information. Thus the difference between the two lines on the same graph represents the amount of missing information due to right censoring, estimated via EM algorithm.

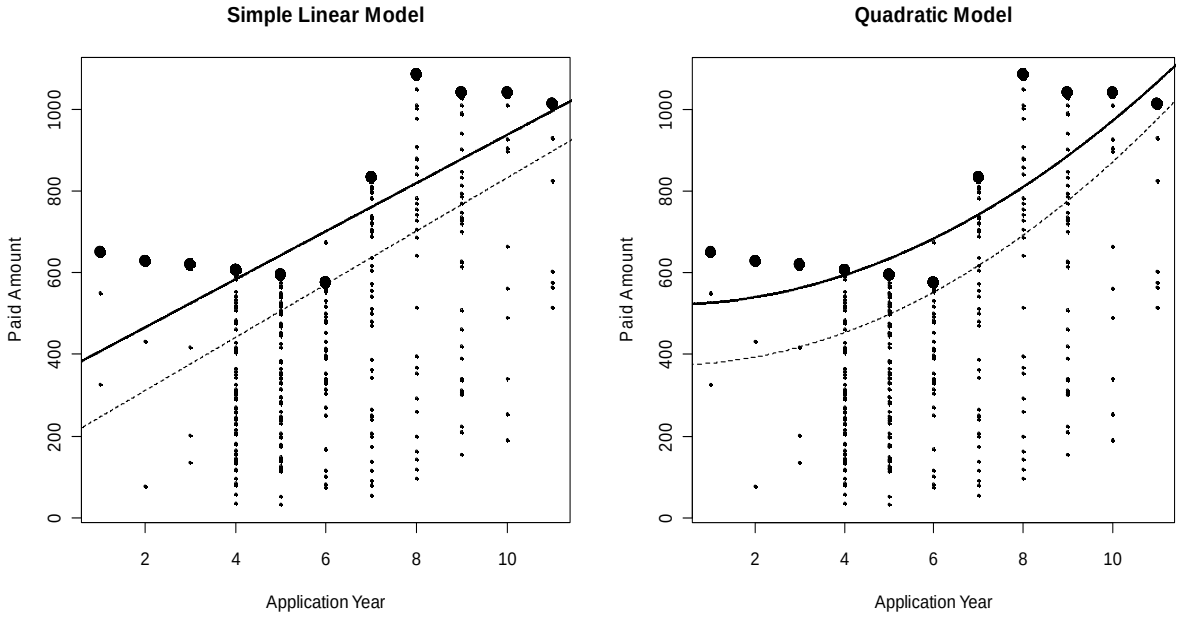

Figure 9.1. Trend in Paid Benefit Amount by Application Year

In subsequent model selections, five additional models were examined. A summary of the log likelihood values, AIC, and BIC results for these models are shown in Table 9.5. The minimum values of AIC and BIC are reported for Model-6, which is proposed to be the best model.

Parameter estimates for Model-6 with their confidence intervals and corresponding p-values are summarized in Table 9.6. If we consider the same portfolio of applicants, the total dental benefit needs of ND veterans for the period 2003-2009, calculated based on the EM algorithm, was $407,562 compared to the amount of $333,472 actually spent. The difference of $74,090 can be used to help ACOVA increase the cap on benefits in the future and suggest to the State Treasurer that additional investments were needed in funding this grant program.

Table 9.5. Summary of Different Criteria Used in the EM Model Selection

Model	Log-likelihood	AIC	BIC
Model-1	-2126.44	4268.88	4303.72
Model-2	-2128.37	4264.74	4282.16
Model-3	-2129.39	4264.78	4277.84
Model-4	-2128.18	4264.37	4281.78
Model-5	-2129.14	4264.28	4277.35
Model-6	-2126.92	4222.08	4243.85

Model-1: Full Model equation (9.1)
Model-2: $E(Benefit\ Paid) = \beta_0 + \beta_1(Application\ year) + \beta_6(Widower)$
Model-3: $E\ (Benefit\ Paid) = \beta_0 + \beta_1(Application\ year)$
Model-4: $E\ (Benefit\ Paid) = \beta_0 + \beta_1(Application\ year)^2\ + \beta_6(Widower)$
Model-5: $E\ (Benefit\ Paid) = \beta_0 + \beta_1(Application\ year)^2$
Model-6: $E\ (Benefit\ Paid) = \beta_0 + \beta_1(Application\ year)^2 + \beta_2(Gender) + \beta_3(Widower)$

Model-6 suggests that gender is a non-significant variable. According to this model, widowers generate $143.42 less in benefit payments on average compared to a living veteran or a spouse of a living veteran. On average, female applicants require $71 more in benefits compared to a male applicant. While there is a larger proportion of a male veterans compared to female veterans or dependents, it seems that females are using benefits more than males. Benefits are also a function of money that is available in the state budget for that purpose. When more money is available in state budgets more needy veterans will potentially benefit.

Table 9.6. Parameter Estimates for the EM Model-6. * Indicates ≤ 0.05 significance.

EM_Parameters	EM_Estimates	EM_ 95% CI	EM_ p-value
Intercept	522.45	(457.54, 587.36)	0.0000*
(Application year)^2	4.41	(3.25, 5.58)	0.0000*
Gender	71.00	(-23.25, 165.26)	0.1300
Widower	-143.42	(-286.90, 0.05)	0.0500*

The data show that in more recent years, a higher amount of money was available for spending even when the benefits are adjusted for inflation. The intercept coefficient provides us with a fixed cost per person for running this program. In other words, the ND DVA paid, based

on Model-6, an amount of about $522.45 per applicant/ per year irrespective of the number of applicants and their characteristics. We observe that income level is an insignificant predictor of benefits used. If the overall veteran population was considered in the analysis, one might expect that the lower income veterans are the most likely to use the benefits. However, most veterans eligible for benefits have income below the 100% poverty threshold. Hence the income level is very low and it does not segregate people further into subgroups. Age is another insignificant variable in Model-6 suggesting that benefits are used across all age groups 23-94. This analysis helps our understanding of what are the determinants of the distribution of the available benefit funds. It also helps us determine the total benefit need of the veteran population in ND.

The reconstructed coefficient of determination for Model-6 is 10.75%, lower than the coefficient of determination of 25.74% for the same model when censoring is ignored. The overall fit is poor but this is due to the large variability observed in the data set and the large proportion of censored points being above the fitted line. However, the results are in line with the findings obtained from the simulation study in Chapter 8. In addition, the reconstructed values for the censored observations can be used to validate the reasonability of the existing benefit caps. Based on the selected model, one can obtain more information about the average amount of expenses in excess of the existing cap.

The EM results above were benchmarked using the BJ method. Even though an R-library for Buckley-James estimation is available in R with BJ functions, an independent R-code was built in R (Appendix) and the results were compared to those generated by the commercially available R-functions. The differences in the results between independent programing and automated R-functions were found to be negligible. Model selection based on the R-squared formula equation (6.3) is consistent with the results of the EM algorithm above in a sense that

Model-6 is selected to be the best fitting model. The R-squared for Model-6 is 23.02%. Gender and widow/er have p-values very close to 5% , so they are considered significant predictors.

Table 9.7. Parameter Estimates for the Full BJ Model. * Indicates ≤ 0.05 significance.

BJ_Parameters	BJ_Estimates	BJ_95% CI	BJ_p-value
Intercept	318.37	(235.76, 400.97)	0.0001*
Application year	57.20	(50.82, 63.58)	0.0000*
Age	-0.44	(-1.60, 0.72)	0.7009
Gender	91.01	(37.42, 144.59)	0.0894
Income Level	0.06	(0.02, 0.095)	0.0889
Spouse	-55.90	(-124.72, 12.93)	0.4167
Widower	-158.60	(-227.29, -89.91)	0.0209*

Table 9.8. Parameter Estimates for the BJ Model-6. * Indicates ≤ 0.05 significance.

BJ_Parameters	BJ_Estimates	BJ_95% CI	BJ_p-value
Intercept	342.05	(300.49, 383.61)	0.0000*
Application year	56.38	(50.09, 62.66)	0.0000*
Gender	73.59	(35.41, 111.77)	0.0507
Widower	-151.77	(-205.05, -98.49)	0.0537

We observe that the Buckley-James variance estimate produces smaller values compared to those generated by the EM algorithm. The Buckley-James variance estimate accounts for uncensored observations only, while the EM algorithm approach is based on the likelihood of both uncensored and censored observations. Parameter estimates produced by the BJ model are somewhat smaller than those produced by the EM model. Also application year did not enter this model as a quadratic term. Thus from a policy maker's prospective, the EM model is more conservative and it should be the preferred choice between these two models.

9.4. Application of the New Influence and Diagnostics Tools

Formulas proposed in Chapter 7 were applied in order to identify outliers in the data and analyze the influence diagnostics on the parameter estimates. Six uncensored outliers (1% of the total number of observations) were found in the data. These outliers had t-values above the

critical value of 1.96 used for their detection. After careful inspection of the data, it was found that these observations reported extremely low amounts of benefits in the range of $31 to $75. Without additional knowledge as to whether these observations are results of errors or true benefit values, it was decided that they should not be removed.

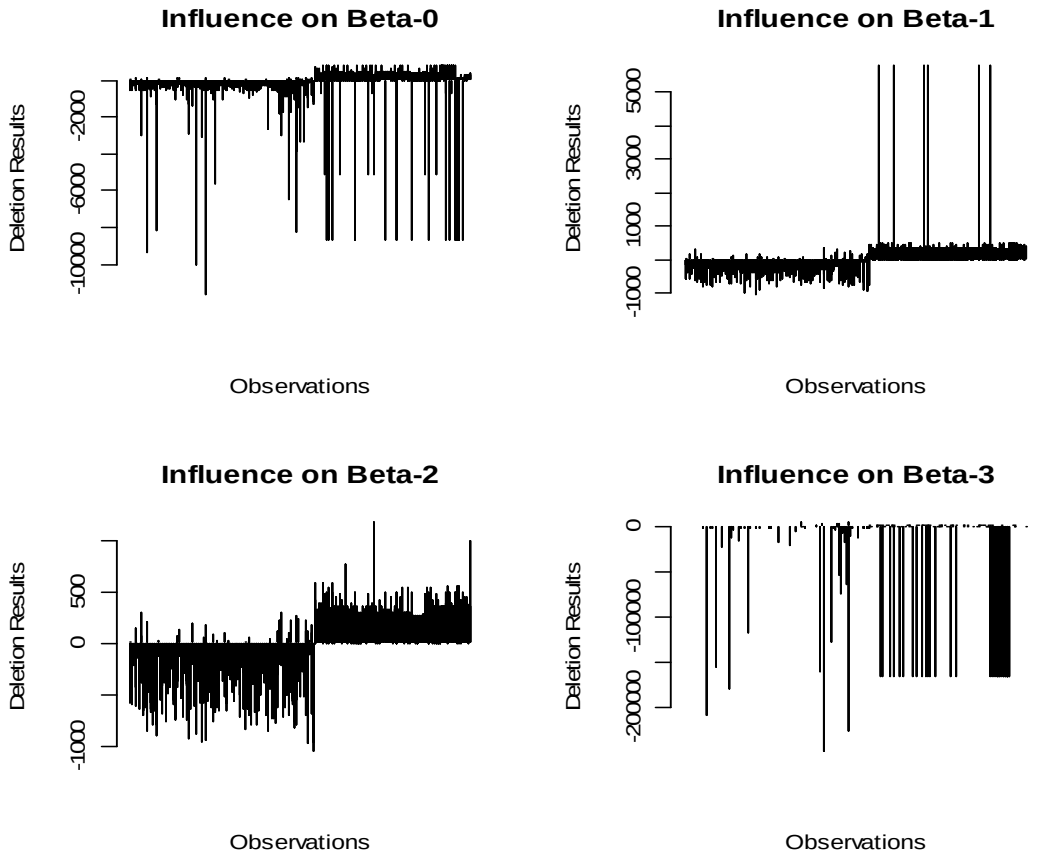

Figure 9.2. One-step Deletion Results for Four Parameters of the EM Model-6

Influence diagnostics based on the one-step deletion method were applied to the ND DVA dental data. Formula (7.7), proposed in Chapter 7, was used in these calculations. The results for the four parameters from Model-6, based on all 575 data points, are plotted in Figure 9.2. The highest spikes correspond to the most influential points. By careful inspection, it was found that these influential points correspond to most of the censored data reported for years

2006 and 2007 as well as uncensored outliers from these years. If we recall that the cap on dental benefits increased from $500 to $750 as of 1/2006 and further increased from $750 to $1000 as of 11/2007, these results are expected. The jumps in the censoring levels as well as several uncensored outliers explain the high influence of the corresponding observations on parameter estimates.

CHAPTER 10. CONCLUSION

The primary goal of this research was to study two algorithms for right censored regression with application to actuarial science. These algorithms include EM and BJ algorithms. The research contribution was made in area of validation and influence diagnostics based on the EM algorithm. The following quantities were proposed: the reconstructed coefficient of determination (R-squared), the Jackknife residual and test for outliers, and influence diagnostics based on the one-step deletion method.

Extensive simulation studies were performed to compare the model parameter estimates of the EM and BJ algorithms. It was found that the EM algorithm performs very similar to the BJ algorithm with the best performance achieved in the case of normally distributed errors. Simulation studies also showed that the EM algorithm can improve the R-squared for the model when the data are censored low (below the fitted line) and when the data are randomly censored above and below the fitted line with larger proportion of points below the fitted line. These simulation studies used the new reconstructed R-squared formula. The EM algorithm is also capable of detecting outliers. Several cases were examined based on the type (censored or uncensored) and location of the outliers, which confirmed that the EM algorithm successfully detects the outliers based on the proposed formulas. Influence diagnostic based on the proposed formula for one-step deletion method was analyzed using insurance data on fire losses. This formula successfully assessed the magnitude of influence of each data point on the parameter estimates, with influential points reporting the highest influence.

Finally, real data provided by ND DVA was used to model right censored regression based on both the EM and BJ algorithms. Further model validation and diagnostics were

employed for the EM algorithm only. As the results of modeling ND DVA data would be useful to the government policy makers, these methods in general can be used in actuarial science. As it is common to see an insurance product with coverage being subject to a certain limit, modeling right censored regression with the EM algorithm would allow an actuary to evaluate losses in the presence of rating variables and therefore determine the appropriate premium level to be charged.

REFERENCES

Aitkin, M. (1981). A note on Regression Analysis of Censored Data. Technometrics, Vol. 23, No. 2.

Akaike, H. (1974). A new look at the statistical modeling identification. IEEE Transactions on Automatic Control, vol. 19, 716-723.

Aziz, N. and Wang D. Q. (2009), A Renovated Cook's Distance Based on the Buckley-James Estimate In Censored Regression. World Academy of Science, Engineering and Technology 29.

Buckley, J. and James, I. (1979). Linear regression with censored data. Biometrika, 429-36.

Bureau of Economic Analysis, Bureau of Labor Statistics, U.S. Census. http://www.fedstats.gov/qf/states/38000.html

Bureau of Economic Analysis (BEA); US Department of Commerce. www.bea.gov

Chatterjee, S. and Hadi, A.S. (1986). Influential Observations, High Leverage Points, and Outliers in Linear Regression. Statistical Science, Vol.1, No. 3, 379-416.

Cook, R. D. (1977). Detection of Influential Observation in Linear Regression. Technometrics, Vol.19, No.1.

Cox, D. R. (1972). Regression Models and Life-Tables. Journal of the Royal Statistical Society, Vol. 34, No. 2, 187-220.

Dempster, A. P. , Laird, N.M., and Rubin, D. B.(1977). Maximum Likelihood from Incomplete data Using EM Algorithm. Journal of the Royal Statistical Society, Series B, vol. 39, 1-38.

Glasson, S.(2007). Ph.D. Dissertation. RMIT University.

Guiahi, F. (2001). Fitting Loss Distributions in the Presence of Rating Variables. Journal of Actuarial Practice, Vol.9, 97-129.

Hahn, G. J. and Nelson, W. B. (1974). A comparison of methods for analyzing censored life data to estimate relationship between stress and product life. Transactions on Reliability, R-23, 2-11.

Hocking, R. R. (2003). Methods and applications of linear models: regression and the analysis of variance. Wiley. Hoboken, NJ.

Kaplan, E. L., & Meier, P. (1958). Non-parametric estimation from incomplete observations. JASA, 53, 457-81.

Keiding, N. (1998). The Cox Regression Model for Claims Data in Non-Life Insurance. AUSTIN Bulletin, Vol. 28, No.1, 95-118.

Klein, J. P. and Moeschberger, M. L. (2003). Survival Analysis. New York: Springer.

Klugman, S.A. , Panjer, H.H. , Willmot, G.E. (2004). Loss Models. New York: John Wiley & Sons Inc.

Koul, H., Susarla, V., Ryzin, J. V. (1981). Regression Analysis with Randomly Right-Censored Data. The Annals of Statistics, Vol. 9, No. 6, 1276-1288.

Lee, C. T. (1997). Applied Survival Analysis. New York: John Wiley & Sons, Inc.

McCullagh, P. and Nelder, J. (1989). Generalized Linear Models. London: Chapman & Hall. 2nd Ed.

McLachlan, G. J. and Krishnan, T. (2007). EM Algorithm and Extensions, New York: John Wiley & Sons Inc.

McLachlan. G. and Peel, P. (2000). Finite Mixture Models. New York: John Wiley & Sons, Inc.

Mendenhall W. and Sincich T. (2012). Regression Analysis. Person Education, Inc. 7th Ed., 136.

Miller, R. G. (1976). Least Squares Regression with Censored Data. Biometrika, Vol. 63, No.3, 449-464.

National Association of State Directors of Veterans Affairs (NASDVA). www.nasdva.net

North Dakota Department of Veterans Affairs. www.nd.gov/veterans/

Rencher, A.C. and Schaalje, G. B. (2008). Linear Models in Statistics. New York: John Wiley &Sons, Inc.

Schmee, J. and Hahn, G. J. (1979). A Simple Method for Regression Analysis With Censored Data. Technometrics, Vol. 21, 417-434.

Schwarz, G. (1978). Estimating the dimension of model. The Annals of Statistics, vol.6, 461-464.

Wedderburn, R. W. M. (1974). Quasi-likelihood function, generalized linear models, and the Gauss-Newton Method. Biometrika, Vol. 61. No.3, 439-447.

Weissfeld, L.A. and Schneider, H. (1990). Influence Diagnostics for the Normal Linear Model With Censored Data.Austal. J. Statist., 32(1).

Yu, L., Yu, R., Liu, L. (2009). Quasi-Likelihood for Right –Censored Data in the Generalized Linear Model. Communication in Statistics-Theory and Methods, 2187-2200.

U.S. Census Bureau, Department of Commerce, National Security and Veterans Affairs.

U.S. Census Bureau, Current Population Surveys, ASEC2000 to 2009.

U.S. Department of Health and Human services: http://aspe.hhs.gov/poverty/08poverty.shtml

APPENDIX . SEVERAL IMPORTANT R-FUNCTIONS

```
##################
### EM-algorithm ###
##################

E.step<-function(beta,  s, W2, z){
        muc <-  W2 %*% beta
        A <-  pnorm(- (z  -  muc) /  s)
        Ez <- (muc * A + (s * dnorm((z  -  muc)  /  s)))  /  A
        Ez2 <- ((muc^2 + s^2) *  A  +  s  *  (muc + z)  * dnorm((z - muc)  /  s))  /  A
        EzDz <- sum(Ez2)
return(list(Ez = Ez, EzDz = EzDz))
}
M.step <-function(beta, W1, W2,  y,  Ez,  EzDz){
        n <- dim(W1)[1] + dim(W2)[1]
       B <- t(W1)  %*%  W1  +  t(W2)  %*%  W2
       C <- t(W1)  %*%  y  + t(W2)  %*%  Ez
       beta <- solve(B)  %*%  (t(W1) %* % y +  (t(W2)  %*%  Ez))
       D <-  t(y) %*% y  +  EzDz  +  t(beta)  %*%  B  %*%  beta – 2 *  t(beta) %*%  C
        s <- drop(sqrt(D / n))
return(list(beta = beta, s = s))
}
EM <- function(W){
        q <- dim(W)[2]
        u <- W[,2]
        W11 <- W[(W[,3] == 1),]    # W1 uncensored partition
```

```
 d1 <- dim(W11)[1]
 I1 <- rep(1, d1)
 W1 <- cbind(I1, W11[, 4])
 y  <- W11[,2]
W22 <- W[(W[,3] == 0),]     # W2 censored partition
d2 <- dim(W22)[1]
I2 <- rep(1, d2)
W2 <- cbind(I2, W22[, 4])
 z <- W22[,2]
 beta <- lm(W[,2] ~ W[,4])$coef
beta.old <- rep(Inf, (q - 2))
eps <- 0.000001
s <- 0.182
iter <- 0
while(any(abs(beta  -  beta.old)  >  eps)){
            iter <- iter  +  1
            beta.old <- beta
            ZZ <- E.step(beta,  s,  W2,  z)
            Ez <- ZZ$Ez
            EzDz <- ZZ$EzDz
            MM <- M.step(beta,  W1,  W2,  y,  Ez,  EzDz)
            beta <- MM$beta
            s <- MM$s
            cat("Iteration", iter, "beta = ", beta, "s = ", s, "\n")
 }
```

```
        return(list(beta = beta, s = s, Ez = Ez, EzDz = EzDz, W1 = W1, W2 = W2, y = y))
}
                    ##############################################
                    ###  Variability Assessment based on EM-algorithm ###
                    ##############################################
Cov.par <- function(W1, W2, y, b.new, s, Ez, Ez2){
            n1 <- dim (W1)[1]
            n2 <- dim (W2)[1]
            n <- n1 + n2
            p <- length(b.new)
            dqi_par1 <- matrix(NA,  p + 1,  n1)
            dqi_par2 <- matrix(NA,  p + 1,  n2)
            for (i in 1:n1){
                 dqi_par1[1:p,i] <- (1 /s ^ 2)  *  (W1[i, ]  *  y[i]  -  W1[i, ]  %*%  t(W1[i,])
                %*%  b.new)
                dqi_par1[p+1,i] <- (- 1 / s)  +  (1 / s^3) * (y[i]^2  -  2  *  t(b.new)  %*%
                W1[i, ] * y[i]  + t(b.new)  %*%  W1[i, ]  %*%  t(W1[i, ])  %*%  b.new)
               }
            for (i in 1:n2){
                dqi_par2[1:p,i] <- (1/s^2) * (W2[i,] * Ez[i] - W2[i,] %*% t(W2[i,]) %*% b.new)
                dqi_par2[p+1,i] <- (-1/s) + (1/s^3)*(Ez2[i] - 2 * t(b.new) %*% W2[i,]* Ez[i] +
                    t(b.new) %*% W2[i,] %*% t(W2[i,]) %*% b.new)
    }
    H <- cbind(dqi_par1, dqi_par2)
```

```
        Ie <- H %*% t(H)
        Cov <- solve(Ie)
        Var <- diag(Cov)
      return(Cov)
}
CI <- function(b.new, s, sigma){
    par <- c(b.new,  s)
    p <- length(par)
    SE <- sqrt(diag(sigma))
    Lower <- rep(NA,  p)
    Upper <- rep(NA,  p)
    CI <- matrix(NA,  p,  2)
            for (i in 1:p){
                Lower[i] = par[i]  -  qnorm(0.975) *  SE[i]
                Upper[i] = par[i]  +  qnorm(0.975) *  SE[i]
                CI[i,] <- cbind(Lower[i],  Upper[i])
            }
  return(list(CI=CI))
}
p.val <- function(b.new,  s,  sigma){
       par <- c(b.new, s)
      p <- length(b.new)
      pvalue <- matrix(NA, (p + 1), 1)
```

```
        SE <- sqrt(diag(sigma))
              for (i in 1:(p+1)){
                    pvalue[i,] <- round(2  *  pnorm( -abs(par[i] / SE[i,i])), 8)
                    }
  return(list(pvalue = pvalue))
}
                                  ############################
                                  ###  Log-likelihood Function ###
                                  ############################
logL <- function (pars, y, z, W1, W2, Ez, Ez2){
        beta <- pars[-1]
        s <- pars[1]
        n1 <- length(y)
        muc <- W2  %*%  beta
        A <- pnorm(z, mean  =  muc, sd  =  s, lower.tail  =  F)
        Res <- -n1 * log(2 * pi) / 2  -  n1 * log(s^2) / 2 – 1 / (2 * s^2)  *  (t(y  - W1  %*%  beta)
              %*%  (y  -  W1  %*%  beta))  +  sum(A)
return(-Res)
}
                                  ############################
                                  ### Reconstructed R-square ###
                                  ############################
Jlin <- function (pars, yu,  Ez,  EzDz,  M1,  M2){
        l <- length(pars)-1
```

```
        beta <-pars[1:l]
        s <- pars[3]

         Res <- norm(yu - M1 %*% beta, "F")^2 + norm(Ez - M2 %*% beta, "F")^2 +
                EzDz - norm(Ez, "F")^2
 return(Res)
}
J0 <- function (pars, yu, Ez, EzDz){
       n1 <- length(yu)
       m <- length(Ez)
       beta <-pars[1:2]
       s <- pars[3]
      I1 <- rep(1,n1)
      I2 <- rep(0, n1)
      Iu <- cbind(I1, I2)
      Ic1 <- rep(1, m)
      Ic2 <- rep(0, m)
      Ic <- cbind(Ic1, Ic2)
      Res <- norm(yu - Iu %*% beta, "F")^2 + norm(Ez - Ic %*% beta, "F")^2 +
            EzDz - norm(Ez, "F")^2
return(Res)
}
OpJlin <- optim(pars, Jlin, gr = NULL, method="BFGS", y, Ez, EzDz, W1, W2,
```

```
            hessian = TRUE)
OpJ0 <- optim(pars, J0,  gr  =  NULL,  method="BFGS",  y,  Ez,  EzDz,
           hessian  =  TRUE)
R2c <- 1  -   OpJlin$value /  OpJ0$value
                                  ############################
                                  ### Buckle-James Algorithm ###
                                  ############################
E.step.bj <- function(W,  b.new){
              n <- dim(W)[1]
              flag <- 0
              yhat <- W[,4] * b.new
              residual <- W[, 2] - yhat
              workdata <- cbind(W,  residual)
             orderr <- order(residual)
             workdata <- workdata[orderr,]
             if(workdata[n,3] == 0){
                                    flag <- 1
                                   workdata[n,3] <- 1}
              km <- summary(survfit(Surv(workdata[, 5],  workdata[, 3])  ~  1))
            survival <- km$surv
            workdata1 <- workdata[workdata[, 3]  == 1, ]  # uncensored partition
            n1 <- dim(workdata1)[1]
            jumpsurvival <- c(1, survival[1:length(survival) - 1]) # shift km starting at 1
            jump <- (jumpsurvival - survival) / km$n.event
```

```
 jump <- rep(jump, km$n.event)
 survival <- rep(survival, km$n.event)
 survival <- 1  -  survival
 survival <- matrix(survival, ncol = 1)
 jump <- matrix(jump,  ncol  =  1)
 workdata1 <- cbind(workdata1,  survival)
 workdata1 <- cbind(workdata1,  jump)
 workdata2 <- workdata[workdata[, 3] == 0,]  # censored partition
 n2 <- dim(workdata2)[1]
 if(sum(workdata[, 2]) == n - 1){
 workdata2 <- matrix(workdata2,  nrow = 1)}

{if(dim(workdata2)[1]! = 0){
 zero <- matrix(0,  nrow = dim(workdata2)[1])
 workdata2 <- cbind(workdata2,  zero)
 workdata2 <- cbind(workdata2,  zero)
 workdata <- rbind(workdata1,  workdata2) }
 else{workdata <- workdata1}
          }
o <- order(workdata[, 5])
 workdata <-workdata[o, ]
workdata[, 5] <- workdata[, 5]
        for (i  in 2 : n){
```

```
                if (workdata[i,6] == 0){workdata[i, 6]<-workdata[i - 1, 6]}
            }
        denom <- rep(1, n)
        denom <- denom  - workdata[, 6]
        workdata <- cbind(workdata, denom)
        workdata <- cbind(workdata[, 1], workdata[, 2], workdata[, 3], workdata[, 4],
                    workdata[, 5], workdata[, 6], workdata[, 8], workdata[, 7])
return(list(workdata = workdata, flag = flag))
 }
weights <- function(workdata){
        n <- dim(workdata)[1]
        www <- diag(workdata[, 3])
                for (i in 1 : n){
                    if (www[i, i] == 0){
                            for (k in (i + 1) : n){
                                www[i,k] <- workdata[k, 8] / workdata[i, 7]
                                }
                        }
                }
  return(www)
}
EMbj <- function(W){
        n <- dim(W)[1]
```

```
eps <- 0.000001
iter <- 0
b.new <- 1.9
b.old <- Inf
b1 <- vector()
m <- 0
                while(abs(b.new - b.old) > eps && m < 10){
                 iter <- iter + 1
                b.old <- b.new
                b1<-c(b1, b.old[1])
                level.old<-levels(as.factor(b1))
                length.old<-length(level.old)
               ee <- E.step.bj(W, b.new)
               Wdata <- ee$workdata
               ff <- ee$flag
              Wgt <- weights(Wdata)
              X <- Wdata[, 4] - mean(Wdata[, 4])
              A <- solve(t(X) %*% X) %*% t(X)
              Ystar <- Wdata[, 4] * b.new + Wgt %*% Wdata[, 5]
              b.new <- A %*% Ystar
              b1 <- c(b1, b.new[1])
              level.new <- levels(as.factor(b1))
              length.new <- length(level.new)
```

```
                if(length.new == length.old) m <- m + 1
                else m <- 0
                 if (ff == 1){W[n, 3] <- 0}
                 cat("Iteration",  iter,  "b.new=",  b.new,  "\n")
              }
  alpha <- mean(Ystar)  -  b.new  *  mean(Wdata[, 4])
  return(list(alpha = alpha,  b.new = b.new))
}
              ##########################################################
              ### Generating Normal Data Used in Chapter 8.1 Simulations###
              ##########################################################
Ran <- function(n, l){
        library(splines)
        library(survival)
        library(rms)
        library(Hmisc)
        b0 <- 1
        b1 <- 2
        i <-1 : n
        x <- i  /  n
        e <- rnorm(n, 0,  0.182)
        y <- b0  +  b1  *  x  +  e
        k <- l * n
        WC <- matrix(NA,  k,  4)
```

```
cens <- rep(0, k)
        for (j in 1 : k){
                repeat{
                id <- round(runif(1,  1,  n))
                c <- runif(1,  1,  3)
                 if((y[id]  >  c)  &  (!any(id == cens))) break
        }
        WC[j,1] <- id
        WC[j,2] <- c
        WC[j,3] <- 0
        WC[j,4] <- id / n
        cens[j] <- id
}
        WW <- matrix(NA,  n,  4)
                for (i in 1 : n){
                 if(any(I == cens)){
                 WW[i,]<- WC[WC[, 1] == i, ]
                        }else{
                                WW[i, 1] <- i
                                WW[i, 2] <- y[i]
                                WW[i, 3]<- 1
                                WW[i, 4]<- I / n
                         }
```

```
                }
return(WW)
}
                ########################################
                ### Outlier Detection Chapter 8.3 Simulations###
                ########################################
OutlierU <- function(M, M1, M2, yu, zc, ut){
            alpha <- 0.05
            n <- dim(M)[1]
            l <- dim(M1)[1]
           p <- dim(M)[2]
          Out <- matrix(NA, l, 2)
          Res_unc <- Udelete(M, M1, M2, yu, zc, ut, beta)
         Num <- Res_unc$e
         si <- Res_unc$d_sdu
         h <- diag(M1 %*% solve(t(M1) %*% M1) %*% t(M1))
         Den <-si/sqrt(1 - h)
         t <- abs(Num / Den)
         CritVal <- qt(1-alpha/(2*n), df = n - p)
              for (i in l : l){
                      if(abs(t[i]) >  CritVal){(Out[i, 1] <- t[i])& (Out[i, 2] <- 1)}
                        else
                      {(Out[i,1] <- t[i]) & (Out[i, 2] = 0)}
                  }
return(list(h = h, Num = Num, Den = Den, Out = Out))
```

```
}

OutlierC <- function(M, M1, M2, yu, zc, ut, AA, d_sd, lc){
              alpha <- 0.05
              l <- dim(M1)[1]
              p <- dim(M)[2]
             n <- dim(M)[1]
             k <- n - l
             Out <- matrix(NA, k, p)
              h <- rep(NA, k)
              Res_cen <- Cdelete(M, M1, M2, yu, zc, ut, AA)
             Num <- Res_cen$e
             SS <- Res_cen$d_sd
                        for (i in 1 : k){
                                    X <- rbind(M1, M2[i, ])
                                     H <- diag(X %*% solve(t(X) %*% X) %*% t(X))
                                     h[i] <-H[l + 1]
                           }
              Den <- SS / sqrt(1 - h)
              t <- abs(Num / Den)
              CritVal <- qt(1 - alpha / (2*n), df = n - p)
             a <- dim(M2)[1]
                         for (i in 1 : a){
                             if(abs(t[i]) > CritVal){(Out[i, 1] <- t[i])& (Out[i, 2]<- 1)}
                                 else
```

```
                    {(Out[i,1] <- t[i]) & (Out[i, 2] = 0)}
                }
return(list(h = h, Num = Num, Den = Den, Out = Out, t = t))
}
                ######################################
                ### One-step Deletion Used in Chapter 7.3###
                ######################################
Udelete <- function(M, M1, M2, yu, zc, ut){
        q <- dim(M)[2]
        m <- length(yu)
        dbeta <- matrix(NA, m, q)
        dbeta_def <- matrix(NA, m, q)
        dr <- matrix(NA, m, q)
        do <- rep(NA, m)
        deltau_new <- matrix(NA, m, q)
                for(i in 1 : m) {
                        ystar <- yu
                        W1 <- M1[-i, ]
                        W2 <- M2
                        y <- yu[-i]
                        z <- zc
                        W <- rbind(M1[-i, ], M2)
                        u <- c(yu[-i], zc)
                        dr[i, ] <- M1[i, ]
                        do[i] <- yu[i]
```

```
b.new <- lm(u ~ W[,-1])$coef
b.old <- rep(Inf, q)
eps <- 0.0000001
s <- 1
iter <- 0
    while(any(abs(b.new - b.old) > eps)){
        iter <- iter + 1
        #E-step#
        muc <- W2 %*% b.new
        A <- pnorm(-(z - muc) / s)
        Ez <- (muc * A + (s * dnorm((z - muc) / s))) / A
        Ez2 <- ((muc^2 + s^2)* A + s * (muc + z) *
            dnorm((z - muc) / s)) / A
        EzDz <- sum(Ez2)
        #M-step#
        b.old <- b.new
        B <- t(W1) %*% W1 + t(W2) %*% W2
        C <- t(W1) %*% y +t(W2) %*% Ez
        b.new <- solve(B) %*% (t(W1)%*%y + (t(W2)
            %*% Ez))
        D <- t(y)%*% y + EzDz + t(b.new) %*% B %*%
            b.new - 2* t(b.new)%*% C
        s <- drop(sqrt(D/n))
#cat("Iteration", iter, "b.new=", b.new, "s=", s, "\n")
 }
```

```
                dbeta[i,] <- b.new
                dbeta_def[i,] <- (t(beta) - b.new) / solve(t(M) %*% M) %*% dr[i, ]
                yhat_star <- c(y, Ez)
                Num1 <- solve(t(M) %*% M)
                I <- diag(q)
                Den <- as.numeric(1 - t(dr[i, ]) %*% solve(t(M) %*% M) %*%  dr[i, ])
                Num2 <- as.numeric(1 - t(dr[i, ]) %*% solve(t(M) %*% M) %*% dr[i, ])
                Num3 <- Num2 * I
                Num4 <- dr[i, ]%*%t(dr[i,])%*%solve(t(M)%*%M)
                Num5 <- t(W) %*% (u - yhat_star)
                Num6 <- dr[i, ] %*% (yu[i] - dr[i,]%*% t(beta))
                deltau_new[i, ] <- abs(Num1 %*% ((Num3 + Num4) %*%  Num5 + Num6)/Den)
                dbeta_def[i,] <- (deltau_new[i,]) / solve(t(M)%*%M)%*%dr[i,]
    }
  return(list(dbeta_def =  dbeta_def))
}
Cdelete <- function(M,  M1,  M2,  yu,  zc,  ut,  AA){
              q <- dim(M)[2]
              m <- length(yu)
              n <- dim(M)[1]
              k <- n - m
             dbeta <- matrix(NA,  k,  q)
            dbeta_def <- matrix(NA,  k,  q)
            d_sd <- rep(NA,  k)
           drc <- matrix(NA,  k,  q)
```

```
doc <- rep(NA,  k)
z_d <- rep(NA,  k)
EZ <- matrix(NA,  k,  k - 1)
deltac_new <- matrix(NA,  k,  q)
 h <- rep(NA,  k)
        for(i in 1 : k) {
                W2 <- M2[-i,]
                W1 <- M1
                z <- zc[-i]
                y <- yu
                W <- rbind(M1, M2[-i, ])
                u <- c(yu, zc[-i])
                drc[i,] <- M2[i, ]
                doc[i] <- zc[i]
               b.new <- lm(u ~ W[,-1])$coef
                b.old <- rep(Inf,  q)
                eps <- 0.0000001
                s <- 1
                iter <- 0
                        while(any(abs(b.new  -  b.old)  > eps)){
                        iter <- iter  + 1
                        #E-step#
                        muc <- W2  %*%  b.new
                        A <- pnorm(- (z - muc)  /  s)
                        Ez <- (muc * A + (s  * dnorm((z -  muc)  /  s)))  /  A
```

```
                    Ez2 <- ((muc^2 + s^2) * A  + s * (muc + z)  * dnorm((z - muc)
                          / s)) / A
                    EzDz <- sum(Ez2)
                    #M-step#
                    b.old <- b.new
                    B <- t(W1) %*%  W1 + t(W2)  %*%  W2
                    C <- t(W1) %*% y +t(W2) %*% Ez
                    b.new <- solve(B) %*% (t(W1) %*% y  +  (t(W2)  %*%  Ez))
                    D <- t(y)  %*%  y  + EzDz  +  t(b.new)  %*%  B  %*%  b.new
                          - 2 * t(b.new) %*%  C
                     s <- drop(sqrt(D / n))
  }
  EZ[i,] <- Ez
  sigma  <- s
  h_u <- (doc[i]  -  drc[i,]  %*%  t(beta)) / sigma
  A <- dnorm(h_u)
  B <- 1 - pnorm(h_u)
   h[i] <- A / B
  dbeta[i,] <- b.new
 yhat_star <- c(y,Ez)
  Num1 <- solve(t(M) %*% M)
  I <- diag(q)
  Den <- 1-t(drc[i, ]) %*% solve(t(M) %*% M) %*% drc[i, ]
  Num2 <- 1-t(drc[i, ]) %*% solve(t(M) %*% M) %*% drc[i, ]
  Num3 <- as.numeric(Num2) * I
```

```
                Num4 <- drc[i,] %*% t(drc[i, ]) %*% solve(t(M) %*% M)
                Num5 <- t(W) %*% (u - yhat_star)
                Num6 <- drc[i,] %*% (EZ[i] - drc[i,] %*%  t(beta))
               deltac_new[i,] <- abs(Num1 %*% ((Num3 + Num4) %*%  Num5 +
                                Num6) / as.numeric(Den))
               dbeta_def[i,] <- (delta_new[i, ]  / solve(t(M) %* %M) %*% drc[i, ]

    }
  return(list(beta =  beta,  dbeta  =  dbeta,  dbeta_def  =  dbeta_def))
}
                        ########################################
                        ### Simulation Function used in Chapter 8.1 ###
                        ########################################
sim <- function(N,  n,  l){
        par.bj <- matrix(NA,  N,  2)
        par.p <- matrix(NA,  2,  N)
        par.s <- rep(NA,  N)
                        for (i in 1 : N){
                        W <- Ran(n, l)
                        e.bj <- EMbj(W)
                        e.p <- EM(W)
                        par.p[,i] <-e.p$beta
                        par.s[i] <-e.p$s
                        par.bj[i, 1] <- e.bj$alpha
                        par.bj[i, 2] <- e.bj$b.new
```

```
                    }
        mean0_p <- mean(par.p[1, ])
        mean1_p <- mean(par.p[2, ])
       MSE0_p <-  mean((par.p[1, ] - 1)^2)
        MSE1_p <-  mean((par.p[2, ] - 2)^2)
        mean0_bj <- mean(par.bj[, 1])
       mean1_bj <- mean(par.bj[, 2])
       MSE0_bj <- mean((par.bj[, 1] - 1)^2)
       MSE1_bj <- mean((par.bj[, 2] - 2)^2)
 return(list(mean0_p =  mean0_p, mean1_p  =  mean1_p,  MSE0_p  =  MSE0_p,
 MSE1_p = MSE1_p,  mean0_bj = mean0_bj,  mean1_bj = mean1_bj,
MSE0_bj = MSE0_bj, MSE1_bj = MSE1_bj,  par.p=  par.p,  par.bj = par.bj))
}
```