

Comparison of a cyclic algorithm with some classical optimization methods for constrained estimation of a multinomial distribution

Assi N'GUESSAN ^{*1} and Issa Cherif GERALDO ^{†2}

¹Ecole Polytechnique Universitaire de Lille et Laboratoire Paul Painlevé UMR CNRS 8524, Université de Lille 1, 59655 Villeneuve d'Ascq Cedex, France – assi.nguessan@polytech-lille.fr

²Département de Mathématiques et Informatique, Université Catholique de l'Afrique de l'Ouest, Unité Universitaire du Togo (UCAO–UUT), B.P. 1502 Lomé (Togo) – cherifgera@gmail.com

Abstract

This paper deals with the numerical convergence of a cyclic iterative optimization algorithm (CA algorithm) recently developed for maximum likelihood estimation of the parameters of a multinomial model used in road safety modelling. We compare the CA algorithm to some classical constrained optimization algorithms as Nelder–Mead's (NM), Broyden–Fletcher–Goldfarb–Shanno's (BFGS) and Conjugate Gradient (CG). Road accident data is simulated afterwards each of the compared method is applied. In addition to the estimates of the parameters, some useful information as the mean squares error have been computed. The results obtained, for an overall fifty-four thousand (54 000) simulations, show that the CA algorithm is more efficient not only as far as the accuracy is concerned but also and most importantly it is upto 1300 times quicker than the three others.

Keywords - Road safety modelling, Cyclic algorithm, Maximum likelihood, Multinomial distribution, Constrained optimization.

1 Presentation of the model

This is the before-after multinomial model used by N'GUESSAN and TRUFFIER [6] to assess the impact of a change in road conditions on the incidence of crashes. This model is an extension of the one described by TANNER [8] to evaluate the effect of modifications on the road network in the case where several mutually exclusive types of crashes are considered simultaneously.

Consider an integer R representing the number of different types of crashes, the vectors $x_1 = (x_{11}, \dots, x_{1R})^T$ and $x_2 = (x_{21}, \dots, x_{2R})^T$ representing respectively the accidents data before and after installation of changes in the experimental zone, with x_{1j} (resp. x_{2j}) the number of accidents of type j occurred before (resp. after) installation of changes in the experimental zone. Also consider a vector $c = (c_1, c_2, \dots, c_R)^T$ of known constants.

*First corresponding author: Assi.Nguessan@polytech-lille.fr, Bât. Polytech'Lille, Université Lille 1, 59655 Villeneuve d'Ascq, France. Fax : +33 3 28 76 73 01, Phone : +33 3 28 76 74 57 (Assi N'Guessan)

†Second corresponding author: cherifgera@gmail.com, Bât. S.A.P., Université Catholique de l'Afrique de l'Ouest – Unité Universitaire du Togo, B.P. 1502, Togo. Phone: +228 22 38 92 68 (Cherif Geraldo)

Let $\alpha = (\alpha_0, \pi_1, \pi_2, \dots, \pi_R)^T$ a $(1 + R) \times 1$ parameter vector of unknown components such that:

$$\alpha_0 > 0, \quad \pi_j > 0 \quad \text{and} \quad \sum_{j=1}^R \pi_j = 1 \quad (1)$$

The parameter α_0 is the average effect on the incidence of crashes in the experimental zone after installation of changes, $\pi_j(\alpha)$ is the probability of having a type j accident in the experimental zone both periods (before and after) confounded.

The vector $(x_1, x_2) = (x_{11}, \dots, x_{1R}, x_{21}, \dots, x_{2R})^T$ is assumed to have the following multinomial distribution:

$$\mathcal{M}(n; p_{11}(\alpha), \dots, p_{1R}(\alpha), p_{21}(\alpha), \dots, p_{2R}(\alpha))$$

with

$$p_{1j}(\alpha) = \frac{\pi_j}{1 + \alpha_0 \sum_{j=1}^R c_j \pi_j}, \quad p_{2j}(\alpha) = \frac{\alpha_0 c_j \pi_j}{1 + \alpha_0 \sum_{j=1}^R c_j \pi_j} \quad \forall j = 1, \dots, R \quad (2)$$

respectively the probabilities of having a type j crash in the before (resp. after) period and

$$n = \sum_{j=1}^R x_{\bullet j}, \text{ the total number of accidents.}$$

2 Estimation of the parameters

2.1 The constrained maximum likelihood approach

The log-likelihood associated to a given data $(x_{11}, \dots, x_{1R}, x_{21}, \dots, x_{2R})$ can be calculated as:

$$LL(x_1, x_2; \alpha) = \text{Constant} + \sum_{j=1}^R \left\{ x_{\bullet j} \log(\pi_j) + x_{2j} \log(\alpha_0) - x_{\bullet j} \log \left(1 + \alpha_0 \sum_{m=1}^R c_m \pi_m \right) \right\} \quad (3)$$

where $x_{\bullet j} = x_{1j} + x_{2j}$.

An estimate $\hat{\alpha} = (\hat{\alpha}_0, \hat{\pi}_1, \dots, \hat{\pi}_R)^T$ of the true parameter vector $\alpha = (\alpha_0, \pi_1, \dots, \pi_R)^T$ can be obtained by resolving the following constrained optimization problem:

$$\left\{ \begin{array}{l} \max_{\alpha} LL(x_1, x_2; \alpha) \\ \text{subject to:} \\ \alpha_0 > 0, \quad \pi_j > 0, \forall j = 1, \dots, R \\ \text{and} \quad \sum_{j=1}^R \pi_j = 1 \end{array} \right. \quad (4)$$

Many iterative methods can be used to solve the problem (4) such as Newton-Raphson (NR), Nelder-Mead algorithm (NM) (see [4]), Boyden-Fletcher-Goldfarb-Shanno method (BFGS) (see [1, 2, 3, 7]), Conjugate Gradient (CG). It is well known that the NR method requires the computation of the log-likelihood gradient and most importantly the Hessian matrix while NM, CG and BFGS do not need the expression of the log-likelihood function's hessian matrix. Indeed BFGS and CG only need the log-likelihood's gradient while NM does not even need it.

In order to avoid the computation of the log-likelihood gradient and most importantly the Hessian matrix that can be very costly, N'GUESSAN [5] proposed the following approach. After introduction of one Lagrange multiplier due to the equality constraint in (4) and after calculation of the partial derivatives of the log-likelihood function with regards to each component of the parameter vector α , it is found that (4) is equivalent to the following system:

$$\begin{cases} \sum_{j=1}^R \left\{ x_{2j} - x_{\bullet j} \frac{\hat{\alpha}_0 \sum_{m=1}^R c_m \hat{\pi}_m}{1 + \hat{\alpha}_0 \sum_{m=1}^R c_m \hat{\pi}_m} \right\} = 0 \\ x_{\bullet j} - n \frac{\hat{\pi}_j (1 + c_j \hat{\alpha}_0)}{1 + \hat{\alpha}_0 \sum_{m=1}^R c_m \hat{\pi}_m} = 0, \quad (j = 1, 2, \dots, R) \end{cases} \quad (5)$$

2.2 The CA Algorithm

The Cyclic Algorithm (CA algorithm) was built using the following approach. N'GUESSAN [5] proposed to set the first component α_0 and then solve the second subsystem with regards to the components $\pi_1, \pi_2, \dots, \pi_R$ using some matrix transformations. Afterwards he used the solution to solve the first equation of (5) with respect to α_0 , and vice versa. It is proved (see [5]) that the components of the solution $\hat{\alpha} = (\hat{\alpha}_0, \hat{\pi}_1, \dots, \hat{\pi}_R)$ of the problem (5) can be obtained as:

$$\begin{cases} \hat{\alpha}_0 = \frac{\sum_{m=1}^R x_{2m}}{\left(\sum_{m=1}^R c_m \hat{\pi}_m \right) \times \left(\sum_{m=1}^R x_{1m} \right)}; \\ \hat{\pi}_j = \frac{1}{1 - \frac{1}{n} \sum_{m=1}^R \frac{\hat{\alpha}_0 c_m x_{\bullet m}}{1 + \hat{\alpha}_0 c_m}} \times \frac{x_{\bullet j}}{n(1 + \hat{\alpha}_0 c_j)}, \\ (j = 1, 2, \dots, R). \end{cases} \quad (6)$$

The algorithm can be summarized in the following lines:

1. **Initialisation:** Give a value to $\hat{\pi}^{(0)} = \left(\hat{\pi}_1^{(0)}, \dots, \hat{\pi}_R^{(0)} \right)^T$.
2. At the $(k + 1)$ -th step:
 - (a) Calculate $\hat{\alpha}_0^{(k+1)}$ from $\hat{\pi}^{(k)}$ using equation (6).
 - (b) Calculate $\hat{\pi}^{(k+1)}$ from $\hat{\alpha}_0^{(k+1)}$ using equation (6).
3. **Stop criterion:** stop when no considerable increase is observed in the log-likelihood function.

3 Numerical results

N'GUESSAN [5] has already dealt with the Newton-Raphson algorithm for solving the problem (4) and he showed that the CA Algorithm is more accurate than NR. Our work focuses on comparison of the CA algorithm to the other classical constrained optimization algorithms mentioned above i.e. NM, BFGS and CG.

The results presented in this work correspond to the case $R = 3$, $\alpha_0 = 0.8$, $\pi_1 = 0.025$, $\pi_2 = 0.232$, $\pi_3 = 0.743$ and $c_1 = c_2 = c_3 = 1$.

The initial parameter vector $\hat{\pi}^{(0)}$'s components are calculated as: $\pi_j^{(0)} = e_j / \left(\sum_{m=1}^R e_m \right)$, $j = 1, \dots, R$ where $e = (e_1, \dots, e_R)$ is a vector of R randomly generated values from an uniform distribution between 0.01 and 1. The parameter $\alpha_0^{(0)}$ is set by the user.

Knowing the true values of $p = (p_{11}(\alpha), \dots, p_{1R}(\alpha), p_{21}(\alpha), \dots, p_{2R}(\alpha))$, $N = 1\,000$ repetitions of the following process are done: a vector x whose distribution is the multinomial distribution

$$\mathcal{M}(n; p_{11}(\alpha), \dots, p_{1R}(\alpha), p_{21}(\alpha), \dots, p_{2R}(\alpha))$$

is simulated; afterwards vector x_1 (resp. x_2) components are deduced as the first (resp. last) R values of x ; afterwards the parameter vector is estimated with each of the four compared methods.

For a given method $meth \in \{CA, NM, CG, BFGS\}$ and a parameter vector estimate $\hat{\alpha}_{meth(k)} = (\hat{\alpha}_{0,meth(k)}, \hat{\pi}_{1,meth(k)}, \dots, \hat{\pi}_{R,meth(k)})$ of dimension $(1 + R)$ estimated with $meth$ for the k^{th} repetition ($k = 1, \dots, N$), the estimated parameter vector, $\hat{\alpha}_{meth}$, is calculated as:

$$\hat{\alpha}_{meth} = \frac{1}{N} \sum_{k=1}^N \hat{\alpha}_{meth(k)}$$

In order to check the efficiency of the estimation, the mean squared error (MSE) associated to the method $meth$ is calculated as:

$$\frac{1}{N(1 + R)} \sum_{k=1}^N \|\hat{\alpha}_{meth(k)} - \alpha\|^2 \quad (7)$$

with α the true parameter vector.

A few results corresponding to two values of the total number of crashes n (small value $n = 50$ and great value $n = 5000$) are presented in tables 1 and 2. In these tables, mean values of iterations are calculated only when there is convergence. The duration ratio of the method $meth$ is the ratio between the mean duration of $meth$ and the mean duration of the CA algorithm (i.e. the duration ratio of the CA algorithm always equals 1). A duration ratio greater than 1 means that CA algorithm is quicker. It is computer-free while the time duration depends on the computer.

It can be seen in tables 1 and 2 that CA algorithm is at least as accurate as the others but most importantly CA algorithm is much more quicker than the three others i.e. it needs much less iterations and time.

References

- [1] BROYDEN, C. G. The Convergence of a Class of Double-rank Minimization Algorithms. *Journal of the Institute of Mathematics and Its Applications*, (6): 76–90, 1970.
- [2] FLETCHER, R. A new approach to variable metric algorithms. *The Computer Journal*, 13(3):317–322, 1970.
- [3] GOLDFARB, D. A Family of Variable Metric Updates Derived by Variational Means. *Mathematics of Computation*, (24): 23–26, 1970.

Table 1: Case of small number of road crashes

	TRUE	CA	NM	CG	BFGS
$\hat{\alpha}_0$	0.8	0.8357	0.8679	0.852	0.8261
$\hat{\pi}_1$	0.025	0.0537	0.0491	0.0541	0.0535
$\hat{\pi}_2$	0.232	0.2245	0.4452	0.224	0.2243
$\hat{\pi}_3$	0.743	0.7218	0.5057	0.722	0.7222
Number of repetitions		1000	1000	1000	1000
Number of convergence		1000	1000	693	1000
Mean number of iterations		2	14.632	18.0981	17.355
Mean duration in seconds		2,00E-04	0.0386	0.1962	0.2395
Duration ratio		1	217.78	1108.43	1353.28
MSE	0	0.017	0.0623	0.0168	0.0167

Table 2: Case of great number of road crashes

	TRUE	CA	NM	CG	BFGS
$\hat{\alpha}_0$	0.8	0.8002	0.8295	0.7988	0.7974
$\hat{\pi}_1$	0.025	0.025	0.0229	0.0251	0.025
$\hat{\pi}_2$	0.232	0.232	0.4518	0.2319	0.232
$\hat{\pi}_3$	0.743	0.743	0.5253	0.7431	0.743
Number of repetitions		1000	1000	1000	1000
Number of convergence		1000	1000	782	1000
Mean number of iterations		2	14.838	3.8427	4.219
Mean duration in seconds		2,00E-04	0.0381	0.0405	0.0947
Duration ratio		1	226.55	240.87	563.79
MSE	0	1,00E-04	0.0443	1,00E-04	1,00E-04

- [4] NELDER, J. A., MEAD, R. A simplex algorithm for function minimization. *Computer Journal*, (7): 308–313, 1965.
- [5] N’GUESSAN, A. Analytical Existence of solutions to a system of nonlinear equations with application. *Journal of Computational and Applied Mathematics*, (234): 297–304, 2010.
- [6] N’GUESSAN, A., TRUFFIER, M. Impact d’un aménagement de sécurité routière sur la gravité des accidents de la route. *Journal de la Société Française de Statistiques*, (2): 99–117, 2008.
- [7] SHANNO, D. F. Conditioning of Quasi-Newton Methods for Function Minimization. *Mathematics of Computation*, (24): 647–656, 1970.
- [8] TANNER, J. C. A problem in the combination of accident frequencies. *Biometrika*, (45): 331–342, 1958.