Comput Stat (2009) 24:67–81
DOI 10.1007/s00180-007-0106-4

ORIGINAL PAPER

The constrained Fisher scoring method for maximum likelihood computation of a nonparametric mixing distribution

Yong Wang

Received: 4 October 2007 / Accepted: 20 December 2007 / Published online: 12 January 2008

Abstract This paper proposes a new method for computing the nonparametric maximum likelihood estimate of a mixing distribution. It uses the Fisher scoring quadratic approximation to the log-likelihood function of the mixing proportions. At each iteration, new candidate support points are found and included, as guided by the gradient function, and bad support points are discarded, after being found redundant by optimizing the quadratic approximation. Numerical studies show that the CFS method is generally competitive with the fast and stable constrained Newton method; it may even have an advantage over the latter when the initial estimate is badly chosen.

Keywords Constrained Newton method · EM algorithm · Fisher scoring · Information matrix · Iteratively reweighted least squares · Mixture model · Nonparametric maximum likelihood

1 Introduction

Computing the nonparametric maximum likelihood estimate (NPMLE) of a mixing distribution has been known to be challenging (Lindsay 1995; Böhning 2000). Although the consistency of the estimator was established by Kiefer and Wolfowitz (1956), computational methods only started to emerge nearly 20 years later. Methods that can be used for this purpose include those proposed by Wynn (1970); Fedorov (1972); Atwood (1976); Simar (1976); Laird (1978); Wu (1978a,b); Böhning (1985); Lesperance and Kalbfleisch (1992) and Wang (2007c). Some of them were proposed

Y. Wang (✉)
Department of Statistics, The University of Auckland, Private Bag 92019,
Auckland 1142, New Zealand
e-mail: yongwang@stat.auckland.ac.nz

for finding optimal designs of experiments but, as realized later, can also be used for finding an NPMLE.

The constrained Newton algorithm with multiple inclusion (CNM) that is recently proposed by Wang (2007c) extends the quadratic method of Atwood (1976) and appears to be most efficient for finding an accurate NPMLE. The CNM algorithm is iterative and can be simply described as follows. At each iteration, it (a) expands the support set of the mixing distribution by including all local maxima of the gradient function; (b) updates all mixing proportions via a quadratically convergent method; and, (c) shrinks the support set by discarding the support points with mass 0. The quadratic programming subproblem suggested for updating the mixing proportions has been shown to be equivalent to a linear regression problem, which can be solved conveniently and accurately. Since the algorithm can quickly find and discard redundant support points, the support set that is maintained throughout the computation is usually small, thus avoiding an explosive increase in the number of support points, a problem that plagues all previous methods. The numerical studies conducted by Wang (2007c) show that the CNM algorithm outperforms the other methods remarkably and is very fast and stable.

Instead of using the second-order Taylor series expansion to approximate the log-likelihood function, as in the CNM algorithm and Atwood (1976) quadratic method, one may also consider replacing the exact Hessian matrix with others. Wu (1978a,b) extends Atwood (1976) quadratic method to a family of similar methods, which may include the matrices that are used in the gradient projection method, the conjugate gradient projection method, the Newton method and the quasi-Newton methods.

In this paper, we propose a new algorithm for computing the NPMLE, as an alternative to the CNM algorithm. It differs from the CNM algorithm mainly in that it is the expected Fisher information matrix that is used in the quadratic approximation function, not the observed Fisher information matrix, following an idea explored in Wang (2007b). This thus leads to an algorithm that is similar to the Fisher scoring method, but it works in a constrained parameter space. We thus call it the constrained Fisher scoring (CFS) method. As demonstrated below, the CFS algorithm is nearly as fast as the CNM algorithm and similarly stable. It is acknowledged that CFS is only competitive and does not generally outperform CNM, which is understandable, but in certain situations CFS or its variants may offer some advantages, as discussed below.

For reason of simplicity, throughout the paper we will frequently omit the word “candidate” before “support point” or “support set.” It should be aware that these points do not necessarily have a positive mass.

The rest of the paper is organized as follows. In Sect. 2, we propose the CFS method for computing the mixing proportions. Section 3 discusses some numerical strategies for implementing the method and advocates a particular technique called iteratively reweighted least integrated squares. The CFS method for NPMLE computation is described and studied in Sect. 4. Section 5 gives two examples of numerical studies of some NPMLE algorithms, including CFS. Summary and some remarks are given in Sect. 6.

2 The constrained Fisher scoring method for computing mixing proportions

This section proposes the constrained Fisher scoring method for computing mixing proportions. We first briefly introduce the problem and the notations used below, which follow those by Wang (2007c). Let $x_1, \ldots, x_n \in \mathcal{X} \subset \mathbb{R}$ be a random sample drawn from the mixture distribution with density

$$f(x; G) = \int f(x; \theta)\,\mathrm{d}G(\theta). \tag{1}$$

Since the NPMLE is known to be discrete (or, if not unique, there must be a discrete NPMLE) (Laird 1978; Lindsay 1983), we only need to concentrate on discrete mixing distributions. Denote $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_m)^{\mathrm{T}}$, $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_m)^{\mathrm{T}}$, and

$$G(\theta) = \sum_{j=1}^{m} \pi_j \delta_{\theta_j}(\theta), \quad \boldsymbol{\pi}^{\mathrm{T}}\mathbf{1} = 1 \text{ and } \boldsymbol{\pi} \geq \mathbf{0},$$

where $\mathbf{1} = (1, \ldots, 1)^{\mathrm{T}}$, $\mathbf{0} = (0, \ldots, 0)^{\mathrm{T}}$, and δ_{θ_j} puts probability 1 at θ_j. Density (1) will thus also be written as

$$f(x; \boldsymbol{\pi}, \boldsymbol{\theta}) = \sum_{j=1}^{m} \pi_j f(x; \theta_j), \tag{2}$$

and the log-likelihood function as

$$\ell(G) = \ell(\boldsymbol{\pi}, \boldsymbol{\theta}) = \sum_{i=1}^{n} \log\{f(x_i; \boldsymbol{\pi}, \boldsymbol{\theta})\}.$$

We will use ∇ and ∇^2 for the first and second differentiation operators with respect to $\boldsymbol{\pi}$, respectively.

The gradient vector of $f(x; \boldsymbol{\pi}, \boldsymbol{\theta})$ with respect to $\boldsymbol{\pi}$ is

$$\mathbf{s}(x; \boldsymbol{\pi}, \boldsymbol{\theta}) \equiv \nabla \log f(x; \boldsymbol{\pi}, \boldsymbol{\theta}) = \frac{(f(x; \theta_1), \ldots, f(x; \theta_m))^{\mathrm{T}}}{f(x; \boldsymbol{\pi}, \boldsymbol{\theta})}.$$

Let

$$\mathbf{d} = \mathbf{d}(\boldsymbol{\pi}, \boldsymbol{\theta}) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{s}(x_i; \boldsymbol{\pi}, \boldsymbol{\theta}),$$
$$\mathbf{D} = \mathbf{D}(\boldsymbol{\pi}, \boldsymbol{\theta}) = \mathrm{E}_{f(x; \boldsymbol{\pi}, \boldsymbol{\theta})}\{\mathbf{s}(x; \boldsymbol{\pi}, \boldsymbol{\theta})\mathbf{s}(x; \boldsymbol{\pi}, \boldsymbol{\theta})^{\mathrm{T}}\}.$$

Note that the vector $n(\mathbf{d} - \mathbf{1})$ contains the directional gradient values for all support points, as defined later in Eq. (16). Denote $\mathbf{s}(x) = \mathbf{s}(x; \boldsymbol{\pi}, \boldsymbol{\theta})$ and $f(x) = f(x; \boldsymbol{\pi}, \boldsymbol{\theta})$.

By noting $\mathrm{E}_f\{\mathbf{s}(x)\} = \mathbf{1}$ (not, as one may expect, $\mathbf{0}$) and $\mathbf{s}(x)^{\mathrm{T}}\boldsymbol{\pi} = 1$, we get

$$\boldsymbol{\pi}^{\mathrm{T}}\mathbf{d} = 1, \tag{3}$$

$$\mathbf{D}\boldsymbol{\pi} = \mathbf{1}. \tag{4}$$

Because

$$\nabla^2 f = f \nabla \log f \nabla \log f^{\mathrm{T}} + f \nabla^2 \log f$$

and, for mixing proportions, $\nabla^2 f$ is a matrix of zeros, we have

$$\mathbf{s}(x)\mathbf{s}(x)^{\mathrm{T}} = -\nabla^2 \log f(x; \boldsymbol{\pi}, \boldsymbol{\theta}),$$

or

$$\mathrm{E}_f\{\mathbf{s}(x)\mathbf{s}(x)^{\mathrm{T}}\} = -\mathrm{E}_f\{\nabla^2 \log f(x; \boldsymbol{\pi}, \boldsymbol{\theta})\}.$$

As $n \to \infty$,

$$\nabla^2 \ell = \sum_{i=1}^{n} \nabla^2 \log f(x_i; \boldsymbol{\pi}, \boldsymbol{\theta})$$

converges to its expectation, $-n\mathbf{D}$, if the expectation exists. Therefore, $n\mathbf{D}$, the expected Fisher information matrix for the log-likelihood function of $\boldsymbol{\pi}$ is a good approximation to $-\nabla^2\ell$ if the sample size is not too small.

Given $\boldsymbol{\pi}$, we can thus compute a new estimate $\boldsymbol{\pi}'$ by minimizing the following quadratic function of $\boldsymbol{\pi}'$:

$$-\mathbf{d}^{\mathrm{T}}(\boldsymbol{\pi}' - \boldsymbol{\pi}) + \frac{1}{2}(\boldsymbol{\pi}' - \boldsymbol{\pi})^{\mathrm{T}}\mathbf{D}(\boldsymbol{\pi}' - \boldsymbol{\pi}),$$

subject to $\boldsymbol{\pi}'^{\mathrm{T}}\mathbf{1} = 1$ and $\boldsymbol{\pi}' \geq \mathbf{0}$. Using (3), (4) and $\boldsymbol{\pi}'^{\mathrm{T}}\mathbf{1} = 1$, this is equivalent to minimizing

$$Q(\boldsymbol{\pi}') \equiv -\mathbf{d}^{\mathrm{T}}\boldsymbol{\pi}' + \frac{1}{2}\boldsymbol{\pi}'^{\mathrm{T}}\mathbf{D}\boldsymbol{\pi}', \tag{5}$$

under the same constraints. Readily we can get that

$$\nabla Q(\boldsymbol{\pi}') = -\mathbf{d} + \mathbf{D}\boldsymbol{\pi}'. \tag{6}$$

$$\nabla^2 Q(\boldsymbol{\pi}') = \mathbf{D}. \tag{7}$$

Thus

$$\nabla Q(\boldsymbol{\pi}')\big|_{\boldsymbol{\pi}'=\boldsymbol{\pi}} = -\mathbf{d} + \mathbf{1}.$$

Therefore, $-nQ(\boldsymbol{\pi}')$ is a quadratic function of $\boldsymbol{\pi}'$ and has the same directional gradient as the log-likelihood at $\boldsymbol{\pi}' = \boldsymbol{\pi}$ and a Hessian that is constant in $\boldsymbol{\pi}'$. To update the mixing proportions, we propose to solve the following quadratic programming problem:

$$\min_{\boldsymbol{\pi}'} \; Q(\boldsymbol{\pi}'), \quad \text{subject to } \; \boldsymbol{\pi}'^{\mathrm{T}}\mathbf{1} = 1, \boldsymbol{\pi}' \geq \mathbf{0}. \tag{8}$$

Solving problem (8) is equivalent to using the Fisher scoring method within a constrained parameter space, i.e., the probability simplex $\{\boldsymbol{\pi}' : \boldsymbol{\pi}'^{\mathrm{T}}\mathbf{1} = 1, \boldsymbol{\pi}' \geq \mathbf{0}\}$. This hence gives the name of the algorithm the constrained Fisher scoring; see Wang (2007b) for a framework development of this idea.

To ensure the monotonic increase of the log-likelihood after each iteration, the backtracking strategy can be adopted. Let $\boldsymbol{\pi} + \boldsymbol{\eta}$ be the solution to problem (8). Then find the smallest $k \in \{0, 1, 2, \dots\}$ so that the inequality

$$\ell(\boldsymbol{\pi} + \sigma^k \boldsymbol{\eta}, \boldsymbol{\theta}) \geq \ell(\boldsymbol{\pi}, \boldsymbol{\theta}) + \alpha \sigma^k n \mathbf{d}^{\mathrm{T}} \boldsymbol{\eta}, \quad 0 < \alpha < 1/2,$$

is satisfied, and the mixing proportion vector $\boldsymbol{\pi}$ is replaced by $\boldsymbol{\pi} + \sigma^k \boldsymbol{\eta}$. The numerical studies given in Sect. 5 use the popular step-halving, which sets $\sigma = \frac{1}{2}$.

3 Numerical strategies

In this section, we study some numerical strategies for minimizing $Q(\boldsymbol{\pi}')$. These strategies are critical for a successful implementation of the CFS method for computing the NPMLE.

3.1 Numerical evaluation of D

In order to solve problem (8), first of all we need to evaluate the expected information matrix $\mathbf{D}$, explicitly or implicitly. Although $\mathbf{D}$ does not have a closed-form expression here, it can be evaluated numerically. For a discrete $\mathcal{X}$, we have

$$\mathbf{D} = \sum_{z \in \mathcal{X}} f(z) \mathbf{s}(z) \mathbf{s}(z)^{\mathrm{T}},$$

and only those values in $\mathcal{X}$ that have a numerical effect on evaluating $\mathbf{D}$ should be included. The number of such values is usually quite small. For a continuous $\mathcal{X}$, consider a numerical integration rule, say, Simpson's rule. Let $a \equiv z_1 < \cdots < z_k \equiv b$ be equally spaced evaluation points over some interval $[a, b]$ with width $h = z_2 - z_1$. Let $c_1, \dots, c_k$ be the coefficients of the integration rule associated with these points; for Simpson's rule, they are $(1, 4, 2, 4, \dots, 2, 4, 1)/3$. Then given $\boldsymbol{\pi}$ and $\boldsymbol{\theta}$, we have

$$\mathbf{D} \approx \frac{h}{k} \sum_{i=1}^{k} c_i f(z_i; \boldsymbol{\pi}, \boldsymbol{\theta}) \mathbf{s}(z_i; \boldsymbol{\pi}, \boldsymbol{\theta}) \mathbf{s}(z_i; \boldsymbol{\pi}, \boldsymbol{\theta})^{\mathrm{T}}. \tag{9}$$

Alternatively, it is also possible to use a Gaussian quadrature rule, which needs much fewer number of evaluation points. Note that the added computational cost for each evaluation point is about the same as for an observed point.

Since **D** is needed by the constrained Fisher scoring method merely as to play a role of approximating $-\nabla^2 \ell / n$, it does not have to be accurately computed. In other words, **D** itself can be approximated. In the simulation studies presented in Sect. 5, we use Simpson's rule with only 101 evaluation points. From the viewpoint of numerical integration, using these few evaluation points is insufficient for an accurate evaluation of **D**, but for the purpose of optimization here, it makes little difference in terms of the number of iterations before convergence but saves the computational cost within each iteration.

3.2 Solution via decomposing **D**

The quadratic programming problem (8) can be turned into a least squares linear regression problem under the same constraints. To see this, let $\mathbf{D} = \mathbf{R}^{\mathrm{T}}\mathbf{R}$, where **R** can be obtained by, say, the Cholesky decomposition. Then minimizing $Q(\boldsymbol{\pi}')$ is equivalent to

$$\min_{\boldsymbol{\pi}'} \; ||\mathbf{R}\boldsymbol{\pi}' - \left(\mathbf{R}^{\mathrm{T}}\right)^{-1} \mathbf{d}||^2,$$

with the objective functions differing only in a constant. Since similar support points may be used, the matrix **D** is not bounded away from singularity and $(\mathbf{R}^{\mathrm{T}})^{-1}$ needs to be a generalized matrix inverse. Although the solution can usually be found by pivoting and choosing linearly independent variables from **D**, the ill-conditioned **D** can cause numerical instability and even make it fail to produce a valid **R**. We do not advocate this approach.

3.3 Solution without explicit evaluation of **D**

For the function $Q(\boldsymbol{\pi}')$, there is a nice, equivalent expression, thanks to the fact that $f(x; \boldsymbol{\pi}', \boldsymbol{\theta})$ is linear in $\boldsymbol{\pi}'$. Let us first consider a discrete $\mathcal{X}$, for which we have

$$Q(\boldsymbol{\pi}') = \frac{1}{2} \sum_{x \in \mathcal{X}} \frac{\{f_n(x) - f(x; \boldsymbol{\pi}', \boldsymbol{\theta})\}^2}{f(x; \boldsymbol{\pi}, \boldsymbol{\theta})} + C_1, \tag{10}$$

for some constant C_1, where $f_n(x)$ is the usual empirical mass function that allocates the probability mass $\frac{1}{n}$ to each x_i (not each x). The equivalence of $Q(\boldsymbol{\pi}')$ given by (10) to that given by (5) can be easily confirmed, by noticing that both are quadratic in $\boldsymbol{\pi}'$ and share the same gradient vector and Hessian matrix as given in (6) and (7), respectively.

For a continuous $\mathcal{X}$, the equivalent expression is similar, not much more than a mere replacement of the summation symbol in (10) with an integral one. What needs to be modified is the function $f_n(x)$, whose above definition does not make it a valid density function on a continuum. Therefore, let us use an approximation:

$$f_{n\epsilon}(x) = \begin{cases} \frac{1}{n\epsilon}, & \text{if } x \in \left[x_i - \frac{\epsilon}{2}, x_i + \frac{\epsilon}{2}\right], \\ 0, & \text{otherwise,} \end{cases}$$

where ϵ is an arbitrarily small value, say, 10^{-100}. In the above definition, for reason of simplicity we assume that there are no duplicated values in the data; if this is not true, $f_{n\epsilon}$ can be easily modified, by counting in all duplicated values in each interval. As a result, we define

$$Q_\epsilon(\boldsymbol{\pi}') = \frac{1}{2} \int\limits_{\mathcal{X}} \frac{\{f_{n\epsilon}(x) - f(x; \boldsymbol{\pi}', \boldsymbol{\theta})\}^2}{f(x; \boldsymbol{\pi}, \boldsymbol{\theta})} \mathrm{d}x + C_2, \tag{11}$$

for some constant C_2. It is easy to verify that for sufficiently small ϵ, $Q_\epsilon(\boldsymbol{\pi}')$ is almost identical to $Q(\boldsymbol{\pi}')$, in the sense that $Q_\epsilon(\boldsymbol{\pi}')$ is a quadratic function of $\boldsymbol{\pi}'$, which has, as $\epsilon \to 0$,

$$\nabla Q_\epsilon(\boldsymbol{\pi}') = -\mathbf{d} + \mathbf{D}\boldsymbol{\pi}', \tag{12}$$

$$\nabla^2 Q_\epsilon(\boldsymbol{\pi}') = \mathbf{D}. \tag{13}$$

In order to approximate the integral on the right-hand side of (11), let us consider using the same evaluation points $z_1, \ldots, z_k$ described in Sect. 3.1 and the observed values $x_1, \ldots, x_n$. Therefore,

$$Q_\epsilon(\boldsymbol{\pi}') \approx \frac{\epsilon}{2} \sum_{i=1}^{n} \frac{\left\{\frac{1}{n\epsilon} - f(x_i; \boldsymbol{\pi}', \boldsymbol{\theta})\right\}^2}{f(x_i; \boldsymbol{\pi}, \boldsymbol{\theta})} + \frac{h}{2} \sum_{i=1}^{k} \frac{c_i \left\{0 - f(z_i; \boldsymbol{\pi}', \boldsymbol{\theta})\right\}^2}{f(z_i; \boldsymbol{\pi}, \boldsymbol{\theta})} + C_2, \tag{14}$$

which, as $\epsilon \to 0$, has the same expressions for the gradient and Hessian. The matrix $\mathbf{D}$ is evaluated implicitly as if approximation (9) were used.

Given expressions (10) and (14), minimizing $Q(\boldsymbol{\pi}')$ is now turned into a least squares linear regression problem. An advantage of this is that the explicit evaluation of $\mathbf{D}$ is avoided and so is the associated numerical instability caused by similar support points, which exist inevitably for computing an NPMLE.

Note that the mixing proportion vector $\boldsymbol{\pi}$ can thus be updated by solving weighted least squares linear regression problems repeatedly. This results in an algorithm that is similar in spirit to the well-known iteratively reweighted least squares algorithm, see e.g., Nelder and Wedderburn (1972), Green (1984) and McCullagh and Nelder (1989), but it differs from the latter in that the design matrix contains not only the observed points but also some unobserved ones, such as $z_1, \ldots, z_k$ defined above. More precisely, it is really the whole sample space $\mathcal{X}$ involved, as in Eqs. (10) and (11). Therefore, we also call the algorithm "iteratively reweighted least integrated squares" (IRLIS). In practice, of course, a finite number of unobserved points is sufficient for the purpose of optimization.

3.4 The NNLSE solution

In aid of expression (10) or (14), minimizing $Q(\boldsymbol{\pi}')$ is a least squares linear regression problem with the linear equality constraint $\boldsymbol{\pi}'^{\mathrm{T}}\mathbf{1} = 1$ and the non-negativity constraint $\boldsymbol{\pi}' \geq \mathbf{0}$ (problem NNLSE). For this type of problem, Wang (2007c) adopts a strategy proposed by Haskell and Hanson (1981) to avoid the numerical instability caused by the equality constraint, which is crucial for finding an accurate NPMLE. For the CFS method here, the above NNLSE problem can be turned into a least squares linear regression problem under the non-negativity constraint only (problem NNLS):

$$\min_{\boldsymbol{\pi}'} |\boldsymbol{\pi}'^{\mathrm{T}}\mathbf{1} - 1|^2 + \gamma Q(\boldsymbol{\pi}'), \quad \text{subject to } \boldsymbol{\pi}' \geq \mathbf{0}, \tag{15}$$

for some small $\gamma > 0$; $\gamma = 2 \times 10^{-6}$ is used in the numerical studies given in Sect. 5. To satisfy the equality constraint, the solution to problem (15) is normalized afterward. We have found this strategy also working satisfactorily for the CFS method and adopted it in our implementation.

4 Computing the nonparametric maximum likelihood estimate

This section extends the constrained Fisher scoring method to the computation of the NPMLE, by incorporating the gradient function for locating new support points.

4.1 The gradient function

The gradient function is defined as the directional derivative of the log-likelihood at a mixing distribution G towards the degenerate mixing distribution δ_θ:

$$d(\theta; G) \equiv \left. \frac{\partial \ell\{(1-\epsilon)G + \epsilon\delta_\theta\}}{\partial \epsilon} \right|_{\epsilon=0} = \sum_{i=1}^{n} \left\{ \frac{f(x_i; \theta)}{f(x_i; G)} - 1 \right\}. \tag{16}$$

It is the instantaneous rate of change in the log-likelihood in the direction from one mixing distribution towards a point mass. A positive value of $d(\theta; G)$ indicates an ascent direction, in the sense that a sufficiently small step in this direction increases the value of the log-likelihood, while an negative value indicates the opposite. In order to distinguish from the gradient vector obtained by the operator ∇, throughout the paper we will always use the term “directional gradient” for a value of the gradient function.

The gradient function plays a critical role in NPMLE computation, in particular for locating new support points and for terminating an algorithm. It characterizes the NPMLE via the general equivalence theorem:

$$\widehat{G} \text{ maximizes } \ell(G) \iff \widehat{G} \text{ minimizes } \sup_{\theta}\{d(\theta; G)\} \iff \sup_{\theta}\{d(\theta; \widehat{G})\} = 0.$$

Since for any G,

$$\sup_{\theta}\{d(\theta; G)\} \geq \ell(\widehat{G}) - \ell(G),$$

the gradient function also provides an NPMLE algorithm with an ideal stopping criterion, which guarantees how inaccurate in the sense of likelihood the current estimate is, compared with the NPMLE. Lindsay (1995) provides an excellent exposition of these and other theoretical results.

4.2 The CFS method

With the CFS method developed above, we can easily extend it to NPMLE computation, as an analogous development to the CNM method. As the name suggests, it differs from CNM in whether the Fisher scoring or Newton's quadratic approximation is used for the log-likelihood function of $\boldsymbol{\pi}$. As in CNM, the gradient function is used for locating new support points.

Algorithm 1 (CFS) *Set $s = 0$. From an initial estimate G_0 with finite support and $\ell(G_0) > -\infty$, repeat the following steps:*

Step 1: compute all local maxima $\theta^*_{s1}, \ldots, \theta^*_{sp_s}$ of $d(\theta; G_s)$, $\theta \in \Omega$. If $\max_j d(\theta^*_{sj}, G_s) = 0$, stop.

Step 2: set $\boldsymbol{\theta}^+_s = (\boldsymbol{\theta}^{\mathrm{T}}_s, \theta^*_{s1}, \ldots, \theta^*_{sp_s})^{\mathrm{T}}$. Compute $\boldsymbol{\pi}^-_{s+1}$ by solving problem (8) and then a backtracking.

Step 3: discard all support points with zero entries in $\boldsymbol{\pi}^-_{s+1}$, which gives $\boldsymbol{\theta}_{s+1}$ and $\boldsymbol{\pi}_{s+1}$ of G_{s+1}. Set $s = s + 1$.

4.3 Convergence theorem

To establish the convergence of the CFS algorithm, we need the next two assumptions. Both assumptions are not difficult to verify in practice. The implications of Assumption 2 will be further discussed in Sect. 4.4. Denote by $\boldsymbol{\pi}^+_s$ the proportion vector expanded from $\boldsymbol{\pi}_s$ by including zeros for new support points. Let $\boldsymbol{\eta}_s = \boldsymbol{\pi}^-_{s+1} - \boldsymbol{\pi}^+_s$, $\mathbf{D}^+_s = \mathbf{D}(\boldsymbol{\pi}^+_s, \boldsymbol{\theta}^+_s)$ and $\mathcal{G}_0 = \{G : \ell(G) \geq \ell(G_0) > -\infty\}$. We may also drop the subscripts to indicate an arbitrary such expansion.

Assumption 1 $f(x; \theta)$ is bounded above for all $x \in \mathcal{X}$ and $\theta \in \Omega$.

Assumption 2 For all $G \in \mathcal{G}_0$ and all ascent directions $\boldsymbol{\eta}$, there exists a finite $V \in \mathbb{R}$ such that

$$\boldsymbol{\eta}^{\mathrm{T}}\mathbf{D}^+\boldsymbol{\eta} \leq V.$$

The following theorem establishes the convergence of the CFS method.

Theorem 1 *Under Assumptions 1 and 2, $\ell(G_s) \to \ell(\widehat{G})$ monotonically as $s \to \infty$ for any sequence $\{G_s\}$ created by Algorithm 1.*

The proof of this theorem is similar to that of Theorem 1 of Wang (2007c) and is omitted here.

4.4 Bounding the information matrix

Assumption 2 holds for most commonly used distributions $f(x; \theta)$, for example, the exponential family. Nevertheless, it may fail when a distribution has light or truncated tails, for example, the triangular distribution. Apart from the theoretical necessity for establishing the convergence, the failure of Assumption 2 also has a numerical consequence. An unbounded $\mathbf{D}$ can result that the constrained Fisher scoring solution is not sufficiently different from the current iterate, making the algorithm fail to converge.

The CFS algorithm can be easily modified to make it also work in such situations, by bounding the information matrix $\mathbf{D}$ as follows. For some sufficiently large yet finite $v > 0$, let $\bar{s}_j = \min\{s_j, v\}$, s_j being the jth element of $\mathbf{s}$, and $\bar{\mathbf{s}}(x) \equiv \bar{\mathbf{s}}(x; \boldsymbol{\pi}, \boldsymbol{\theta})$ be the vector consisting of all $\bar{s}_j$'s. Denote

$$\overline{\mathbf{D}} = \mathrm{E}_f \left\{ \bar{\mathbf{s}}(x)\bar{\mathbf{s}}(x)^{\mathrm{T}} \right\}.$$

This means that for $G \in \mathcal{G}_0$ and all directions $\boldsymbol{\eta}$,

$$\boldsymbol{\eta}^{\mathrm{T}}\overline{\mathbf{D}}^{+}\boldsymbol{\eta} \le v^2.$$

Therefore, the CFS algorithm can be modified to minimize the function $Q(\boldsymbol{\pi}')$ with $\mathbf{D}$ replaced with $\overline{\mathbf{D}}$. Its convergence can be established by Theorem 1.

5 Examples

This section presents two practical applications of NPMLE computation. The first example uses a practical data set with discrete values from Simar (1976), for which the Poisson mixture is considered. Four algorithms are applied in this example: ISDM (Lesperance and Kalbfleisch 1992), EM (Dempster et al. 1977; Laird 1978), CNM and CFS. The second example compares CFS with CNM in a simulation study following a design of Wang (2007c), which mimics a practical scenario in Efron (2004) and which involves the normal mixture with unit component variance. All algorithms are implemented in R and the computations were conducted on a PC with a Pentium 4 2.40 GHz CPU.

5.1 Example 1

Simar (1976) studied a data set, given in Table 1, about vehicle accident claims. It contains the frequency for each number of accident claims x_i out of 9461 policies

Table 1 The accident data

x_i	0	1	2	3	4	5	6	7
Frequency	7,480	1,317	239	42	14	4	4	1

submitted to an insurance company in a single year. Fitting the Poisson mixture to the data appears suitable, whose component distribution has the density

$$f(x;\theta) = e^{-\theta}\frac{\theta^x}{x!}, \quad \theta \geq 0, \;\; x \in \{0, 1, 2, \dots\}.$$

The NPMLE provides not only a much better fit than the unicomponent maximum likelihood estimate but also insights into the heterogeneity of the population. We note that the total number of observations is 9461 and the number of distinct ones is 8.

An accurate NPMLE can be easily found by either the CNM or CFS method, which is (after rounding):

$$\widehat{\boldsymbol{\theta}} = (0.00000, 0.23260, 0.35291, 2.56170)^{\mathrm{T}},$$
$$\widehat{\boldsymbol{\pi}} = (0.40998, 0.10488, 0.47665, 0.00849)^{\mathrm{T}}.$$

This solution is better than that provided by Simar (1976):

$$\widehat{\boldsymbol{\theta}} = (0.08854, 0.58020, 3.17606, 3.66871)^{\mathrm{T}},$$
$$\widehat{\boldsymbol{\pi}} = (0.75997, 0.23617, 0.00370, 0.00016)^{\mathrm{T}};$$

and than the improved solution found by Leroux (1992):

$$\widehat{\boldsymbol{\theta}} = (0.00000, 0.33554, 2.54498)^{\mathrm{T}},$$
$$\widehat{\boldsymbol{\pi}} = (0.41830, 0.57302, 0.00868)^{\mathrm{T}}.$$

Simar's and Leroux's solutions have the maximum directional gradients 9.04 and 0.0388, respectively, whereas the accurate NPMLE solution has a maximum directional gradient value less than 10^{-6}. Statistically speaking, such differences in the solutions are not necessarily significant, but they are indicative of the algorithmic efficiency. It can be easily found that the directional gradient values are very small over the interval $[0, 0.4]$, which implies an extremely flat likelihood surface in this area. This can cause difficulties for many NPMLE algorithms. CFS, however, like CNM, appears to be immune of the flat likelihood problem.

Table 2 contains the computational results for four NPMLE algorithms: ISDM, EM, CNM and CFS. All algorithms are started with the initial support points $\boldsymbol{\theta}_0 = (0.0, 0.5, \dots, 7.0)^{\mathrm{T}}$, which are allocated with equal masses. A grid of 200 points equally spaced between 0 and 7 is used for locating all local maxima of the gradient function; using 100 points appears insufficient. ISDM and EM converge extremely slowly because of the flat likelihood. Even though they are stopped prematurely at $\sup_\theta\{g(\theta; G_s)\} \leq 10^{-2}$, they still need tens of thousands of iterations. CFS and CNM, by contrast, are much faster, and competitive with each other. Both

Table 2 Performance of four NPMLE algorithms for the accident data

Algorithm	s	$\ell(\widehat{G}) - \ell(G_s)$	$\sup_\theta\{g(\theta; G_s)\}$	Time (s)
ISDM	92,882	5.39×10^{-3}	9.37×10^{-3}	691.59
EM	66,498	2.11×10^{-4}	1.00×10^{-2}	143.20
CNM	30	1.27×10^{-11}	1.58×10^{-7}	0.45
CFS	22	1.32×10^{-7}	5.75×10^{-7}	0.36

being stopped at $\sup_\theta\{g(\theta; G_s)\} \leq 10^{-6}$, the final iterate provided by CNM is more accurate than that by CFS, in the sense of $\ell(G_s)$ and $\sup_\theta\{g(\theta; G_s)\}$, indicating that CNM converges faster than CFS near the NPMLE. This is understandable, since Newton's method for updating $\boldsymbol{\pi}$ has a quadratic order of convergence while Fisher scoring is only of linear order. On the other hand, CFS takes fewer number of iterations and a shorter execution time than CNM, suggesting that CFS increases the likelihood faster than CNM during the early stage of the computation.

5.2 Example 2

In the simulation study of this example, we follow the design of Example 2 of Wang (2007c), which investigates the performance of the NPMLE algorithms used for Efron (2004) empirical Bayes plan of estimating a null hypothesis distribution from a large number of z-values. These z-values are obtained from large-scale, simultaneous hypotheses testing. In Efron's study, an 8-component normal mixture with unit component variance was obtained, as follows:

$$\widehat{\boldsymbol{\theta}} = (-10.9, -7.0, -4.9, -1.8, -1.1, 0.0, 2.4, 6.1)^{\mathrm{T}},$$
$$\widehat{\boldsymbol{\pi}} = (1.5, 1.3, 5.6, 12.3, 13.6, 60.8, 2.7, 2.2)^{\mathrm{T}}/100,$$

with θ being the component mean. For the simulation study here, 100 random samples, for the respective sample size 100 and 1,000, were generated from the above mixture model and each algorithm was employed to compute the NPMLE from each sample. Only CNM and CFS are used in this study. They were started by taking Efron's 8-component mixture as G_0 and stopped at $\sup_\theta\{d(\theta; G_s)\} \leq 10^{-5}$.

The results of the simulation study are given in Table 3, which include a five-number summary for, respectively, the number of iterations and execution times for each algorithm and each sample size. It can be seen that the CFS algorithm is slightly slower than CNM, but it is also very fast and stable, and competitive with CNM, with an increasing competitiveness with the sample size. This is understandable, since the expected and observed information matrices tend to be identical as the sample size approaches infinity.

Table 3 Five-number summaries of the number of iterations and execution times required by CNM and CFS over 100 simulated data sets in each case

n	Algorithm	Number of iterations					Time (s)				
		Min	Q1	Median	Q3	Max	Min	Q1	Median	Q3	Max
100	CNM	7	8	9	9	12	0.26	0.32	0.36	0.39	0.64
	CFS	9	12	13	16	35	0.37	0.52	0.59	0.78	1.71
1,000	CNM	7	8	9	10	15	3.20	3.63	3.93	4.47	6.93
	CFS	7	11	12	14	17	3.73	5.07	5.68	6.52	8.70

6 Summary and remarks

In previous sections, we have proposed a new method named the constrained Fisher scoring for computing the nonparametric maximum likelihood estimate of a mixing distribution. At each iteration, it approximates the log-likelihood function of the mixing proportions with the Fisher scoring quadratic approximations and updates all mixing proportions by solving the resulting quadratic programming problem within the probability simplex. At each iteration, new support points are found and included, as guided by the gradient function, and bad support points are discarded, after being found redundant by optimizing the quadratic approximation. It is generally competitive with the fast and stable CNM method.

The idea of using quadratic approximations for NPMLE computation has been studied previously by other researchers. Atwood (1976) considered the second-order Taylor series expansion of the log-likelihood and Wu (1978a,b) generalized the idea to include methods such as the gradient projection, the conjugate gradient projection, Newton's and the quasi-Newton. Unfortunately, these previous work, owing to the difficulties in both theory and algorithm, requires that the Hessians of the quadratic approximations to the log-likelihood be negative definite and bounded away from singularity. This restriction implies that similar support points can not be used, and thus makes it difficult to find an accurate NPMLE solution, not only because the NPMLE may contain similar support points, but also because support points often need to be replaced, sometimes partially, by similar support points to increase the likelihood value. The developments of CNM and CFS have resolved this issue in both theory and algorithm, to such an extent that even identical support points can be included in the support set.

In order to cope with the explosive increase in the number of support points, some methods, such as VDM (Wynn 1970; Fedorov 1972), ISDM (Lesperance and Kalbfleisch 1992) and SIP (Coope and Watson 1985; Lesperance and Kalbfleisch 1992), can be implemented without storing the support points by updating the likelihood vector, i.e., the vector of the mixture density values at the distinct observed points. In principle, once the computation is finished, the support set can be recovered from the final gradient function and the mixing proportions from solving a set of linear equations. In practice, however, such recovered NPMLEs may have significant reduction in the likelihood value, especially when the likelihood is flat, as in Example 1. By contrast, CNM and CFS always store the support set, which expands and shrinks in an adaptive fashion, and are not subject to any loss in the achieved likelihood value. Since redundant support points are quickly found and discarded, the support set maintained by both algorithms throughout the computation is usually quite small, typically less than twice the number of support points in the NPMLE (for computation within the limit of numerical precision).

The proposed CFS algorithm provides a new tool in the toolkit for NPMLE computation. Further research will look into possible extensions of both CNM and CFS to the maximum likelihood computation of a nonparametric mixing distribution with penalty or under constraint (Susko et al. 1998; Wang and Lindsay 2005; Wang 2007a) and of semiparametric mixtures (Böhning 1995; Lindsay and Lesperance 1995; Lindsay 1995). In such situations, we wonder whether CFS variants may have an advantage over CNM-type ones. Since $\nabla^2 \ell = -\mathbf{S}^{\mathrm{T}}\mathbf{S}$, Newton's method happens to be identical to the Gauss-Newton method for computing mixing proportions and hence is easily applicable, but it may have difficulties in other situations as the method usually has. Wang (2007b) investigated both the Fisher-scoring and Gauss–Newton methods for maximum likelihood computation of finite mixtures and has found that the Fisher-scoring method is much better than Gauss–Newton. The Gauss-Newton method may perform extremely badly for small-sized samples or in the cases of a similar nature, while Fisher-scoring does not.

Acknowledgements The author would like to thank two anonymous referees for helpful suggestions.

References

Atwood CL (1976) Convergent design sequences, for sufficiently regular optimality criteria. Ann Stat 4:1124–1138

Böhning D (1985) Numerical estimation of a probability measure. J Stat Plan Inference 11:57–69

Böhning D (1995) A review of reliable algorithms for the semi-parametric maximum likelihood estimator of a mixture distribution. J Stat Plan. Inference 47:5–28

Böhning D (2000) Computer-assisted analysis of mixtures and applications: meta-analysis, disease mapping, and others. Chapman & Hall/CRC, Boca Raton

Coope ID, Watson GA (1985) A projected Lagrangian algorithm for semi-infinite programming. Math Program 32:337–356

Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. J Roy Stat Soc Ser B 39:1–22

Efron B (2004) Large-scale simultaneous hypothesis testing: the choice of a null hypothesis. J Am Stat Assoc. 99:96–104

Fedorov VV (1972) Theory of optimal experiments. Academic Press, New York

Green PJ (1984) Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives. J Roy Stat Soc Ser B 46:149–192

Haskell KH, Hanson RJ (1981) An algorithm for linear least squares problems with equality and nonnegativity constraints. Math Program 21:98–118

Kiefer J, Wolfowitz J (1956) Consistency of the maximum likelihood estimator in the presence of infinitely many incidental parameters. Ann Math Stat 27:886–906

Laird NM (1978) Nonparametric maximum likelihood estimation of a mixing distribution. J Am Stat Assoc 73:805–811

Leroux BG (1992) Consistent estimation of a mixing distribution. Ann Stat 20:1350–1360

Lesperance ML, Kalbfleisch JD (1992) An algorithm for computing the nonparametric MLE of a mixing distribution. J Am Stat Assoc 87:120–126

Lindsay BG (1983) The geometry of mixture likelihoods: a general theory. Ann Stat 11:86–94

Lindsay BG (1995) Mixture models: theory, geometry and applications, volume 5 of NSF-CBMS regional conference series in probability and statistics. Institute for Mathematical Statistics, Hayward

Lindsay BG, Lesperance ML (1995) A review of semiparametric mixture models. J Stat Plan Inference 47:29–39

McCullagh P, Nelder JA (1989) Generalized linear models, 2nd edn. Chapman & Hall, Boca Raton

Nelder JA, Wedderburn RWM (1972) Generalized linear models. J Roy Stat Soc Ser A 135:370–384

Simar L (1976) Maximum likelihood estimation of a compound Poisson process. Ann Stat 6:1200–1209

Susko E, Kalbfleisch J, Chen J (1998) Constrained nonparametric maximum-likelihood estimation for mixture models. Can J Stat 26:601–617

Wang J-P (2007a) A linearization procedure and a VDM/ECM algorithm for penalized and constrained nonparametric maximum likelihood estimation for mixture models. Comp Stat Data Anal 51:2946–2957

Wang Y (2007b) Maximum likelihood computation based on the Fisher scoring and Gauss–Newton quadratic approximations. Comp Stat Data Anal 51:3776–3787

Wang Y (2007c) On fast computation of the non-parametric maximum likelihood estimate of a mixing distribution. J Roy Stat Soc Ser B 69:185–198

Wang J-P, Lindsay BG (2005) Penalized nonparametric maximum likelihood approach to species richness estimation. J Am Stat Assoc 100:942–959

Wu CF (1978) Some algorithmic aspects of the theory of optimal designs. Ann Stat 6:1286–1301

Wu CF (1978) Some iterative procedures for generating nonsingular optimal designs. Commun Stat Theory Methods 7:1399–1412

Wynn HP (1970) The sequential generation of D-optimal experimental design. Ann Math Stat 41:1655–1664