

An Approximation Method for a Maximum Likelihood Equation System and Application to the Analysis of Accidents Data

Assi N'Guessan¹, Issa Cherif Geraldo², Bezza Hafidi³

¹Laboratoire Paul Painlevé (UMR CNRS 8524), Université de Lille 1, Villeneuve d'Ascq, France

²Department of Mathematics and Computer science, Université Catholique de l'Afrique de l'Ouest-Unité Universitaire du Togo, Lomé, Togo

³Department of Mathematics, Faculty of Science, University Ibn Zohr, Agadir, Morocco

Email: assi.nguessan@polytech-lille.fr, cherifgera@gmail.com, h.bezza@uiz.ac.ma

How to cite this paper: N'Guessan, A., Geraldo, I.C. and Hafidi, B. (2017) An Approximation Method for a Maximum Likelihood Equation System and Application to the Analysis of Accidents Data. *Open Journal of Statistics*, 7, 132-152. <http://dx.doi.org/10.4236/ojs.2017.71011>

Received: August 8, 2016

Accepted: February 25, 2017

Published: February 28, 2017

Copyright © 2017 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

There exist many iterative methods for computing the maximum likelihood estimator but most of them suffer from one or several drawbacks such as the need to inverse a Hessian matrix and the need to find good initial approximations of the parameters that are unknown in practice. In this paper, we present an estimation method without matrix inversion based on a linear approximation of the likelihood equations in a neighborhood of the constrained maximum likelihood estimator. We obtain closed-form approximations of solutions and standard errors. Then, we propose an iterative algorithm which cycles through the components of the vector parameter and updates one component at a time. The initial solution, which is necessary to start the iterative procedure, is automated. The proposed algorithm is compared to some of the best iterative optimization algorithms available on R and MATLAB software through a simulation study and applied to the statistical analysis of a road safety measure.

Keywords

Constrained Maximum Likelihood, Partial Linear Approximation, Schur's Complement, Iterative Algorithms, Road Safety Measure, Multinomial Model

1. Introduction

Approximation methods for Maximum Likelihood (ML) systems of equations are of interest and are motivated in this paper by the need to find estimation methods that are simple and easy to implement in the specific field of the

statistical evaluation of the impact of a road safety measure. In practice, the estimation methods dedicated to this evaluation depend both on the nature of the measure and the available data. Methods based on frequencies combination have received considerable attention [1] [2] [3] and for the most part of them, we are faced with the estimation of unknown parameters which are often functionally dependent.

Many approximation methods for maximum likelihood estimation need to solve systems of linear or non-linear equations with or without constraints [4]-[9]. Newton-Raphson's method and Fisher scoring are certainly the most commonly used approximation methods. They consist in updating the whole parameter vector using the iterative scheme:

$$\phi^{(k+1)} = \phi^{(k)} + [M(\phi^{(k)})]^{-1} \nabla \ell(\phi^{(k)}) \quad (1)$$

where $\phi^{(k)}$ is the estimate of the vector parameter at the step k , ℓ is the log-likelihood function, $\nabla \ell(\phi^{(k)})$ is the gradient of ℓ and $M(\phi^{(k)})$ is the observed or the expected information matrix. Both methods require the computation of second-order partial derivatives and a matrix inversion in each iteration, which can be very costly. Some authors such as Wang [10] have proposed quadratic approximations and extended Fisher scoring. The main point is to use quadratic approximations to the log-likelihood function and optimize these approximations within the constrained parameter space.

Within the framework of crash data analysis, different iterative estimation methods have been proposed [2] [11]. For example, Mkhadri *et al.* [11] propose a Minorization-Maximization (MM) algorithm for the maximum likelihood estimation of the parameter vector of a multinomial distribution modelling crash data. Their proposed MM algorithm cycles through the components of the parameter vector and updates one component at a time which leads to closed-form expressions of the parameters. They claim that their MM algorithm is simple to implement without any matrix inversion and constraints are integrated easily.

Despite the above advantages, the choice of the starting value $\phi^{(0)}$ remains a major issue since a value of $\phi^{(0)}$ relatively far from the true unknown value of the vector parameter can lead to erroneous solutions or to non-convergence. In addition to this, it must be noted that obtaining explicit expressions of standard errors is not generally easy.

In this paper, we present an estimation method without matrix inversion based on a linear approximation of the likelihood equations in a neighborhood of the constrained maximum likelihood estimator. We obtain closed-form approximations of solutions and standard errors. Then, we propose a partial linear approximation (PLA) algorithm which cycles through the components of the vector parameter and updates one component at a time. The initial solution is automated and standard errors are obtained in a closed-form. The PLA is compared to some of the best available algorithms on R and MATLAB software through a simulation study and applied to real crash data.

The remainder of the paper is organized as follows. Section 2 is devoted to the statistical model and the main assumptions used to get closed-form approximations and standard errors. The proposed estimation method and the method for computing standard errors are presented in Section 3. The general framework of the proposed algorithm is also described. In Section 4, we give an illustration of our results using a crash data model. The numerical performance of the proposed algorithm is examined in Section 5 through a simulation study while a real-data application is given in Section 6. The appendix is devoted to the technical details of the main results.

2. Statistical Model and Main Assumptions

2.1. Statistical Model

Let $Y = (Y_{11}, \dots, Y_{1r}, Y_{21}, \dots, Y_{2r})$ be a random vector with $2r(r > 1)$ components and $\pi(\phi) = (\pi_{11}(\phi), \dots, \pi_{1r}(\phi), \pi_{21}(\phi), \dots, \pi_{2r}(\phi))$ be a vector of probabilities such that

$$\sum_{i=1}^2 \sum_{j=1}^r \pi_{ij}(\phi) = 1$$

where ϕ is a parameter vector. It is assumed that the vector Y has the multinomial distribution $\mathcal{M}(n; \pi(\phi))$ where $n > 0$ is a known integer. The basic principle of the multinomial distribution $\mathcal{M}(n; \pi(\phi))$ consists in distributing n items in $2r$ categories or classes ($2r$ being the number of components of vector $\pi(\phi)$). The probability for an object to fall in a class is called class probability with the sum of all class probabilities equal to 1. Here, the class probabilities $\pi_{ij}(\phi)$ depend on the unknown vector parameter ϕ .

Given a vector of integers $y = (y_{11}, \dots, y_{1r}, y_{21}, \dots, y_{2r})$ such that

$$\sum_{i=1}^2 \sum_{j=1}^r y_{ij} = n,$$

the probability function related to $\mathcal{M}(n; \pi(\phi))$ is defined by

$$f(y) = \frac{n!}{\prod_{i=1}^r \prod_{j=1}^2 y_{ij}!} \prod_{i=1}^r \prod_{j=1}^2 (\pi_{ij}(\phi))^{y_{ij}}. \tag{2}$$

2.2. ML Estimation and Main Assumptions

Assumption 1. The vector parameter ϕ is partitioned as $\phi = (\theta, \beta)$ where $\theta > 0$ is a real parameter, $\beta = (\beta_1, \dots, \beta_r)^T$ is a vector and $\beta_j > 0$ for all $j = 1, \dots, r$.

Assumption 2. The unknown vector ϕ is subject to a linear constraint $C(\phi) = 0$ where C is a continuously differentiable function from \mathbb{R}^{r+1} to \mathbb{R} .

Let $y = (y_{11}, \dots, y_{1r}, y_{21}, \dots, y_{2r}) \in \mathbb{R}^{2r}$ be a vector of observed data and $\ell(\phi)$ be the logarithm of the probability density function $f(y)$ defined by Equation (2). The constrained maximum likelihood estimator $\hat{\phi} = (\hat{\theta}, \hat{\beta})$, is solution to the optimization problem

$$\text{Maximize } \ell(\phi) \text{ subject to } C(\phi) = 0. \tag{3}$$

Problem (3) is equivalent to the maximization of function

$$\mathcal{L}(\phi, \lambda) = \ell(\phi) - \lambda C(\phi) \tag{4}$$

where λ is the Lagrange multiplier. The information matrix linked to the constrained maximum likelihood estimator $\hat{\phi}$ is

$$\Gamma_{\phi} = \begin{bmatrix} J_{\phi} & C_{\phi}^T \\ C_{\phi} & 0 \end{bmatrix}, \quad J_{\phi} = \begin{bmatrix} \tau_{\phi} & U_{\phi}^T \\ U_{\phi} & B_{\phi} \end{bmatrix} \tag{5}$$

where $C_{\phi} = (\partial C / \partial \theta, \partial C / \partial \beta_1, \dots, \partial C / \partial \beta_r)^T \in \mathbb{R}^{1+r}$, $J_{\phi} \in \mathbb{R}^{1+r} \times \mathbb{R}^{1+r}$,

$$\tau_{\phi} = E\left(-\frac{\partial^2 \ell}{\partial \theta^2}\right) \in \mathbb{R}, \quad U_{\phi} = E\left(-\frac{\partial^2 \ell}{\partial \theta \partial \beta_1}, \dots, -\frac{\partial^2 \ell}{\partial \theta \partial \beta_r}\right)$$

and B_{ϕ} is a $r \times r$ matrix whose entries are $E(-\partial^2 \ell / \partial \beta_m \partial \beta_j)$, $m, j = 1, \dots, r$. We also assume that the following conditions are verified:

Assumption 3. $\partial C / \partial \theta = 0$ and $\langle C_{\beta}, \beta \rangle = \kappa \neq 0$ where $\langle \cdot, \cdot \rangle$ is the inner product, κ is a constant and $C_{\beta} = (\partial C / \partial \beta_1, \dots, \partial C / \partial \beta_r)^T \in \mathbb{R}^r$;

Assumption 4. For any $\theta > 0$, there exists a non-singular $r \times r$ matrix $\Omega_{\hat{\theta}, y}$ such that $C_{\hat{\beta}}^T \Omega_{\hat{\theta}, y}^{-1} C_{\hat{\beta}} > 0$ and the non-linear system

$$\left(\frac{\partial \ell}{\partial \beta}\right)_{\hat{\phi}} - \frac{\left\langle \hat{\beta}, \left(\frac{\partial \ell}{\partial \beta}\right)_{\hat{\phi}} \right\rangle}{\left\langle \hat{\beta}, C_{\hat{\beta}} \right\rangle} C_{\hat{\beta}} = 0_r$$

is approximated by the linear system

$$\begin{bmatrix} \Omega_{\hat{\theta}, y} & C_{\hat{\beta}} \\ C_{\hat{\beta}}^T & 0 \end{bmatrix} \begin{bmatrix} \hat{\beta} \\ 0 \end{bmatrix} = \begin{bmatrix} D_y \\ \kappa \end{bmatrix}$$

where $0_r = (0, \dots, 0)^T \in \mathbb{R}^r$ and D_y is a $r \times 1$ vector whose components are obtained from y .

Assumption 5. There exists a function $g : \mathbb{R}^{r-1} \rightarrow \mathbb{R}$ such that the equation $\left(\frac{\partial \ell}{\partial \theta}\right)_{\hat{\phi}} = 0$ is equivalent to $\hat{\theta} = g(\hat{\beta})$.

Assumption 6. There exist two strictly positive real numbers $a_{n, \phi}$ and $b_{n, \phi}$, a non-singular $r \times r$ diagonal matrix Σ_{ϕ} and a vector $V_{\phi} \in \mathbb{R}^r$ such that $\tau_{\phi} = a_{n, \phi}^{-1} b_{n, \phi}^2$, $B_{\phi} = a_{n, \phi} (\Sigma_{\phi} + V_{\phi} V_{\phi}^T)$ and $U_{\phi} = b_{n, \phi} V_{\phi}$.

Assumption 3 specifies the form of function C . Particularly, C is only a function of sub-vector β . Assumptions 4 and 5 enable us to get $\hat{\beta}$ from $\hat{\theta}$ and inversely. Assumption 6 enables us to transform the Fisher information matrix in order to use classical results on matrix inversion with Schur's complement [17].

3. The Estimation Method

3.1. Partial Linear Approximation Principle

The general problem of finding the constrained maximum likelihood estimator

has been discussed by many authors [12] [13] [14]. The classical approach is based on a Newton-type algorithm and computes the components of $\hat{\phi}$ at once. Except from some few simple cases, it is not generally possible to get explicit expressions of the components of $\hat{\phi}$. One shows the following lemma (we refer the reader to the appendix for a proof).

Lemma 1. *The constrained maximum likelihood estimator $\hat{\phi}$, provided it exists, is solution to the non-linear system*

$$\left(\frac{\partial \ell}{\partial \theta}\right)_{\hat{\phi}} = 0 \quad \text{and} \quad \left(\frac{\partial \ell}{\partial \beta}\right)_{\hat{\phi}} - \frac{\left\langle \hat{\beta}, \left(\frac{\partial \ell}{\partial \beta}\right)_{\hat{\phi}} \right\rangle}{\left\langle \hat{\beta}, C_{\hat{\beta}} \right\rangle} C_{\hat{\beta}} = 0_r \tag{6}$$

where $0_r = (0, \dots, 0)^T \in \mathbb{R}^r$.

Our approach consists in dividing Equation (6) into two parts: one concerning the first component of $\hat{\phi}$ and the other one concerning the sub-vector $\hat{\beta}$.

Theorem 1. *Under assumptions 3 - 5, the constrained MLE $\hat{\phi} = (\hat{\theta}, \hat{\beta})$ is given by:*

$$\hat{\theta} = g(\hat{\beta})$$

$$\hat{\beta} = \Omega_{\hat{\theta}, y}^{-1} D_y.$$

The non-obvious part of the proof consists in the determination of $\hat{\beta}$ by inverting the $(r+1) \times (r+1)$ matrix linked to the linear system in Assumption 4. This result based on the inversion of partitioned matrices will not be demonstrated in this paper. We refer the reader to classical papers on Schur complement [15] [16].

From Theorem 1, it is seen that the MLE $\hat{\phi} = (\hat{\theta}, \hat{\beta})$ is a fixed point of the function from \mathbb{R}^{r+1} to itself defined by $(\theta, \beta) \mapsto (g(\beta), \Omega_{\theta, y}^{-1} D_y)$. We can then build an iterative algorithm to estimate ϕ . The classical fixed point method which consists in simultaneously updating $\hat{\theta}$ and $\hat{\beta}$ may be hard to implement because of the link between $\hat{\theta}$ and $\hat{\beta}$. We propose instead to alternate between updating $\hat{\theta}$ holding $\hat{\beta}$ fixed and vice-versa. Starting from a given $\theta^{(0)}$, we compute $\beta^{(0)}$ and then $\theta^{(1)}$ and $\beta^{(1)}$ and so on. The process is repeated until a stopping criteria is satisfied. For example, we can stop the iterations when successive values of the log-likelihood satisfy the condition $|\ell(\phi^{(k+1)}) - \ell(\phi^{(k)})| < \epsilon$ where $\epsilon > 0$.

The estimation process may be completed by the computation of standard errors with Theorem 2 below.

Theorem 2. *Under assumptions 3 - 6, the asymptotic variance of the components of $\hat{\phi}$ are*

$$\hat{\sigma}^2(\hat{\theta}) = a_{n, \hat{\phi}} b_{n, \hat{\phi}}^{-2} \left(1 + \left\| V_{\hat{\phi}} \right\|_{\Sigma^{-1}}^2 - \left\| C_{\hat{\beta}} \right\|_{\Sigma^{-1}}^{-2} \xi_{\hat{\phi}}^2 \right) \tag{7}$$

and

$$\hat{\sigma}^2(\hat{\beta}_j) = a_{n,\hat{\phi}}^{-1} \left(\Sigma_{\hat{\phi},j}^{-1} - \|C_{\hat{\beta}}\|_{\Sigma_{\hat{\beta}}^{-1}}^{-2} \times (\Sigma_{\hat{\phi},j}^{-1} \times C_{\hat{\beta},j})^2 \right), \quad j = 1, \dots, r \quad (8)$$

where $\xi_{\phi} = V_{\phi}^T \Sigma_{\phi}^{-1} C_{\beta} > 0$ and the real values $\Sigma_{\phi,j}^{-1}$ (resp. $C_{\beta,j}$), $j = 1, \dots, r$, are the diagonal elements (resp. components) of matrix Σ_{ϕ}^{-1} (resp. of vector C_{β}).

The proof is given in the appendix. It stems from the results of N'Guessan and Langrand [17].

3.2. General Framework of the Partial Linear Approximation Algorithm

Algorithm 1 (The partial linear approximation algorithm).

Step 0 (Initialization) Given $\theta^{(0)}$, $\epsilon > 0$, D_y , compute $\beta^{(0)} = \Omega_{\theta^{(0)},y}^{-1} D_y$.

Step 1 (Loop for computing $\hat{\phi}$) For a given $k \geq 0$,

a) Compute $\theta^{(k+1)} = g(\beta^{(k)})$ and $\beta^{(k+1)} = \Omega_{\theta^{(k+1)},y}^{-1} D_y$.

b) If $|\ell(\phi^{(k+1)}) - \ell(\phi^{(k)})| \geq \epsilon$, then replace k by $k+1$ and return to Step 1.

Else, set $\hat{\theta} = \theta^{(k+1)}$, $\hat{\beta} = \beta^{(k+1)}$ and go to Step 2.

Step 2 (Computation of standard errors)

a) Compute $\hat{\sigma}^2(\hat{\theta}) = a_{n,\hat{\phi}} b_{n,\hat{\phi}}^{-2} \left(1 + \|V_{\hat{\phi}}\|_{\Sigma_{\hat{\phi}}^{-1}}^2 - \|C_{\hat{\beta}}\|_{\Sigma_{\hat{\beta}}^{-1}}^{-2} \xi_{\hat{\phi}}^2 \right)$.

b) For $j = 1, \dots, r$, compute $\hat{\sigma}^2(\hat{\beta}_j) = a_{n,\hat{\phi}}^{-1} \left(\Sigma_{\hat{\phi},j}^{-1} - \|C_{\hat{\beta}}\|_{\Sigma_{\hat{\beta}}^{-1}}^{-2} \times (\Sigma_{\hat{\phi},j}^{-1} \times C_{\hat{\beta},j})^2 \right)$.

The aim of this paper is not to conduct a theoretical study of the convergence of the proposed algorithm. We rather focus on the numerical aspect of this convergence through an application model. Nevertheless, we can notice that the estimation of $\hat{\phi}$ using our algorithm does not require any matrix inversion. It is thus easy to think that getting the constrained maximum likelihood estimator $\hat{\phi}$ is improved in terms of computation time.

4. Application to the Combination of Crash Data

4.1. Statistical Model

We apply the above algorithm to estimate the parameters of a statistical model used to assess the effect of a road safety measure applied to an experimental site presenting $r > 0$ mutually exclusive accident types (fatal accidents, seriously injured people, slightly injured people, material damage, etc...) over a fixed period. Let us consider the random vector $Y = (Y_{11}, \dots, Y_{1r}, Y_{21}, \dots, Y_{2r})$ where Y_{1j} and Y_{2j} ($j = 1, \dots, r$) respectively represent the number of accidents of type j registered on the experimental site before and after the application of the road safety measure. In order to take into account some external factors (such as traffic flow, speed limit variation, weather conditions, regression to the mean effect, etc...), the experimental site is linked to a control area where the safety measure was not directly applied. The accidents data for the control area over the same period is given by the non-random vector $Z = (z_1, \dots, z_r)^T$ where z_j denotes the ratio of the number of accidents of type j registered in the period

after to the number of accidents of type j registered in the period before. Following N'Guessan *et al.* [18], we assume that the vector Y has the multinomial distribution

$$Y \sim M(n, \pi(\phi))$$

where n is the total number of crashes recorded at the experimental site and $\pi = (\pi_{11}, \dots, \pi_{1r}, \pi_{21}, \dots, \pi_{2r})$ is a vector of class probabilities whose components are

$$\pi_{ij}(\phi) = \begin{cases} \frac{\beta_j}{1 + \theta \sum_{m=1}^r z_m \beta_m} & \text{if } i = 1; j = 1, \dots, r \\ \frac{\theta \beta_j \sum_{m=1}^r z_m \beta_m}{1 + \theta \sum_{m=1}^r z_m \beta_m} & \text{if } i = 2; j = 1, \dots, r. \end{cases} \tag{9}$$

By construction, the parameter vector $\phi = (\theta, \beta)$ of this model satisfies the conditions $\theta > 0$, $\beta_j > 0$ and the linear constraint

$$C(\phi) = 0, \quad \text{with } C(\phi) = \langle 1_r, \beta \rangle - 1 \tag{10}$$

where $1_r = (1, \dots, 1) \in \mathbb{R}^r$. The scalar θ denotes the unknown parameter average effect of the road safety measure while each β_j ($j = 1, \dots, r$) denotes the risk of having an accident of type j on a site having the same characteristics as the experimental site. This model is a special case of the multinomial model proposed by N'Guessan *et al.* [18] which was applied simultaneously on several sites.

4.2. Cyclic Estimation of the Average Effect and the Different Accident Risks

The log-likelihood is specified up to an additive constant by

$$\ell(\phi) = \sum_{m=1}^r \left[y_{1m} \log(\beta_m) + y_{2m} \log(\theta) - y_{1m} \log\left(1 + \theta \sum_{j=1}^r z_j \beta_j\right) + y_{2m} \log\left(\sum_{j=1}^r z_j \beta_j\right) \right] \tag{11}$$

where $y_{.m} = y_{1m} + y_{2m}$. Different iterative methods can be used to compute the constrained MLE $\hat{\phi}$. Most of them look for stationary points of the Lagrangian

$$\mathcal{L}(\phi, \lambda) = \ell(\phi) - \lambda \left(\sum_{m=1}^r \beta_m - 1 \right).$$

N'Guessan *et al.* [18] showed that a stationary point of $\mathcal{L}(\phi, \lambda)$ must be the solution to the following system of non-linear equations:

$$\begin{cases} \sum_{m=1}^r \left(y_{2m} - y_{1m} \frac{\hat{\theta} \hat{E}(Z)}{1 + \hat{\theta} \hat{E}(Z)} \right) = 0 \\ y_j - \frac{n \hat{\beta}_j (1 + \hat{\theta} z_j)}{1 + \hat{\theta} \hat{E}(Z)} - \frac{y_{2j} \hat{\beta}_j (\hat{E}(Z) - z_j)}{\hat{E}(Z)} = 0, \quad j = 1, \dots, r \\ \hat{\theta} > 0, \hat{\beta}_j > 0, \quad j = 1, \dots, r \end{cases} \tag{12}$$

where $y_{2.} = \sum_{m=1}^r y_{2m}$ and $\hat{E}(Z) = \sum_{m=1}^r \hat{\beta}_m z_m$.

The main idea proposed in this paper consists in neglecting the term $\left[y_{2.} \hat{\beta}_j (\hat{E}(Z) - z_j) \right] / \hat{E}(Z)$ so that we can write the remaining equations

$$y_{.j} - \frac{n \hat{\beta}_j (1 + \hat{\theta} z_j)}{1 + \hat{\theta} \hat{E}(Z)} = 0, \quad j = 1, \dots, r$$

as the linear system of equations

$$\Omega_{\hat{\theta}, y} \hat{\beta} = D_y \tag{13}$$

where $\Omega_{\hat{\theta}, y} = M_{\hat{\theta}} - \hat{\theta} D_y Z^T$, $M_{\hat{\theta}} = \text{diag}(1 + \hat{\theta} z_1, \dots, 1 + \hat{\theta} z_r)$ is a diagonal $r \times r$ matrix and

$$D_y = \left(\frac{y_{11} + y_{21}}{n}, \dots, \frac{y_{1r} + y_{2r}}{n} \right)^T \in \mathbb{R}^r.$$

Drawing our inspiration from the work by N'Guessan [19], we show that the linear system (13) has the vector $\hat{\beta} = (1 - \hat{\theta} Z^T M_{\hat{\theta}}^{-1} D_y)^{-1} M_{\hat{\theta}}^{-1} D_y$ as unique solution. One shows that

$$(1 - \hat{\theta} Z^T M_{\hat{\theta}}^{-1} D_y) = 1 - \frac{1}{n} \sum_{m=1}^r \frac{\hat{\theta} z_m y_{.m}}{1 + \hat{\theta} z_m}.$$

The components of the constrained MLE $\hat{\phi}$ can then be computed as follows.

Corollary 1. *The components of $\hat{\phi} = (\hat{\theta}, \hat{\beta})$ are given by*

$$\hat{\theta} = \frac{\sum_{m=1}^r y_{2m}}{\left(\sum_{m=1}^r \hat{\beta}_m z_m \right) \times \left(\sum_{m=1}^r y_{1m} \right)} \tag{14}$$

$$\hat{\beta}_j = \frac{1}{1 - \Delta_n(\hat{\theta})} \times \frac{1}{1 + \hat{\theta} z_j} \times \frac{y_{.j}}{n}, \quad j = 1, \dots, r \tag{15}$$

where $\Delta_n(\hat{\theta}) = \frac{1}{n} \sum_{m=1}^r \frac{\hat{\theta} z_m y_{.m}}{1 + \hat{\theta} z_m}$.

Applying Theorem 2's results, we can give the asymptotic variance of the constrained MLE $\hat{\phi}$.

Corollary 2. *The asymptotic variance of the components of the constrained maximum likelihood estimator $\hat{\phi}$ is given by*

$$\hat{\sigma}^2(\hat{\theta}) = \frac{1}{n \hat{E}(Z)} \hat{\theta} + \frac{1}{n} \left(1 + \frac{\hat{E}(Z^2)}{(\hat{E}(Z))^2} \right) \theta^2 + \frac{\hat{E}(Z)}{n} \theta^3 \tag{16}$$

$$\hat{\sigma}^2(\hat{\beta}_j) = \frac{1}{n} \hat{\beta}_j (1 - \hat{\beta}_j), \quad j = 1, \dots, r \tag{17}$$

where $\hat{E}(Z) = \sum_{m=1}^r \hat{\beta}_m z_m$ and $\hat{E}(Z^2) = \sum_{m=1}^r \hat{\beta}_m z_m^2$.

Technical proof uses the Schur complement approach and stems from [17]. One shows (see the appendix) that the elements of the asymptotic information matrix $\Gamma_{\hat{\phi}}$ linked to $\hat{\phi}$ are

$$C_\phi = (0, 1, \dots, 1)^T \in \mathbb{R}^{1+r}, \quad \tau_\phi = \frac{\gamma^2 E(Z)}{n\theta}, \quad U_\phi = \frac{\gamma^2}{n} Z, \quad B_\phi = \gamma(\Sigma_\phi + V_\phi V_\phi^T) \quad (18)$$

where

$$V_\phi = \left(\frac{\theta\gamma}{nE(Z)} \right)^{1/2} Z, \quad \Sigma_\phi = \frac{n}{\gamma} \times \text{diag} \left(\frac{1}{\beta_1}, \dots, \frac{1}{\beta_r} \right), \quad \gamma = \frac{n}{1 + \theta E(Z)}. \quad (19)$$

Setting $a_{n,\phi} = \gamma$ and $b_{n,\phi}^2 = (\gamma^3 E(Z))/(n\theta)$, we show that Assumption 6 is satisfied. We then apply Theorem 2 to get the results of Corollary 2.

4.3. Practical Aspect of the Partial Linear Approximation Algorithm

Algorithm 2.

Step 0 (Initialization) Given $\epsilon > 0$ and D_y , set $\theta^{(0)} = 0$ and compute $\beta^{(0)} = D_y$.

Step 1 (Loop for computing $\hat{\phi}$) For a given $k \geq 0$,

a) Compute
$$\theta^{(k+1)} = \frac{\sum_{m=1}^r y_{2m}}{\left(\sum_{m=1}^r \beta_m^{(k)} z_m \right) \times \left(\sum_{m=1}^r y_{1m} \right)}$$

b) For $j = 1, \dots, r$, compute

$$\beta_j^{(k+1)} = \frac{1}{1 - \frac{1}{n} \sum_{m=1}^r \frac{\theta^{(k+1)} z_m y_m}{1 + \theta^{(k+1)} z_m}} \times \frac{1}{1 + \theta^{(k+1)} z_j} \times \frac{y_{.j}}{n}.$$

c) If $\left| \ell(\phi^{(k+1)}) - \ell(\phi^{(k)}) \right| \geq \epsilon$, then replace k by $k + 1$ and return to **Step 1**.

Else, set $\hat{\theta} = \theta^{(k+1)}$, $\hat{\beta} = \beta^{(k+1)}$ and go to **Step 2**.

Step 2 (Computation of standard errors)

a) Compute
$$\hat{\sigma}^2(\hat{\theta}) = \frac{1}{n\hat{E}(Z)} \hat{\theta} + \frac{1}{n} \left(1 + \frac{\hat{E}(Z^2)}{(\hat{E}(Z))^2} \right) \theta^2 + \frac{\hat{E}(Z)}{n} \theta^3.$$

b) For $j = 1, \dots, r$, compute
$$\hat{\sigma}^2(\hat{\beta}_j) = \frac{1}{n} \hat{\beta}_j (1 - \hat{\beta}_j).$$

The partial linear approximation algorithm for computing the constrained maximum likelihood estimator $\hat{\phi}$ of the model presented in subsection 4.1 stems from the cyclic algorithm of N'Guessan and Geraldo [20]. The PLA proceeds as follows: step 1 allows to estimate $\hat{\phi}$ alternating between its two components $\hat{\theta}$ et $\hat{\beta}$. To start the procedure, we initialize $\theta^{(0)}$. Then we compute $\beta_j^{(0)}$ ($j = 1, \dots, r$) and define $\phi^{(0)} = (\theta^{(0)}, \beta^{(0)})$. But, we could also initialize $\beta^{(0)}$ using the problem's data and get $\theta^{(0)}$. The process is repeated until the stopping criterion is satisfied. We note that our algorithm is automated and can be started as soon as the problem's data D_y is entered.

We can also note that the second partial derivatives of the log-likelihood function are no longer used in our algorithm. The aim of this paper is not to carry out a theoretical study of the convergence of the proposed algorithm. We rather focus on the numerical aspect of this convergence using simulated datasets.

5. Numerical Results with Simulated Datasets

5.1. Data Simulation Principle

For a given value of r (the number of crash types), we generate the components of vector $Z = (z_1, \dots, z_r)^T$ from a uniform random variable $U\left(\frac{1}{2}, \frac{5}{2}\right)$.

The true value of θ denoted θ^0 is fixed and the true value of β , denoted $\beta^0 = (\beta_1^0, \dots, \beta_r^0)^T$, such that $\sum_{j=1}^r \beta_j^0 = 1$, comes from a uniform random variable $U(\epsilon, 1-\epsilon)$ where $\epsilon = 10^{-5}$. Using those values, we compute the true probabilities

$$\pi_{1j}(\phi^0) = \frac{\beta_j^0}{1 + \theta^0 \sum_{m=1}^r z_m \beta_m^0}, \quad \pi_{2j}(\phi^0) = \frac{\theta^0 \beta_j^0 \sum_{m=1}^r z_m \beta_m^0}{1 + \theta^0 \sum_{m=1}^r z_m \beta_m^0}, \quad j = 1, \dots, r$$

linked to the multinomial distribution presented in subsection 4.1. Finally, one generates the total number n of crash data from a Poisson distribution and then randomly shares it between the before and after periods using probabilities $\pi_{1j}(\phi^0)$ and $\pi_{2j}(\phi^0)$. The observed values of y_{ij} such that $\sum_{i=1}^2 \sum_{j=1}^r y_{ij} = n$ are then found.

5.2. Numerical Results

This subsection deals with the numerical convergence of the partial linear approximation algorithm. As usual in the study of iterative algorithms, we analyse the influence of the initial solution $\phi^{(0)} = (\theta^{(0)}, \beta^{(0)})$, the number of iterations, the computation time (CPU time) and the mean squared error. The performances of the partial linear approximation algorithm are compared to those of some classical optimization methods available in R and MATLAB software. The computations presented in this section were executed on a PC with an AMD E-350 Processor 1.6 GHz CPU.

The methods selected for comparison are the Newton-Raphson's method, Nelder-Mead's (NM) simplex algorithm [21], quasi-Newton BFGS algorithm (from the names of its authors Broyden, Fletcher, Goldfarb and Shanno [22] [23] [24] [25]), Interior Point (IP) algorithm [26], the Levenberg-Marquardt (LM) algorithm [27] [28] and Trust Region (TR) algorithms [29]. In our work, the BFGS and NM algorithms are implemented using the package developed by Varadhan [30].

The simulation process was performed on many simulated crash datasets. For each one, small and large values of n were considered. The results presented here correspond to the case $r = 3$, $n \in \{50; 5000\}$, $\phi^0 = (\theta^0, \beta^0)$ with $\theta^0 = 0.6$ and $\beta^0 = (0.025, 0.232, 0.743)$.

Three different ways of setting $\beta^{(0)}$ were considered:

1) Uniform initialisation: $\beta_1^{(0)} = \beta_2^{(0)} = \dots = \beta_r^{(0)} = 1/r$.

2) Proportional initialisation: $\beta_j^{(0)} = \frac{y_{1j} + y_{2j}}{n}$, $j = 1, \dots, r$.

3) Random initialisation: for $j = 1, \dots, r$, $\beta_j^{(0)} = u_j / \left(\sum_{i=1}^r u_i \right)$ where each

$u_i (i = 1, \dots, r)$ is randomly generated from an uniform distribution $U(0.05, 0.95)$.

By combining the two values of n and the three ways to initialize $\beta^{(0)}$, we get six scenarios and for each one, we performed 1000 replications. The stopping criterion is the condition $|\ell(\phi^{(k+1)}) - \ell(\phi^{(k)})| < 10^{-5}$.

Tables 1-6 present a few of the results obtained for an overall 6000 simulations. All computation times are given in seconds and the duration ratio of an

Table 1. Results for uniform initialisation of $\beta^{(0)}$, $r = 3$, $n = 50$.

	R software				MATLAB software			
	PLA	NR	BFGS	NM	PLA	LM	TR	IP
$\hat{\phi}$	0.621	0.617	0.617	0.618	0.643	0.639	0.639	0.639
	0.060	0.054	0.054	0.054	0.060	0.054	0.054	0.054
	0.207	0.227	0.227	0.227	0.203	0.223	0.223	0.223
	0.733	0.719	0.719	0.719	0.737	0.723	0.723	0.723
Min. iterations	2	2	11	8	2	3	2	8
Max. iterations	4	4	11	19	4	4	11	22
Mean iterations	3.1	3.0	11.0	12.1	3.1	3.2	5.8	13.8
CPU time	7.64E-04	3.84E-03	1.83E-01	2.85E-01	1.41E-03	4.37E-02	5.07E-02	4.97E-01
Duration ratio	1	5	239	373	1	31	36	353
MSE	9.90E-03	9.74E-03	9.74E-03	9.79E-03	1.12E-02	1.09E-02	1.09E-02	1.09E-02

Table 2. Results for uniform initialisation of $\beta^{(0)}$, $r = 3$, $n = 5000$.

	R software				MATLAB software			
	PLA	NR	BFGS	NM	PLA	LM	TR	IP
$\hat{\phi}$	0.604	0.601	0.601	0.602	0.604	0.601	0.601	0.601
	0.028	0.025	0.025	0.025	0.028	0.025	0.025	0.025
	0.212	0.232	0.232	0.233	0.212	0.232	0.232	0.232
	0.760	0.743	0.743	0.742	0.760	0.743	0.743	0.743
Min. iterations	4	3	15	12	4	3	3	9
Max. iterations	4	4	15	20	4	4	12	21
Mean iterations	4.0	3.3	15.0	15.1	4.0	3.0	6.4	13.7
CPU time	9.35E-04	4.11E-03	3.23E-01	3.54E-01	1.71E-03	4.24E-02	5.30E-02	5.33E-01
Duration ratio	1	4	345	379	1	25	31	311
MSE	2.82E-04	9.62E-05	9.62E-05	1.18E-04	2.77E-04	9.24E-05	9.24E-05	9.24E-05

Table 3. Results for proportional initialisation of $\beta^{(0)}$, $r=3$, $n=50$.

	R software				MATLAB software			
	PLA	NR	BFGS	NM	PLA	LM	TR	IP
$\hat{\phi}$	0.626	0.622	0.622	0.623	0.632	0.628	0.628	0.628
	0.061	0.055	0.055	0.055	0.060	0.053	0.053	0.053
	0.206	0.226	0.226	0.226	0.205	0.226	0.226	0.226
	0.733	0.720	0.720	0.719	0.735	0.721	0.721	0.721
Min. iterations	3	2	11	8	3	2	2	6
Max. iterations	3	3	11	19	3	3	13	20
Mean iterations	3.0	2.9	11.0	12.0	3.0	3.0	6.1	11.5
CPU time	7.11E-04	3.75E-03	1.85E-01	2.81E-01	1.30E-03	4.26E-02	5.14E-02	4.99E-01
Duration ratio	1	5	260	395	1	33	39	383
MSE	1.00E-02	9.86E-03	9.86E-03	9.84E-03	1.15E-02	1.13E-02	1.13E-02	1.13E-02

Table 4. Results for proportional initialisation of $\beta^{(0)}$, $r=3$, $n=5000$.

	R software				MATLAB software			
	PLA	NR	BFGS	NM	PLA	LM	TR	IP
$\hat{\phi}$	0.604	0.601	0.660	0.662	0.604	0.601	0.601	0.601
	0.028	0.025	0.025	0.026	0.028	0.025	0.025	0.025
	0.211	0.232	0.232	0.233	0.211	0.232	0.232	0.232
	0.760	0.743	0.743	0.742	0.760	0.743	0.743	0.743
Min. iterations	4	2	1	1	4	2	2	3
Max. iterations	4	3	15	21	4	3	12	20
Mean iterations	4.0	2.9	14.6	14.7	4.0	3.0	6.3	12.5
CPU time	9.20E-04	3.69E-03	3.10E-01	3.47E-01	1.59E-03	4.19E-02	5.21E-02	5.31E-01
Duration ratio	1	4	337	378	1	26	33	334
MSE	2.76E-04	8.96E-05	5.17E-02	5.16E-02	2.82E-04	9.30E-05	9.30E-05	9.30E-05

algorithm is defined as the ratio between the mean computation time of this latter and the mean computation time of the PLA (therefore the duration ratio of the partial linear approximation algorithm always equals 1). The computation time depends on the computer used to perform the simulations while the duration ratio is computer-free and therefore more useful.

To analyse the convergence, we used the mean squared error (MSE) defined as:

Table 5. Results for random initialisation of $\beta^{(0)}$, $r = 3$, $n = 50$.

	R software				MATLAB software			
	PLA	NR	BFGS	NM	PLA	LM	TR	IP
$\hat{\phi}$	0.639	0.635	0.932	0.939	0.632	0.628	0.628	0.628
	0.060	0.053	0.094	0.065	0.060	0.054	0.054	0.054
	0.205	0.226	0.243	0.247	0.204	0.224	0.224	0.224
	0.735	0.721	0.663	0.688	0.736	0.722	0.722	0.722
Min. iterations	2	2	1	1	2	3	3	8
Max. iterations	4	4	11	20	4	4	12	22
Mean iterations	3.4	3.1	9.5	10.5	3.4	3.4	5.9	13.5
CPU time	7.76E-04	3.92E-03	1.60E-01	2.75E-01	1.39E-03	4.29E-02	4.96E-02	4.95E-01
Duration ratio	1	5	207	354	1	31	36	357
MSE	1.16E-02	1.13E-02	2.52E-01	2.47E-01	1.08E-02	1.05E-02	1.05E-02	1.05E-02

Table 6. Results for random initialisation of $\beta^{(0)}$, $r = 3$, $n = 5000$.

	R software				MATLAB software			
	PLA	NR	BFGS	NM	PLA	LM	TR	IP
$\hat{\phi}$	0.603	0.600	0.839	0.841	0.603	0.600	0.600	0.600
	0.028	0.025	0.067	0.036	0.028	0.025	0.025	0.025
	0.211	0.232	0.244	0.252	0.211	0.232	0.232	0.232
	0.760	0.743	0.689	0.712	0.760	0.743	0.743	0.743
Min. iterations	3	2	1	1	3	3	3	8
Max. iterations	4	4	15	20	4	4	12	31
Mean iterations	4.0	3.4	13.2	13.4	4.0	3.4	6.5	14.1
CPU time	8.55E-04	4.40E-03	2.84E-01	3.31E-01	1.58E-03	4.30E-02	5.40E-02	5.34E-01
Duration ratio	1	5	332	387	1	27	34	338
MSE	2.80E-04	9.40E-05	1.85E-01	1.75E-01	2.81E-04	9.55E-05	9.55E-05	9.55E-05

$$MSE(\hat{\phi}, \phi^0) = \frac{1}{1+r} \left((\hat{\theta} - \theta^0)^2 + \sum_{j=1}^r (\hat{\beta}_j - \beta_j^0)^2 \right) \tag{20}$$

One can see that the MSE decreases when n (the total number of road crashes) increases. The PLA is at least as efficient as the other algorithms. It always converges to reasonable and acceptable parameter vector estimates and the estimate gets closer to the the true value as n increases.

For a small value of n (see **Table 1**, **Table 3** and **Table 5**), the estimate $\hat{\phi}$

produced by the PLA is relatively close to the true parameter vector and quite close to those of the other methods. More generally, all the compared methods have a MSE of order 10^{-2} . However the estimates produced by the BFGS and Nelder-Mead's methods are very far from the true values (see **Table 5**). In the case of random initialisation, the MSE for Nelder-Mead's and BFGS algorithms is 20 times greater than those of the other algorithms.

When n increases ($n = 5000$), the MSE of the PLA decreases from 10^{-2} to 10^{-4} . So the PLA is also efficient. Unsurprisingly, when n is very great the estimates produced by the other algorithms are closer to the true values than those of the PLA. This is expected because the other algorithms work with the exact gradient. However, the Nelder-Mead's and BFGS algorithms produce estimates very far from the truth. Their MSE's order is 10^{-2} (**Table 4** and **Table 6**) while those of the PLA and the other methods are 10^{-4} .

To analyse the influence of the initial guess, we considered the mean number of iterations and the amplitude of the iterations (*i.e.* difference between the maximum and the minimum number of iterations). An increase in the amplitude of iterations suggests a greater influence of the initial solution $\phi^{(0)}$. The results given in **Tables 1-6** suggest that the PLA is stable and robust to initial guesses of the parameter being estimated. For the 6000 replications, the number of iterations needed by the PLA to converge lies between 2 and 4. In other words, setting the initial guess to 6000 different values chosen in the parameter vectors space does not disturb the PLA. This performance is as good as those of Newton-Raphson and Levenberg-Marquardt and by far better than those of BFGS, Nelder-Mead's and Interior Point which have their number of iterations varying respectively from 1 to 15, 1 to 21 and 3 to 31.

As far as the computation time is concerned, it can be noticed that all the CPU time ratio are greater than 1 which means that none of the compared algorithms is faster than the PLA. On average, the PLA is 4 to 5 times quicker than Newton-Raphson, 239 to 345 times quicker than BFGS algorithm, 354 to 395 times quicker than Nelder-Mead, 27 to 31 times quicker than Levenberg-Marquardt, 33 to 39 times quicker than Trust region algorithms and 311 to 383 times quicker than Interior point algorithms. This can be an important factor when larger values of r are considered.

6. Real-Data Analysis

We apply the partial linear approximation algorithm to the data concerning the changes applied to road markings on a rural Nord Pas-de-Calais road (France) [31]. This road lay-out consisted in what is called "Italian marking". The road markings were modified in order to make overtaking impossible in both directions at the same time. On a short distance, there are two lanes for the first direction (overtaking is then allowed in that direction) and there is only one for the other direction. Both directions are separated by a white line. A bit farther away, the system is inverted so that overtaking is possible in the opposite direction, and so on. The data recorded for eight years (four years before and

four years after) on the marked area are given by **Table 7**.

Note that the number of fatal crashes (Fatal) decreased from four (in the before period) to one after the lay-out change. Indeed there were four accidents with at least one person killed in the period before the road works and one crash with at least one person killed in the period after. The number of slight crashes (no serious bodily injuries involved) recorded in the same period were more than halved. For the same lengths of time, on a portion of National Road 17 used as a control area, accident reports are the following.

The values given in **Table 8** are obtained by dividing, for each accident type, the number of crashes after the changes by the number of crashes before. On the whole, a decrease can be noted in comparison with the control area accident numbers between the 4-year period after the changes and the 4-year period before. All the algorithms can be applied to **Table 7** and **Table 8**. Since the simulation results suggested that the partial linear approximation algorithm converges after a few iterations and remains steady, we only present the estimations related to the partial linear approximation algorithm (**Table 9**).

Parameters β_1 , β_2 and β_3 enable us to assess the risks for each type of accident on the test area in the eight years when the road markings' effects are studied. Estimated values $\hat{\beta}_1$, $\hat{\beta}_2$ and $\hat{\beta}_3$ are respectively 0.1525, 0.1605 and 0.6870. These values suggest that in the eight years of road marking analysis, 15.25% of the crashes recorded in the test area were fatal, 16.05% were serious and 68.70% were slight as compared to the crashes recorded in the control area.

The estimated mean efficiency index ($\hat{\theta}$) is 0.7054. It corresponds to an average decrease in proportion of $29.46\% = (1 - 0.7054) \times 100\%$ of the whole

Table 7. Crash data for experimental zone.

Accident data before			Accident data after			Total
Fatal	Serious	Slight	Fatal	Serious	Slight	
4	4	16	1	1	7	33

Table 8. Crash data for control zone.

Fatal	Serious	Slight
0.519	0.422	0.560

Table 9. Pas-De-Calais's crash data using partial linear approximation algorithm.

Parameters	Estimations	Standard error	95% Confidence interval	
			Lower bound	Upper bound
θ (Mean efficiency index)	0.7054	0.2760	0.1645	1.2463
β_1 (Fatal risk)	0.1525	0.0626	0.0298	0.2752
β_2 (Serious risk)	0.1605	0.0639	0.0353	0.2858
β_3 (Slight risk)	0.6870	0.0807	0.5287	0.8452

set of accidents in the test area as compared to the average trend in the control area. The mean effect significance for this type of lay-out may also be tested by using the confidence interval at the 95% level associated to parameter θ . This confidence interval reveals a bracket of values whose lower (resp. upper) bound is strictly superior to 0 (resp. 1). Indeed, as $1 \in [0.1646, 1.2463]$, we cannot rule out the hypothesis $H_0 : \theta = 1$ versus $H_1 : \theta \neq 1$ with a type-1 error of 5%. Even if in the case studied here, we can notice a decrease in proportion of $29.46\% = (1 - 0.7054) \times 100\%$ in the average accident number in the test area, the above test shows that the mean efficiency index value is not significantly different from 1 to enable us to conclude that this type of road marking is efficient. In practice, an analysis according to periods, recorded data and control area should be carried out in order to get more appropriate conclusions.

7. Concluding Remarks

We propose in this paper, under assumptions, a principle of 2-block splitting of a constrained Maximum Likelihood (ML) system of equations linked to a parameter vector. We obtain analytical approximations of the components of the constrained ML estimator. The standard asymptotic errors are also obtained in closed-form.

We then build an iterative algorithm by initializing the first component of the parameter vector without inverting (at each iteration) the information matrix nor the Hessian matrix. Our partial linear approximation algorithm cycles through the components of the parameter vector and updates one component at a time. It is very simple to program and the constraints are integrated easily. To implement our algorithm, we use a particular version of the multinomial model of N'Guessan *et al.* [18] used to estimate the average effect of a road safety measure and the different accident risks when road conditions are improved. We prove that the assumptions are all satisfied and we obtain simple expressions of the estimators and their asymptotic variance. Afterwards, we give a practical version of our algorithm.

The numerical performance of the proposed algorithm is examined through a simulation study and compared to those of classical methods available in R and MATLAB software. The choice of these other methods is dictated not only by the fact that they are relevant and integrated in most statistical software but also by the fact that some need second order derivatives and others do not. The comparisons suggest that not only the partial linear approximation algorithm is competitive in statistical accuracy and computational speed with the best currently available algorithm, but also it is not disturbed by the initial guess.

The link between the numerical performance of our algorithm and the particular model used in this paper seems to be a limitation to the competitiveness of our algorithm with regards to the other methods considered in this paper. However, this particular choice of model allows not only to show the feasibility of our algorithm but also represents a good basis for approaching, under additional conditions, more general models.

The simulation results obtained on the particular model considered in this paper suggest that our algorithm may be extended to other families of multinomial models (such as the model of N'Guessan *et al.* [18]). Drawing our inspiration from [32], we may also prove the asymptotic strong consistency of the estimator obtained from our algorithm. This perspective will give a wider interest to our algorithm in the context of a large-scale use.

Acknowledgements

We thank the Editor and the anonymous referee for their remarks and suggestions which enabled a substantial improvement of this paper. This article was partially written while the second author was visiting the Lille 1 University (France).

References

- [1] Hauer, E. (1997) *Observational Before-After Studies in Road Safety Estimating the Effect of Highway Traffic Engineering Measures on Road Safety*. Pergamon, Oxford.
- [2] N'Guessan, A., Essai, A. and Langrand, C. (2001) Estimation multidimensionnelle des contrôles et de l'effet moyen d'une mesure de sécurité routière. *Revue de Statistique Appliquée*, **49**, 85-102.
- [3] Lord, D. and Mannering, F. (2010) The Statistical Analysis of Crash-Frequency Data: A Review and Assessment of Methodological Alternatives. *Transport Research Part A*, **44**, 291-305.
- [4] Aitchison, J. and Silvey, S.D. (1958) Maximum Likelihood Estimation of Parameters Subject to Restraints. *The Annals of Mathematical Statistics*, **29**, 813-828. <https://doi.org/10.1214/aoms/1177706538>
- [5] Crowder, M. (1984) On the Constrained Maximum Likelihood Estimation with Non-I.I.D Observations. *Annals of the Institute of Statistical Mathematics*, **36**, 239-249. <https://doi.org/10.1007/BF02481968>
- [6] Haber, M. and Brown, M.B. (1986) Maximum Likelihood Methods for Log-Linear Models When Expected Frequencies Are Subject to Linear Constraints. *Journal of the American Statistical Association*, **81**, 477-482. <https://doi.org/10.1080/01621459.1986.10478293>
- [7] Lange, K. (2010) *Numerical Analysis for Statisticians*. 2nd Edition, Springer, Berlin. <https://doi.org/10.1007/978-1-4419-5945-4>
- [8] Li, Z.F., Osborne, M.R. and Prvan, T. (2003) Numerical Algorithms for Constrained Maximum Likelihood Estimation. *ANZIAM Journal*, **45**, 91-114. <https://doi.org/10.1017/S1446181100013171>
- [9] Slivapulle, M.J. and Sen, P.K. (2005) *Constrained Statistical Inference*. Wiley, Hoboken.
- [10] Wang, Y. (2007) Maximum Likelihood Computation Based on the Fisher Scoring and Gauss-Newton Quadratic Approximations. *Computational Statistics & Data Analysis*, **51**, 3776-3787. <https://doi.org/10.1016/j.csda.2006.12.037>
- [11] Mkhadri, A., N'Guessan, A. and Hafidi, B. (2010) An MM Algorithm for Constrained Estimation in a Road Safety Measure Modeling. *Communications in Statistics - Simulation and Computation*, **39**, 1057-1071. <https://doi.org/10.1080/03610911003778119>
- [12] El Barmi, H. and Dykstra, R.L. (1998) Maximum Likelihood Estimates via Duality

- for Log-Convex Models When Cell Probabilities Are Subject to Convex Constraints. *The Annals of Statistics*, **26**, 1878-1893. <https://doi.org/10.1214/aos/1024691361>
- [13] Matthews, G.B. and Crowther, N.A.S. (1995) A Maximum Likelihood Estimation Procedure When Modelling in Terms of Constraints. *South African Statistical Journal*, **29**, 29-51.
- [14] Liu, C. (2000) Estimation of Discrete Distributions with a Class of Simplex Constraints. *Journal of the American Statistical Association*, **95**, 109-120. <https://doi.org/10.1080/01621459.2000.10473907>
- [15] Ouellette, D.V. (1981) Schur Complement and Statistics. *Linear Algebra and Its Applications*, **36**, 187-295. [https://doi.org/10.1016/0024-3795\(81\)90232-9](https://doi.org/10.1016/0024-3795(81)90232-9)
- [16] Zhang, F. (2005) *The Schur Complement and Its Applications*. Springer, Berlin. <https://doi.org/10.1007/b105056>
- [17] N'Guessan, A. and Langrand, C. (2005) A Covariance Components Estimation Procedure When Modelling a Road Safety Measure in Terms of Linear Constraints. *Statistics*, **39**, 303-314. <https://doi.org/10.1080/02331880500108544>
- [18] N'Guessan, A., Essai, A. and N'zi, M. (2006) An Estimation Method of the Average Effect and the Different Accident Risks When Modelling a Road Safety Measure: A Simulation Study. *Computational Statistics & Data Analysis*, **51**, 1260-1277. <https://doi.org/10.1016/j.csda.2005.09.002>
- [19] N'Guessan, A. (2010) Analytical Existence of Solutions to a System of Non-Linear Equations with Application. *Journal of Computational and Applied Mathematics*, **234**, 297-304. <https://doi.org/10.1016/j.cam.2009.12.026>
- [20] N'Guessan, A. and Geraldo, I.C. (2015) A Cyclic Algorithm for Maximum Likelihood Estimation Using Schur Complement. *Numerical Linear Algebra with Applications*, **22**, 1161-1179. <https://doi.org/10.1002/nla.1999>
- [21] Nelder, J.A. and Mead, R. (1965) A Simplex Algorithm for Function Minimization. *Computer Journal*, **7**, 308-313. <https://doi.org/10.1093/comjnl/7.4.308>
- [22] Broyden, C.G. (1970) The Convergence of a Class of Double-Rank Minimization Algorithms. *Journal of the Institute of Mathematics and Its Applications*, **6**, 76-90. <https://doi.org/10.1093/imamat/6.1.76>
- [23] Fletcher, R. (1970) A New Approach to Variable Metric Algorithms. *The Computer Journal*, **13**, 317-322. <https://doi.org/10.1093/comjnl/13.3.317>
- [24] Goldfarb, D. (1970) A Family of Variable Metric Updates Derived by Variational Means. *Mathematics of Computation*, **24**, 23-26. <https://doi.org/10.1090/S0025-5718-1970-0258249-6>
- [25] Shanno, D.F. (1970) Conditioning of Quasi-Newton Methods for Function Minimization. *Mathematics of Computation*, **24**, 647-656. <https://doi.org/10.1090/S0025-5718-1970-0274029-X>
- [26] Waltz, R.A., Morales, J.L., Nocedal, J. and Orban, D. (2006) An Interior Algorithm for Nonlinear Optimization That Combines Line Search and Trust Region Steps. *Mathematical Programming*, **107**, 391-408. <https://doi.org/10.1007/s10107-004-0560-5>
- [27] Levenberg, K. (1944) A Method for the Solution of Certain Problems in Least Squares. *Quarterly of Applied Mathematics*, **2**, 164-168. <https://doi.org/10.1090/qam/10666>
- [28] Marquardt, D. (1963) An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *SIAM Journal on Applied Mathematics*, **11**, 431-441. <https://doi.org/10.1137/0111030>

- [29] Byrd, R.H., Schnabel, R. and Shultz, G.A. (1988) Approximate Solution of the Trust Region Problem by Minimization over Two-Dimensional Subspaces. *Mathematical Programming*, **40**, 247-263. <https://doi.org/10.1007/BF01580735>
- [30] Varadhan, R. (2011) Alabama: Constrained Nonlinear Optimization. R Package Version 9-1.
- [31] N'Guessan, A. and Truffier, M. (2008) Impact d'un aménagement de sécurité routière sur la gravité des accidents de la route. *Journal de la Société Française de Statistiques*, **149**, 23-41.
- [32] Geraldo, I.C., N'Guessan, A. and Gneyou, K.E. (2015) A Note on the Strong Consistency of a Constrained Maximum Likelihood Estimator Used in Crash Data Modelling. *Comptes Rendus Mathématique*, **353**, 1147-1152. <https://doi.org/10.1016/j.crma.2015.09.025>

8. Appendix

8.1. Proof of Lemma 1

Proof. Problem (3) is equivalent to maximizing function

$$\mathcal{L}(\phi, \lambda) = \ell(\phi) - \lambda C(\phi) \tag{21}$$

where λ is the Lagrange multiplier. Any solution $\hat{\phi}$ must satisfy

$$\left(\frac{\partial \mathcal{L}}{\partial \theta}\right)_{\hat{\phi}} = 0 \quad \text{and} \quad \left(\frac{\partial \mathcal{L}}{\partial \beta_j}\right)_{\hat{\phi}} = 0, \quad j = 1, \dots, r. \tag{22}$$

From assumption 3, system (22) is equivalent to

$$\left(\frac{\partial \ell}{\partial \theta}\right)_{\hat{\phi}} = 0 \quad \text{and} \quad \left(\frac{\partial \ell}{\partial \beta_j}\right)_{\hat{\phi}} - \hat{\lambda} \left(\frac{\partial C}{\partial \beta_j}\right)_{\hat{\phi}} = 0, \quad j = 1, \dots, r \tag{23}$$

where $\hat{\lambda} = \lambda(\hat{\phi})$. After multiplication by $\hat{\beta}_j$ and summation on the index j , we get:

$$\sum_{j=1}^r \hat{\beta}_j \left(\frac{\partial \ell}{\partial \beta_j}\right)_{\hat{\phi}} - \hat{\lambda} \sum_{j=1}^r \hat{\beta}_j \left(\frac{\partial C}{\partial \beta_j}\right)_{\hat{\phi}} = 0.$$

that is equivalent to

$$\left\langle \hat{\beta}, \left(\frac{\partial \ell}{\partial \beta}\right)_{\hat{\phi}} \right\rangle - \hat{\lambda} \left\langle \hat{\beta}, C_{\hat{\beta}} \right\rangle = 0.$$

We obtain (6) by substitution of $\hat{\lambda}$ in (23). □

8.2. Proof of Theorem 2

Proof. Under conditions 6, Γ_{ϕ} is non singular and after some matrix manipulations, we get

$$\Gamma_{\phi}^{-1} = \begin{bmatrix} W_{\phi} & J_{\phi}^{-1} C_{\phi}^T R_{\phi}^{-1} \\ R_{\phi}^{-1} C_{\phi} J_{\phi}^{-1} & -R_{\phi}^{-1} \end{bmatrix},$$

$$J_{\phi}^{-1} = \begin{bmatrix} (J_{\phi}/B_{\phi})^{-1} & -(J_{\phi}/B_{\phi})^{-1} U_{\phi}^T B_{\phi}^{-1} \\ -B_{\phi}^{-1} U_{\phi} (J_{\phi}/B_{\phi})^{-1} & (J_{\phi}/\tau_{\phi})^{-1} \end{bmatrix}$$

where $R_{\phi}^{-1} = a_{n,\phi} \|C_{\beta}\|_{\Sigma_{\phi}^{-1}}^{-2}$ is a scalar, $B_{\phi}^{-1} = a_{n,\phi}^{-1} \left(\Sigma_{\phi}^{-1} - \left(1 + \|V_{\phi}\|_{\Sigma_{\phi}^{-1}}^2\right)^{-1} \Sigma_{\phi}^{-1} V_{\phi} V_{\phi}^T \Sigma_{\phi}^{-1} \right)$

is the inverse of B_{ϕ} , $(J_{\phi}/B_{\phi})^{-1} = b_{n,\phi}^{-2} a_{n,\phi} \left(1 + \|V_{\phi}\|_{\Sigma_{\phi}^{-1}}^2\right)$, is a scalar,

$(J_{\phi}/\tau_{\phi})^{-1} = a_{n,\phi}^{-1} \Sigma_{\phi}^{-1}$, is a $r \times r$ matrix and $W_{\phi} = J_{\phi}^{-1} - J_{\phi}^{-1} C_{\phi}^T R_{\phi}^{-1} C_{\phi} J_{\phi}^{-1}$ is the $(1+r) \times (1+r)$ asymptotic variance-covariance matrix of ϕ . We show that

$$W_{\phi} = \begin{bmatrix} W_{\phi}(1,1) & W_{\phi}^T(2,1) \\ W_{\phi}(2,1) & W_{\phi}(2,2) \end{bmatrix}$$

where $W_{\phi}(1,1)$ is a scalar and $W_{\phi}(2,2)$ is a $r \times r$ matrix. The asymptotic variance of $\hat{\theta}$ is given by $W_{\phi}(1,1)$ and $\hat{\sigma}^2(\hat{\beta}_j)$ is obtained by the diagonal

elements of $W_\phi(2, 2)$. After straightforward calculation, we get

$$W_\phi(1, 1) = (J_\phi/B_\phi)^{-1} - R_\phi^{-1} \xi_\phi^2 (J_\phi/B_\phi)^{-2} \frac{b_{n,\phi}^2 a_{n,\phi}^{-2}}{(1 + \|V_\phi\|_{\Sigma_\phi^{-1}}^2)^2}$$

and

$$W_\phi(2, 2) = a_{n,\phi}^{-1} \left(\Sigma_\phi^{-1} - \|C_\beta\|_{\Sigma_\phi^{-1}}^{-2} \Sigma_\phi^{-1} C_\beta C_\beta^T \Sigma_\phi^{-1} \right)$$

where $\xi_\phi = V_\phi^T \Sigma_\phi^{-1} C_\beta$. The results of Theorem 2 follow from a direct calculation. □

8.3. Proof of Corollary 2

Proof. Taking the second derivatives of the negative of \mathcal{L} with respect to the components of ϕ and λ , we show that: $\frac{\partial^2 \mathcal{L}}{\partial \lambda^2} = \frac{\partial^2 \mathcal{L}}{\partial \lambda \partial \theta} = 0$, $\frac{\partial^2 \mathcal{L}}{\partial \lambda \partial \beta_j} = -1$,

$$\frac{\partial^2 \mathcal{L}}{\partial \theta^2} = -\frac{y_2}{\theta^2} + \frac{n(E(Z))^2}{(1 + \theta E(Z))^2}, \quad \frac{\partial^2 \mathcal{L}}{\partial \theta \partial \beta_j} = -\frac{nz_j}{(1 + \theta E(Z))^2} \text{ and}$$

$$\frac{\partial^2 \mathcal{L}}{\partial \beta_j \partial \beta_k} = \begin{cases} -\frac{y_j}{\beta_j^2} + \frac{n\theta^2 z_j^2}{(1 + \theta E(Z))^2} - \frac{y_2 z_j^2}{(E(Z))^2} & \text{if } j = k \\ \frac{n\theta^2 z_j z_k}{(1 + \theta E(Z))^2} - \frac{y_2 z_j z_k}{(E(Z))^2} & \text{if } j \neq k \end{cases}$$

for $j, k = 1, \dots, r$. Now, setting $E(y_j) = n\beta_j$, $E(y_2) = n\theta E(Z)/(1 + \theta E(Z))$ and $\gamma = n/(1 + \theta E(Z))$, we obtain the components of matrix J_ϕ as follows

$$\tau_\phi = \frac{\gamma^2 E(Z)}{n\theta}, \quad U_{\phi,j} = \frac{\gamma^2 z_j}{n}, \quad B_{\phi,ij} = \gamma \left(\frac{n}{\gamma \beta_j} + \frac{\gamma \theta^2 z_j^2}{nE(Z)} \right) \text{ and } B_{\phi,jk} = \frac{\theta \gamma^2 z_j z_k}{nE(Z)}, (j \neq k)$$

where $U_{\phi,j}$ (resp. $B_{\phi,ij}$ and $B_{\phi,jk}$) are the components of vector U_ϕ (resp. the elements of matrix B_ϕ). Using the last expressions of the elements of U_ϕ and B_ϕ and after straightforward calculation, we show that

$$W_\phi(2, 2) = \gamma^{-1} \Sigma_\phi^{-1} - \frac{1}{n} \beta \beta^T \text{ and then, we deduce the expression of } \hat{\sigma}^2(\hat{\beta}_j)$$

given by corollary 2. In the same way, we obtain

$$W_\phi(1, 1) = \frac{n\theta}{\gamma^2 t_\phi E(Z)} - \frac{\theta^2}{n}$$

where $t_\phi = [n^2 E(Z)] / [n^2 E(Z) + \theta \gamma^2 \hat{E}(Z^2)]$. Using the last expression of t_ϕ , we get the expression of $\hat{\sigma}^2(\hat{\theta})$. □

Submit or recommend next manuscript to SCIRP and we will provide best service for you:

Accepting pre-submission inquiries through Email, Facebook, LinkedIn, Twitter, etc.

A wide selection of journals (inclusive of 9 subjects, more than 200 journals)

Providing 24-hour high-quality service

User-friendly online submission system

Fair and swift peer-review system

Efficient typesetting and proofreading procedure

Display of the result of downloads and visits, as well as the number of cited articles

Maximum dissemination of your research work

Submit your manuscript at: <http://papersubmission.scirp.org/>

Or contact ojs@scirp.org