Efficient Algorithms for Computing the Non- and Semi-Parametric Maximum Likelihood Estimates of Panel Count Data

Gang Cheng and Ying Zhang
Department of Biostatistics, University of Iowa
Iowa City, IA 52242, USA

Liqian Lu
School of Mathematics
Fudan University, P.R. China

April 23, 2010

Abstract

Non-parametric and semi-parametric analysis of panel count data have recently been an active research topic in statistical literature. Maximum likelihood method based on non-homogeneous Poisson process has been proved an efficient inference procedure for such analysis. However, computing the non- and semi-parametric maximum likelihood estimates (MLE) can be very intensive numerically. In this manuscript, we develop an efficient numerical algorithm stemmed from the Newton-Raphson method to compute the non- and semi-parametric MLE for panel count data. Simulation studies are carried out to demonstrate the numerical efficiency of the proposed algorithm compared to the existing methods in the literature.

Some key words: Quadratic programming; Interval censored data; Isotonic Regression; Iterative convex minorant algorithm; Monte-Carlo.

1. Introduction

Analysis of panel count data is a common practice in clinical trials, econometrics, system reliability, and social demographic studies. Panel count data are special type of longitudinal count data in which the underlying counting process or covariates' effects on the counting process is often the study of interest. For panel count data, the exact times at which recurrent events occur are not observable but the numbers of events that happen between consecutive observation times are recorded. The number of observations and the observation times are allowed to vary from subject to subject. Such data can be found in many applications, particularly in biomedical follow-up studies, for example, the National Cooperative Gallstone Study (Thall and Lachin, 1988) and the superficial bladder tumor clinical trial (Byar, et. al., 1980).

Recently, non-parametric and semi-parametric analyses of panel count data have drawn considerable attention in statistical literature. Sun and Kalfleisch (1995), Wellner and Zhang (2000), Zhang and Jamshidian (2003), Lu et. al. (2007) and Hu et. al. (2009a, 2009b) developed non-parametric methods for estimating the mean function of underlying counting process. Sun and Fang (2003), Zhang (2006) and Balakrishnan and Zhao (2009) studied some non-parametric testing procedures for comparing the mean function of underlying counting processes. Sun and Wei (2000), Zhang (2002), Hu et. al. (2003), Huang et. al. (2006), Wellner and Zhang (2007), Sun et. al. (2007), He et. al. (2009) and Lu et. al. (2009) conducted various semi-parametric analyses for panel count data using proportional mean model. In particular, Wellner and Zhang (2000, 2007) studied the non-parametric and semi-parametric maximum likelihood methods for panel count data, respectively, based on non-homogeneous Poisson process, and demonstrated the robustness of the methods against the underlying counting process. Moreover, they developed two robust inference procedures for the regression parameter in the proportional mean model and showed that the maximum likelihood inference procedure based on Poisson process is efficient if the underlying counting process is indeed Poisson. However, computing the maximum likelihood estimate is a challenging task. Wellner and Zhang (2000) used the iterative convex minorant algorithm (ICM) developed by Jongbloed (1998) to compute the non-parametric maximum likelihood estimate (NPMLE) of the mean function. The ICM algorithm can be time-consuming for the NPMLE of panel count data when the sample size is large and the observation times are very different among study subjects. Hu et. al. (2009a) elegantly studied panel count data in the framework of missing value problem in which each counting process is assumed observed at pre-specified time points but subjects to right censoring, hence the NPMLE for panel count data can be

computed using an easy-to-implement EM algorithm. Although algorithmically convenient, the EM does not have the advantage in quick convergence compared to the ICM algorithm. For the semi-parametric maximum likelihood estimate (SPMLE) of panel count data with the proportional mean model, Wellner and Zhang (2007) utilized the extended ICM algorithm developed by Pan (1999) to compute the maximum likelihood estimates of regression parameter and baseline mean function jointly. However, the algorithm requires enormous computing effort to achieve the convergence, even for data with moderate sample size.

In this manuscript, we propose some Newton-Raphson based algorithms to compute both the NPMLE and SPMLE of panel count data. The proposed algorithms can greatly reduce the numerical effort in computing the NPMLE and SPMLE of panel count data compared to the existing ICM-type of algorithms and therefore are encouraged to use in maximum likelihood analysis of panel count data.

The rest of article is organized as follows. Section 2 briefly describes the notations in maximum likelihood analysis of panel count data and formally formulate the NPMLE and SPMLE; Section 3 describes the proposed Newton-Raphson based algorithms and connects them to the ICM and extended ICM algorithms, respectively; Section 4 conducts some simulation studies to demonstrate the numerical advantage of the proposed methods over the ICM methods; Section 5 concludes the paper with some remarks.

2. NPMLE and SPMLE of Panel Count Data

Suppose that $\mathbb{N} = \{N(t) : t \geq 0\}$ is a univariate counting process and is observed at K random times $0 \equiv T_{K,0} < T_{K,1} < \cdots < T_{K,K}$, where K, the number of observations, is allowed to be random as well.

For the NPMLE of panel count data, we denote the observed information for each subject as

$$D = (K, T_{K,1}, \cdots, T_{K,K}, N(T_{K,1}), \cdots, N(T_{K,K})) \equiv (K, \underline{T}_K, \underline{N}_K).$$

The panel count data consist of n i.i.d. copies of D, $D_1, \ldots, D_n$. We are interested in estimating the expected number of events at any time t, $\Lambda(t) = E(N(t))$. Using the Poisson process model, the log-likelihood for the observed data can be derived as

$$l(\Lambda(t); Data) = \sum_{i=1}^{n} \sum_{j=1}^{K_i} \{(N(T_{K_i,j}) - N(T_{K_i,j-1})) \log(\Lambda(T_{K_i,j}) - \Lambda(T_{K_i,j-1}))$$

$$-(\Lambda(T_{K_i,j}) - \Lambda(T_{K_i,j-1}))\}$$

$$= \sum_{i=1}^{n}\sum_{j=1}^{K_i}(N(T_{K_i,j}) - N(T_{K_i,j-1}))\log(\Lambda(T_{K_i,j}) - \Lambda(T_{K_i,j-1})) - \sum_{i=1}^{n}\Lambda(T_{K_i,K_i})$$

assuming the distribution of $(K, \underline{T}_K)$ is independent of $N(t)$ and is non-informative to the parameter $\Lambda(t)$. Let $0 \equiv s_0 < s_1 < \cdots < s_m < \infty$ be the ordered distinct observation time points for the set of $\{T_{K_i,0} \equiv 0 : i = 1, \ldots, n\} \cup \{T_{K_i,j} : i = 1, \ldots, n; j = 1, \ldots, K_i\}$. Then the log-likelihood for panel count data can be simplified to

$$l(\Lambda; Data) = \sum_{l'=1}^{m-1}\sum_{l=l'+1}^{m} A_{l,l'}\log(\Lambda_l - \Lambda_{l'}) - \sum_{l=1}^{m} B_l\Lambda_l, \tag{2.1}$$

where

$$A_{l,l'} = \sum_{i=1}^{n}\sum_{j=1}^{K_i}(N(T_{K_i,j}) - N(T_{K_i,j-1}))I_{[T_{K_i,j}=s_l, T_{K_i,j-1}=s_{l'}]}, B_l = \sum_{i=1}^{n} I_{[T_{K_i,K_i}=s_l]},$$

and

$$\Lambda = (\Lambda_1, \ldots, \Lambda_m) \equiv (\Lambda(s_1), \ldots, \Lambda(s_m)).$$

Because the mean function $\Lambda(t)$ is monotone non-decreasing, we naturally require the NPMLE to be monotone non-decreasing as well. Define the cones $\mathcal{C}$ and $\mathcal{C}_+$, respectively, by

$$\mathcal{C} = \{x \in R^m : x_1 \le x_2 \le \cdots \le x_m\} \text{ and } \mathcal{C}_+ = \{x \in \mathcal{C} : x_1 \ge 0\}.$$

The NPMLE of $\Lambda(t)$ is conventionally defined as the step function with jumps only possibly occurred at s_l for $l = 1, 2, \cdots, m$ that maximized the log-likelihood (2.1) (Wellner and Zhang, 2000), that is

$$\hat{\Lambda} = arg\max_{x \in \mathcal{C}_+} l(x; Data).$$

For the SPMLE of panel count data, the observed data consist of $D = (K, \underline{T}_K, \underline{N}_K, Z)$, where $Z \in R^d$ is a vector of covariates that are also available at baseline whose effects on the counting process may be the primary study of interest in applications. The proportional mean model

$$\Lambda(t|Z) \equiv E(N(t)|Z) = \Lambda(t)e^{\beta^T Z}, \tag{2.2}$$

as the most popular model in analysis of counting process, has been proposed in the literature by, for example, Lawless and Nadeau (1995), Sun and Wei (2000), Lin, et.al. (2000), and

Wellner and Zhang (2007). With n i.i.d. copies of D, $D_1, \ldots, D_n$, Wellner and Zhang (2007) derived the log-likelihood for panel count data using the Poisson process model under the proportional mean model (2.2)

$$\begin{aligned} l(\beta, \Lambda(t); Data) &= \sum_{i=1}^{n} \sum_{j=1}^{K_i} \{(N(T_{K_i,j}) - N(T_{K_i,j-1})) \log(\Lambda(T_{K_i,j}) - \Lambda(T_{K_i,j-1})) \\ &\quad + (N(T_{K_i,j}) - N(T_{K_i,j-1}))\beta^T Z_i - e^{\beta^T Z_i}(\Lambda(T_{K_i,j}) - \Lambda(T_{K_i,j-1}))\Big\} \\ &= \sum_{i=1}^{n} \sum_{j=1}^{K_i} \{(N(T_{K_i,j}) - N(T_{K_i,j-1})) \log(\Lambda(T_{K_i,j}) - \Lambda(T_{K_i,j-1}))\} \\ &\quad + \sum_{i=1}^{n} \left\{ N(T_{K_i,K_i})\beta^T Z_i - e^{\beta^T Z_i} \Lambda(T_{K_i,K_i}) \right\}. \end{aligned}$$

This log-likelihood can be similarly simplified to

$$l(\beta, \Lambda; Data) = \sum_{l'=1}^{m-1} \sum_{l=l'+1}^{m} A_{l,l'} \log(\Lambda_l - \Lambda_{l'}) - \sum_{l=1}^{m} B_l \Lambda_l + \sum_{i=1}^{n} N(T_{K_i,K_i})\beta^T Z_i, \tag{2.3}$$

using the same way as for the NPMLE with B_l being modified to $B_l = \sum_{i=1}^{n} e^{\beta^T Z_i} I_{[T_{K_i,K_i} = s_l]}$. Denote $\Theta = R^d \times \mathcal{C}_+$, then the SPMLE of panel count data, $\hat{\theta} = (\hat{\beta}, \hat{\Lambda})$, is the element inside Θ that maximizes the log-likelihood (2.3), that is

$$\hat{\theta} \equiv (\hat{\beta}, \hat{\Lambda}) = arg \max_{\theta \in \Theta} l(\theta; Data).$$

3. Projected Newton-Raphson Algorithms for Computing the NPMLE and SPMLE

The log-likelihood given in (2.1) $l(x)$ (abbreviation for $l(x; Data)$) is a smooth concave function over the cone $\mathcal{C}_+$. Hence the ICM algorithm developed by Jongbloed (1998) for non-parametric estimation is naturally applied by Wellner and Zhang (2000) to compute the NPMLE of panel count data. For the semi-parametric regression analysis, the log-likelihood given in (2.3) $l(\beta, x)$ (abbreviation for $l(\theta; Data)$) is not globally concave function over the parameter space Θ. However, for a fixed β, $l(\beta, x)$ is a smooth concave function over the cone $\mathcal{C}_+$ and for a fixed x, $l(\beta, x)$ is a concave function over R^d. Because of this property,

Wellner and Zhang (2007) applied the extended ICM algorithm, developed by Pan (1999) for computing the SPMLE of interval-censored data, to compute the SPMLE of panel count data. This algorithm is a doubly iterative procedure: (1) given the current estimate of β, $\beta^{(old)}$, update the estimate of Λ by optimizing $l(\beta^{(old)}, x)$ using the ICM algorithm; (2) for newly updated $\Lambda^{(new)}$, update the estimate of β, $\beta^{(new)}$ by optimizing $l(\beta, \Lambda^{(new)})$ using the Newton-Raphson algorithm; (3) repeat the circle (1)-(2) until convergence. In our experience, the ICM algorithm, although globally converging, is generally inefficient as it converges in a super linear rate. It is very time-consuming, in particular, when applied to compute the SPMLE of panel count data as evidenced in Wellner and Zhang (2007). Therefore it will be practically useful to develop more efficient algorithms to compute both the NPMLE and SPMLE for analysis of panel count data.

We start with the generalized gradient algorithm for maximizing an objective function $l(x)$, that is

$$x^{(k+1)} = B(x^{(k)}) \equiv x^{(k)} + H(x^{(k)})\nabla_x l(x^{(k)}), \quad k = 0, 1, \cdots,$$

where $H(x)$ (the weight matrix for the gradient) is any positive definite matrix. If the objective function $l(x)$ is globally concave, then the choice of $H(x) = -\left\{\nabla_x^2 l(x)\right\}^{-1}$ leads to the widely used Newton-Raphson algorithm in statistics that is known to have a quadratic rate of convergence. However the quick convergence of Newton-Raphson algorithm is only theoretically justified for unconstrained optimization problems and it is not demonstrated in constrained problems. In addition, the Newton-Raphson update $x^{(k+1)}$ does not necessarily fall in the feasible region defined by the constrained optimization problem, say $x \in \mathcal{X}$. In this paper, we propose to project the Newton-Raphson update to the feasible region $\mathcal{X}$ during the iterations by

$$\tilde{x}^{(k+1)} = arg\min_{x \in \mathcal{X}} (x - x^{(k+1)})^T W\left(x^{(k)}\right)(x - x^{(k+1)}),$$

with a positive definite matrix $W(x)$. It is apparently that if the Newton-Raphson update happens to be inside $\mathcal{X}$, its projection to $\mathcal{X}$ is simply itself. Combining the two steps and rewriting $\tilde{x}^{(k+1)}$ by $x^{(k+1)}$, the projected Newton-Raphson algorithm (projected-NR) can be expressed as

$$x^{(k+1)} = arg\min_{x \in \mathcal{X}} \left(x - B(x^{(k)})\right)^T W\left(x^{(k)}\right)\left(x - B(x^{(k)})\right). \tag{3.1}$$

It has been showed by Jongbloed (1998) that if the feasible region $\mathcal{X} = \mathcal{C}$ or $\mathcal{C}_+$ and $W(x^{(k)}) = H^{-1}(x^{(k)})$ is chosen and happens to be a positive definite matrix, the algorithm (3.1) with a proper line search generates a sequence converging to the true optimizer. The ICM algorithm developed by Jongbloed (1998) selects $H(x) = -\left\{\text{diag}\left\{\nabla_x^2 l(x)\right\}\right\}^{-1}$ by taking the advantage

that the projection can be easily implemented by finding the left derivative of a cumulative sum diagram in the paradigm of isotonic regression (Robertson et. al., 1988).

We note that for the NPMLE and SPMLE of panel count data described in Section 2, the cone $\mathcal{C}_+$ can be equivalently expressed by the linear inequality constraints $Ax \geq 0$ with

$$A = \begin{bmatrix} 1 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ -1 & 1 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & -1 & 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & 0 & \cdots & -1 & 1 & \cdots & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \cdots & \cdots & -1 & 1 \end{bmatrix}_{m \times m}.$$

Then projecting the Newton-Raphson update to the cone $\mathcal{C}_+$ with the weight matrix given by the negative Hessian matrix can be also easily implemented using the duel method for strictly convex quadratic programming (QP) developed by Goldfarb and Idnani (1983) which has been coded in R package for public use. Taking the advantage of quick convergence of Newton-Raphson method and the easily accessed duel algorithm motivates the adoption of the projected-NR algorithm to compute the NPMLE and SPMLE of panel count data.

For the NPMLE, we choose $H(x) = -\left\{\nabla_x^2 l(x)\right\}^{-1}$. For the SPMLE, we choose

$$H(\beta, x) = \begin{bmatrix} H_1(\beta, x) & 0 \\ 0 & H_2(\beta, x) \end{bmatrix},$$

where $H_1(\beta, x) = -\left\{\nabla_\beta^2 l(\beta, x)\right\}^{-1}$ and $H_2(\beta, x) = -\left\{\nabla_x^2 l(\beta, x)\right\}^{-1}$. Then the projected-NR algorithm (3.1) can be expressed by

$$\begin{pmatrix} \beta^{(k+1)} \\ x^{(k+1)} \end{pmatrix} = \begin{pmatrix} B_1(\beta^{(k)}, x^{(k)}) \\ arg\max_{x \in \mathcal{C}_+} \left(x - B_2(\beta^{(k)}, x^{(k)})\right)^T H_2^{-1}(\beta^{(k)}, x^{(k)}) \left(x - B_2(\beta^{(k)}, x^{(k)})\right) \end{pmatrix}, \quad (3.2)$$

where

$$B_1(\beta^{(k)}, x^{(k)}) = \beta^{(k)} + H_1(\beta^{(k)}, x^{(k)}) \nabla_\beta l(\beta^{(k)}, x^{(k)})$$

and

$$B_2(\beta^{(k)}, x^{(k)}) = x^{(k)} + H_2(\beta^{(k)}, x^{(k)}) \nabla_x l(\beta^{(k)}, x^{(k)}).$$

For the β-part of (3.2), it is simply the Newton-Raphson iterate for given x, because there is no constraint for the regression parameter in the proportional mean model (2.2).

It is noted that the success of both the ICM and projected-NR algorithms require the weight matrix $H(x)$ to be strictly positive definite. Directly using the Hessian matrix of the

log-likelihood will be problematic, because the Hessian matrices that result in the weight matrices $H(x)$ (for the NPMLE) and $H_2(\beta, x)$ (for the SPMLE), may not be strictly positive definite. From the likelihood structures of (2.1) and (2.3), it can be easily seen that if (i) $A_{j,l} = 0$ for $l = 1, 2, \cdots, j-1$, letting $\Lambda_j = \Lambda_{j-1}$ will increase the log-likelihood (ii) $A_{l,j} = 0$ for $l = j+1, \ldots, m$ and $B_j = 0$, letting $\Lambda_j = \Lambda_{j+1}$ will increase the log-likelihood. Therefore, we recommend to perform the dimension reduction procedure as described by (i) and (ii) before implementing the aforementioned algorithms. After reducing the size of $\mathcal{C}_+$, the ICM and extended ICM algorithms for computing the NPMLE and the SPMLE, respectively, will generate a stable sequence of x's that makes the diagonal matrices of $H(x) = -\{\nabla_x^2 l(x)\}^{-1}$ (for the NPMLE) and $H_2(\beta, x) = -\{\nabla_x^2 l(\beta, x)\}^{-1}$ (for the SPMLE) strictly positive definite during iterations. Therefore there is no numerical trouble in implementing the ICM and extended ICM algorithms. However, when the full matrices of $H(x)$ and $H_2(\beta, x)$ are used for the projected-NR algorithm, the strictly positive definiteness of $-\nabla_x^2 l(x)$ or $-\nabla_x^2 l(\beta, x)$ is not guaranteed during iterations due to the fact that two adjacently distinct Λ values are too close resulting in machine overflow in some elements of $\nabla_x^2 l(x)$ and $\nabla_x^2 l(\beta, x)$. It will evidently cause either the largest eigenvalue of $-\nabla_x^2 l(x)$ or $-\nabla_x^2 l(\beta, x)$ overflow, or the smallest eigenvalue of $-\nabla_x^2 l(x)$ or $-\nabla_x^2 l(\beta, x)$ less than zero during iterations that will lead to the breakdown of the projected-NR algorithm. If that happens, we recommend to replace the full matrix of $H(x)$ or $H_2(\beta, x)$ by $-\{\text{diag}\{\nabla_x^2 l(x)\}\}^{-1}$ or $-\{\text{diag}\{\nabla_x^2 l(\beta, x)\}\}^{-1}$, respectively. For example, suppose at the kth iteration for computing the SPMLE, it is found that the largest eigenvalue of $-\nabla_x^2 l\left(\beta^{(k-1)}, x^{(k-1)}\right) > 10^{10}$ or the smallest eigenvalue of $-\nabla_x^2 l\left(\beta^{(k-1)}, x^{(k-1)}\right) < 0$, the matrix $H_2(\beta^{(k-1)}, x^{(k-1)})$ will be replaced by $-\left\{\text{diag}\left\{\nabla_x^2 l(\beta^{(k-2)}, x^{(k-2)})\right\}\right\}^{-1}$ for the kth iteration of (3.2). In our extensive numerical experiments, this modification does not need to be carried out very frequently. For the NPMLE, there is no more than 2% of times that such modification is needed and for the SPMLE, the chance of requiring such modification is between 15% and 21%. When such modification was required for the projected-NR, it only needed once during iterations. With this modification built in our computation, the convergence of the projected-NR algorithm is achieved 100% of times. In the following, we describe fully the projected-NR algorithm for computing the NPMLE and SPMLE, respectively.

The Projected-NP algorithm for the NPMLE

1. Perform the dimension reduction for the feasible region $\mathcal{C}_+$ for the log-likelihood (2.1).

2. Choose an initial value $\Lambda^{(0)}$ (this can be quite arbitrary as long as two adjacent $\Lambda^{(0)}$ are not too close).

3. At the kth iteration ($k = 1, 2, \ldots$), calculate the smallest and largest eigenvalues of $-\nabla^2_\Lambda l(\Lambda^{k-1})$, $\lambda_1^{(k-1)}$ and $\lambda_m^{(k-1)}$, respectively. If neither $\lambda_1^{(k-1)} < 0$ nor $\lambda_m^{(k-1)} > 10^{10}$, choose $H(\Lambda^{(k-1)}) = -\left\{\nabla^2_\Lambda l(\Lambda^{(k-1)})\right\}^{-1}$; else choose $H(\Lambda^{(k-1)}) = -\left\{\text{diag}\left\{\nabla^2_\Lambda l(\Lambda^{(k-2)})\right\}\right\}^{-1}$, then do

   - $B(\Lambda^{(k-1)}) = \Lambda^{(k-1)} + H(\Lambda^{(k-1)})\nabla_\Lambda l(\Lambda^{(k-1)})$
   - $\tilde{\Lambda}^{(k)} = arg\min_{\Lambda \in \mathcal{C}_+} \left(\Lambda - B(\Lambda^{(k-1)})\right)^T H^{-1}(\Lambda^{(k-1)}) \left(\Lambda - B(\Lambda^{(k-1)})\right)$ using the Goldfarb-Idnani's duel method that has been implemented in R package.

4. Perform the line search procedure described in Jongbloed (1998) on the segment

$$seg(\Lambda^{(k-1)}, \tilde{\Lambda}^{(k)}) = \left\{\Lambda^{(k-1)} + \lambda\left(\tilde{\Lambda}^{(k)} - \Lambda^{(k-1)}\right) : \lambda \in [0,1]\right\}$$

   that is

$$\Lambda^{(k)} = \begin{cases} \tilde{\Lambda}^{(k)}, \quad \text{if } \; l(\tilde{\Lambda}^{(k)}) > l(\Lambda^{(k-1)}) + \epsilon\left\{\nabla_\Lambda l(\Lambda^{(k-1)})\right\}^T(\tilde{\Lambda}^{(k)} - \Lambda^{(k-1)}) \\ \left\{y \in seg(\Lambda^{(k-1)}, \tilde{\Lambda}^{(k)}) : (1-\epsilon)\left\{\nabla_\Lambda l(\Lambda^{(k-1)})\right\}^T(y - \Lambda^{(k-1)}) \geq l(y) - l(\Lambda^{(k-1)})\right. \\ \qquad\qquad \left. \geq \epsilon\left\{\nabla_\Lambda l(\Lambda^{(k-1)})\right\}^T(y - \Lambda^{(k-1)})\right\} \text{ elsewhere} \end{cases} \quad (3.3)$$

   for a $\epsilon \in (0, 1/2)$ to guarantee the value of the log-likelihood increase sufficiently,

5. Check if

$$\|\Lambda^{(k)} - \Lambda^{(k-1)}\|_\infty = \max_{1 \leq i \leq m} \left|\Lambda_i^{(k)} - \Lambda_i^{(k-1)}\right| \leq \eta$$

   for some small $\eta > 0$. If it is true, stop the iteration; otherwise let $k = k+1$ and repeat Steps 3-4.

The Projected-NR algorithm for the SPMLE

1. Perform the dimension reduction for the feasible region $\mathcal{C}_+$ for the log-likelihood (2.3).

2. Choose initial values, $\beta^{(0)}$ (it can start with 0) and $\Lambda^{(0)}$ (this can be quite arbitrary as long as two adjacent $\Lambda^{(0)}$ are not too close).

3. At the kth iteration ($k = 1, 2, \ldots$), calculate the smallest and largest eigenvalues of $-\nabla^2_\Lambda l(\beta^{(k-1)}, \Lambda^{(k-1)})$, $\lambda_1^{(k-1)}$ and $\lambda_m^{(k-1)}$, respectively. If neither $\lambda_1^{(k-1)} < 0$ nor $\lambda_m^{(k-1)} > 10^{10}$, choose $H_{2,k-1} = -\left\{\nabla^2_\Lambda l(\beta^{(k-1)}, \Lambda^{(k-1)})\right\}^{-1}$; else choose $H_{2,k-1} = -\left\{\text{diag}\left\{\nabla^2_\Lambda l(\beta^{(k-2)}, \Lambda^{(k-2)})\right\}\right\}^{-1}$, then do

   - $\beta^{(k)} = B_{1,k-1} = \beta^{(k-1)} + H_1(\beta^{(k-1)}, \Lambda^{(k-1)})\nabla_\beta l(\beta^{(k-1)}, \Lambda^{(k-1)})$
   - $B_{2,k-1} = \Lambda^{(k-1)} + H_{2,k-1}\nabla_\Lambda l(\beta^{(k-1)}, \Lambda^{(k-1)})$

- $\tilde{\Lambda}^{(k)} = arg\min_{\Lambda \in \mathcal{C}_+} (\Lambda - B_{2,k-1})^T H_{2,k-1}^{-1} (\Lambda - B_{2,k-1})$ using the Goldfarb-Idnani's duel method that has been implemented in R package.

4. Perform the same line search procedure as described in (3.3) with $\Lambda^{(k-1)}$ replaced by $(\beta^{(k-1)}, \Lambda^{(k-1)})$, $\tilde{\Lambda}^{(k)}$ replaced by $(\beta^{(k)}, \tilde{\Lambda}^{(k)})$, $l(\Lambda^{(k-1)})$ replaced by $l(\beta^{(k-1)}, \Lambda^{(k-1)})$, and $\nabla_\Lambda l(\Lambda^{(k-1)})$ replaced by $\left(\left\{\nabla_\beta l(\beta^{(k-1)}, \Lambda^{(k-1)})\right\}^T, \left\{\nabla_\Lambda l(\beta^{(k-1)}, \Lambda^{(k-1)})\right\}^T\right)^T$, respectively.

5. Check if

$$\|\theta^{(k)} - \theta^{(k-1)}\|_\infty = \max\left(\max_{1\le i\le m}\left|\Lambda_i^{(k)} - \Lambda_i^{(k-1)}\right|, \max_{1\le j\le d}\left|\beta_j^{(k)} - \beta_j^{(k-1)}\right|\right) \le \eta$$

for some small $\eta > 0$. If it is true, stop the iteration; otherwise let $k = k+1$ and repeat the Steps 3-4.

Remark. For the line search, we adopt the doubly step-halving procedure developed in Jongbloed (1998). As an illustration, we describe the line search procedure for the NPMLE as follows: (i) let $x = \Lambda^{(k-1)}$ and $y = \tilde{y} = \tilde{\Lambda}^{(k)}$, if $l(y) > l(x) + \epsilon \{\nabla_\Lambda l(x)\}^T (y-x)$, set $\Lambda^{(k)} = y$ and no further line search is needed; otherwise (ii) let $\lambda = 1, p = 1/2$, and check (ii-a) if $l(y) > l(x) + (1-\epsilon) \{\nabla_\Lambda l(x)\}^T (y-x)$, set $\lambda = \lambda + p$, or (ii-b) if $l(y) < l(x) + \epsilon \{\nabla_\Lambda l(x)\}^T (y-x)$, set $\lambda = \lambda - p$, then let $y = x + \lambda(\tilde{y} - x)$, $p = p/2$ and repeat the circle (ii-a) and (ii-b) until both are not true resulting in the update $\Lambda^{(k)} = y$. This procedure has been proved timely efficient in searching the new update that increases the log-likelihood.

4. Numerical Study

In this section, we report the numerical results from an extensive simulation study and demonstrate the great time-efficiency in computing the NPMLE and SPMLE of panel count data using the projected-NR algorithms, compared to the ICM/extended-ICM algorithms adopted in the literature.

For the NPMLE, we simulate a sample of $\{(K_i, \underline{T}_{K_i}, \underline{N}_{K_i}) : i = 1, 2, \cdots, n\}$ according to the scheme given by Wellner and Zhang (2000). For each subject, K_i is sampled from $\{1, 2, 3, 4, 5, 6\}$; given K_i, a panel of observation times $\underline{T}_{K_i} = (T_{K_i,1}, \cdots, T_{K_i,K_i})$ are made as the order statistics of K_i observations sampled from Unif$(0, 10)$ and are rounded to the second decimal point to make the observation times possibly tied across different

Table 1: Comparison of the algorithms for computing the NPMLE using Monte-Carlo simulation with 100 replicates.

	Dimension of $\mathcal{C}_+$	No. of iterations		Computing time in seconds	
		Proj-NR	ICM	Proj-NR	ICM
		$n = 50$			
mean	122	6	132	1.36	15.56
s.d.	9	1	36	0.27	5.01
		$n = 100$			
mean	224	6	131	4.90	46.24
s.d.	11	1	33	0.90	12.40
		$n = 200$			
mean	400	6	126	16.72	131.94
s.d.	13	1	30	2.83	33.06

subjects; finally given K_i and the panel observation times $\underline{T}_{K_i}$, the panel counts $\underline{N}_{K_i} = (N_i(T_{K_i,1}), \cdots, N_i(T_{K_i,K_i}))$ are sampled from the Poisson process Poisson$(2t)$, i.e.

$$N_i(T_{K_i,j}) - N_i(T_{K_i,j-1}) \sim \text{Poisson}\left(2(T_{K_i,j} - T_{K_i,j-1})\right), \quad j = 1, \cdots, K_i$$

with $T_{K_i,0} \equiv 0$ and $N_i(0) \equiv 0$.

The numerical experiments are conducted with sample size n =50, 100, and 200, respectively. For each sample size, a sample of panel count data are drawn as described above, the NPMLE of the true mean function $\Lambda(t) = 2t$ are computed by the ICM and projected-NR (Proj-NR) with the convergence criterion set as $\eta = 10^{-5}$. The NPMLEs are computed for 100 repeated samples and the results are summarized in Table 1.

Table 1 clearly shows that the quadratic convergence rate of ordinary Newton-Raphson algorithm is also achieved for the proposed projected-NR algorithm, as it reaches the numerical convergence with a single digit of iterations which is much faster than the ICM algorithm. Although each iteration of the projected-NR is costly as it needs to invert a matrix of large dimension, for example, the average dimension of the Hessian matrix is 400 when sample size n=200, the faster convergence of the projected-NR still offsets the numerical complication in inverting a large matrix by consuming much less computing time. In this example, on average, the projected-NR algorithm only needs to spend about 17 seconds to compute the NPMLE with average size of 400 while the simple ICM will spend 7 times more to compute the NPMLE.

Table 2: Comparison of the algorithms for computing the SPMLE using Monte-Carlo simulation with 100 replicates

	Dimension of $\mathcal{C}_+$	Computing time in seconds		
		Proj-NR-1	Proj-NR-2	Ext-ICM
		$n = 50$		
mean	122	23.59	35.98	1045.96
s.d.	9	8.44	11.95	500.10
		$n = 100$		
mean	227	74.54	116.73	2704.45
s.d.	12	16.29	24.34	902.83
		$n = 200$		
mean	402	245.92	373.56	7191.03
s.d.	14	43.02	56.81	1981.34

For the SPMLE, we also generate three covariates for each subject $Z_i = (Z_{i,1}, Z_{i,2}, Z_{i,3})$ using $Z_{i,1} \sim \text{Unif}(0,1)$, $Z_{i,2} \sim N(0,1)$, and $Z_{i,3} \sim \text{Bernoulli}(0.5)$. The simulated data for K_i and $\underline{T}_{K_i}$ are obtained using the same scheme as described above. Given the covariates Z_i, the number of observation times K_i and the panel observation times $\underline{T}_{K_i}$, the panel count data are sampled from a conditional Poisson process with the conditional mean function given by $\Lambda(t|Z_i) = 2t\exp(\beta^T Z_i)$, i.e.

$$N_i(T_{K_i,j}) - N_i(T_{K_i,j-1}) \sim \text{Poisson}\left(2(T_{K_i,j} - T_{K_i,j-1})\exp(\beta^T Z_i)\right), \quad j = 1, \cdots, K_i$$

with $T_{K_i,0} \equiv 0$ and $N_i(0) \equiv 0$, where β is chosen as $\beta^T = (\beta_1, \beta_2, \beta_3) = (-1.0, 0.5, 1.5)^T$.

In addition to the extended ICM algorithm (Ext-ICM) adopted by Wellner and Zhang (2007) and the projected-NR algorithm (Proj-NR-1) proposed in this paper, we also consider another version of the projected-NR algorithm (Proj-NR-2) that is the doubly iterative algorithm described in the beginning of Section 3 with the ICM iteration replaced by the projected-NR iteration. The comparison of the three algorithms in the similarly designed simulation study as that for the NPMLE described above is summarized in Table 2.

For the SPMLE, we do not present the numbers of iterations in Table 2, because they are not comparable as the Proj-NR-2 and Ext-ICM algorithms involve two-level iterations. Actually, the Proj-NR-1 needs more than 160 iterations on average to converge for each of the three scenarios of sample size, showing the convergence in a slower than quadratic rate. It is not surprising since the full Hessian matrix is not always negative definite and only the block diagonal Hessian matrix is utilized in the algorithm in order to insure the global convergence

of the algorithm. Table 2 shows that Ext-ICM algorithm is clearly not an efficient algorithm for computing the SPMLE of panel count data, it needs about 120 minutes on average to achieve the numerical convergence for a moderate sample of $n = 200$. If the ICM step is replaced by the projected-NP in the extended ICM method for updating the estimate of baseline mean function, it shortens the computing time significantly: for sample size $n = 200$, it takes about 6.2 minutes on average to achieve the numerical convergence. With the proposed projected-NR algorithm, despite the lack of quadratic rate of convergence as in the ordinary Newton-Raphson method, the saving in computing time is still substantial: for sample size $n = 200$, it only consumes about 4 minutes to achieve the numerical convergence which is less than 4% of computing time spent in the Ext-ICM algorithm. When sample size is small, the projected-NR algorithm appears even more efficient as the inversion of the Hessian matrix becomes less costly. As for a majority of semiparametric regression applications, estimation with more than 400 unknown parameters is considered to be a relatively "large" problem, the proposed method clearly demonstrated its robustness and numerical efficiency and therefore is recommended to use in the likelihood analysis of panel count data.

Table 3 displays the estimation results of the semi-parametric maximum likelihood method for the regression parameters based on the Monte-carlo simulation study with 100 replicates. It clearly indicates that the estimated regression parameters are asymptotically unbiased and the Monte-Carlo standard deviation decreases as sample size increases, resulting in the mean square errors converging to zero.

Figure 1 plots the mean, 2.5 and 97.5 percentiles of the maximum likelihood estimates of the baseline mean function, respectively, resulted from the Monte-Carlo simulation study with 100 replicates. It is apparently that the mean of the MLEs is very close to the true baseline mean function $\Lambda(t) = 2t$ and the gap between the 2.5 and 97.5 percentiles of the estimated mean function at times $0 < t < 10$ (interior of possible observation times) tends to be smaller as sample size increases from 50 to 200. Hence it provides a numerical evidence for the asymptotic unbiasedness and consistency of the maximum likelihood method for estimating the mean function in semi-parametric regression analysis with panel count data that has been shown theoretically by Wellner and Zhang (2007) under some mild regularity conditions.

All computation tasks for the simulation study are performed with Intel Core 2 CPU 6600 @2.40GHZ and the computing software is developed for R 2.9.1 which can be obtained by requesting to the first author.

Table 3: Estimation Results of the Regression Parameters in the Monte-Carlo Simulation Study with 100 replicates

	$n = 50$	$n = 100$	$n = 200$
Estimation of β_1			
Bias	-0.0063	-0.0012	-0.0076
s.d.	0.1021	0.0705	0.0422
MSE	0.0105	0.0050	0.0018
Estimation of β_2			
Bias	0.0017	0.0025	0.0015
s.d.	0.0318	0.0202	0.0138
MSE	0.0010	0.0004	0.0002
Estimation of β_3			
Bias	0.0091	0.0117	0.0007
s.d.	0.0695	0.0501	0.0311
MSE	0.0049	0.0026	0.0010

5. Summary and Final Remarks

This article proposes a projected Newton-Raphson algorithm, that is, it projects the ordinary Newton-Raphson update to result in a modified Newton-Raphson update inside the feasible region that has the smallest distance (in the sense of a weighted L_2 norm) from the original Newton-Raphson update. This algorithm is successfully applied to compute both the NPMLE and SPMLE of panel count data and demonstrates its great numerical efficiency compared to the ICM methods available in the literature. For the NPMLE, the quadratic rate of convergence for the ordinary Newton-Raphson algorithm is preserved for the proposed project-NR algorithm, as the full Hessian matrix is strictly negative definite. For the SPMLE, this fast convergence rate is not exhibited because only the block diagonal elements of the full Hessian matrix are used to insure the strictly positive definiteness for the weight matrix. Nevertheless, this algorithm still consume much less computing time to compute the SPMLE than the extended-ICM algorithm used in Pan (1999) and Wellner and Zhang (2007).

We should be aware that the proposed method requires the strictly positive definiteness of the weight matrix in both the Newton-Raphson and projection steps. In this paper, we perform dimension reduction before any calculations which guarantees the weight matrix to be strictly positive definite to start with. This step seems to be very crucial from our

numerical experiments: not only it warrants the global convergence, it also largely reduces the computing time due to the dimension reduction.

We also note that the projected-NR algorithm (3.1) can be simplified to

$$x^{(k+1)} = arg\min_{x\in\mathcal{X}} \left\{(x - x^{(k)})^T H^{-1}(x^{(k)})(x - x^{(k)}) - 2(x - x^{(k)})^T \nabla_x l(x^{(k)})\right\}$$

and hence the Goldfarb-Idnani's dual method for QP can be directly applied without the intermediate Newton-Raphson step. However, our numerical studies show that the saving of computing time over the projected-NR is very minimum and is almost ignorable when sample size $n \leq 200$ in our simulation settings.

In our proposed projected-NR algorithm, the (strictly positive definite) weight matrix for the projection step is chosen to be the negative Hessian matrix to accommodate the Newton-Raphson method, the convergence of the algorithm with this choice of the weight matrix along with a proper line search procedure is guaranteed by Theorem 1 of Jongbloed (1998). However, this is not necessary. We may choose some simple weight matrices like the diagonal elements of the negative Hessian matrix and hence the complicated QP algorithm can be replaced by the ICM. This approach has been successfully implemented by Hua (2010) in other applications. However, the adoption of this method, though simpler than the proposed method, does not always lead to numerical convergence in the applications concerned in this paper.

Although the projected-NR algorithm is illustrated in analysis of panel count data, it is generally applicable to other non-parametric and semi-parametric estimation problems in which the infinite-dimensional parameter subjects to monotone constraints. Our recommendation in implementing the projected-NR is as follows: (i) perform dimension reduction first to make sure the Hessian matrix and the block diagonal elements of the Hessian matrix are strictly negative definite; (ii) perform the Newton-Raphson algorithm (for the SPMLE, if the full Hessian matrix is negative definite, using the full Hessian matrix in stead of the block diagonal elements used in this article); (iii) perform the projection step using the diagonal elements of the negative Hessian matrix first (ICM step), if it does not converge, using the full Hessian matrix for the infinite-dimensional parameter instead (Quadratic programming).

REFERENCES

Balakrishnan, N. and Zhao, X. (2009). New multi-sample nonparametric tests for panel count data. *Annals of Statistics* **37**, 1112-1149.

Byar, D. P. (1980). The Veterans Administration study of chemoprophylaxis for recurrent stage I bladder tumors: comparisons of placebo, pyridoxine, and topical thiotepa. In *Bladder Tumors and Other Topics in Urological Oncology*, Eds.M. Pavane-Macaluso, P. H. Smith, and F. Edsmyr. New York: Plenum, 363 - 370.

Goldfarb, D. and Idnani, A. (1983). A numerical stable dual method for solving strictly convex quadratic programs. *Mathematical Programming* **27**, 1-33.

Jongbloed, G (1998). The iterative convex minorant algorithm for nonparametric estimation. *Journal of Computational and Graphical Statistics* **7**, 310-321.

He, X., Tong, X. and Sun, J. (2009). Semiparametric analysis of panel count data with correlated observation and follow-up times. *Lifetime Data Analysis* **15**, 177-196.

Hu, X., Sun, J. and Wei, L. (2003). Regression parameter estimation from panel counts. *Scandinivian Journal of Statistics* **30**, 25-43.

Hu, X., Lagakos, S. and Lockhart, R.. (2009a). Generalized least squares estimation of the mean function of a counting process based on panel counts. *Statistica Sinica* **19**, 561-580.

Hu, X., Lagakos, S. and Lockhart, R.. (2009b). Marginal analysis of panel counts through estimating functions. *Biometrika* **96**, 445-456.

Hua, L.. (2010). *Spline-Based Sieve Semiparametric Generalized Estimating Equation for Panel Count Data.* Ph.D disseartation, Department of Biostatistics, The University of Iowa.

Huang, C-Y., Wang, M-C. and Zhang, Y. (2006). Analysing panel count data with informative observation times. *Biometrika* **93**, 763-775.

Lawless, J. and Nadeau, C. (1995). Some simple robust methods for the analysis of recurrent events. *Technometrics* **37**, 158168.

Lin, D., Wei, L, Yang, I. and Ying, Z. (2000). Semiparametric regression for the mean and rate functions of recurrent events. *Journal of the Royal Statistical Society, Series B* **62**, 711-730.

Lu, M., Zhang, Y. and Huang, J. (2007). Estimation of the mean function with panel count data using monotone polynomial splines. *Biometrika* **94**, 705-718.

Lu, M., Zhang, Y. and Huang, J. (2009). Semiparametric estimation methods for panel count data using monotone B-splines. *Journal of the American Statistical Association* **104**, 1060-1070.

Pan, W. (1999). Extending the iterative convex minorant algorithm to the Cox model for interval-censored data. *Journal of Computational and Graphical Statistics* **8**, 109-120.

Robertson, T., Wright, F. and Dykstra, R. (1988). *Order restricted statistical inference*, New York: Wiley.

Sun, J. and Kalbfleisch, J. D. (1995). Estimation of the mean function of point processes based on panel count data. *Statistica Sinica* **5**, 279 - 290.

Sun, J. and Wei, L.J. (2000). Regression analysis of panel count data with covariate-dependent observation and censoring times. *Journal of the Royal Statistical Society, Series B* **62**, 293-302.

Sun, J. and Fang, H. (2003). A nonparametric test for panel count data. *Biometrika* **90**, 199-208.

Sun, J., Tong, X. and He, X. (2007). Rergression analysis of panel count data with dependent observation times. *Biometrics* **63**, 1053-1059.

Thall, P. F. and Lachin, J. M. (1988). Analysis of recurrent events: nonparametric methods for random-interval count data. *Journal of the American Statistical Association* **83**, 339-347.

Wellner, J. A. and Zhang, Y. (2000). Two estimators of the mean of a counting process with panel count data. *Annals of Statistics* **28**, 779 - 814.

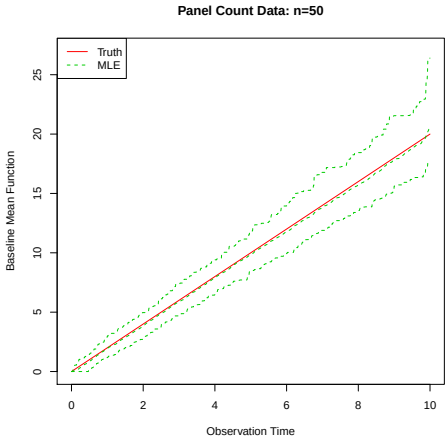
Wellner, J. A. and Zhang, Y. (2007). Two likelihood-based semiparametric estimation methods for panel count data with covariates. *Annals of Statistics* **35**, 2106-2142.

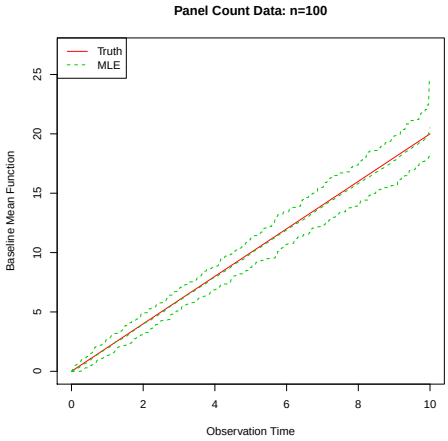
Zhang, Y. (2002). A semiparametric pseudo likelihood estimation method for panel count data. *Biometrika* **89**, 39-48.

Zhang, Y. (2006). Nonparametric k-sample tests with panel count data. *Biometrika* **93**, 777-790.

Zhang, Y. and Jamshidian, M. (2003). The gamma-frailty model for the nonparametric estimation of panel count data. *Biometrics* **59**, 1099-1106.

Zhang, Y. and Jamshidian, M. (2004). On algorithms for the nonparametric maximum likelihood estimator of the failure function with censored data. *Journals of Computational and Graphical Statistics* **13**, 123-140.





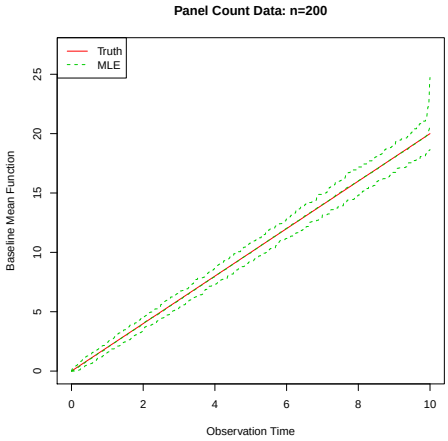


Figure 1. The Monte-Carlo simulation study with 100 replicates for the maximum likelihood estimator of the baseline mean function for the semi-parametric proportional mean model.