西南财经大学

Southwestern University of Finance and Economics

2017 级

第一学期论文（设计）

论文题目：MM Algorithm for Some Statistical Problems

学生姓名：刘　伟

所在学院：统计学院

专　　业：统计学

学　　号：117020208005

指导教师：林华珍

成　　绩：

2019 年 3 月

MM Algorithm for Some Statistical Problems

Wei Liu

Center of Statistical Research and School of Statistics,

Southwestern University of Finance and Economics, Chengdu, Sichuan, China

SUMMARY. With the complexity of data structure, statistical models become more and more complex. Accordingly, the objective function of solving the model also becomes more complex. Two major challenges complicate the statistical computation. First, the objective function is nonconvex, so its solution is not unique and common algorithm only can get local optimal solution but not global optimal solution. Second, the derivative of the objective function does not exist, so the Newton-Raphson iteration can not be directly applied to it and need to turn to some new algorithm. In this situation, MM algorithm emerges. It is increasingly common to apply MM algorithm to solving optimization problems in statistical area. The basic principle and common technique of MM algorithm are introduced in this manuscript. Some common statistical problems are solved by MM algorithm and good performance is illustrated by some simulations.

KEY WORDS: MM algorithm; Complex model; Global solution.

1 Introduction

In the last decade or so, MM algorithm has appeared in many statistical research (Hunter and Lange, 2000, 2002; Hunter, 2004; Hunter and Li, 2002, 2005; Nguyen, 2016; Sabatti and Lange, 2002). Actually, MM algorithm is not a specific algorithm, but a principle for constructing optimization algorithms. An MM algorithm operates by creating a surrogate function that minorizes or majorizes the objective function. When the surrogate function is optimized, the objective function is driven uphill or downhill as needed. In minimization problem, MM stands for majorize/minimize, and in maximization problem, MM stands for minorize/maximize. And the EM algorithm from statistics is a special case of MM

algorithm. As far as we know, Ortega and Rheinboldt (1970) first proposed the MM principle in the context of line search methods. Later, De Leeuw and Heiser (1977) presented a MM algorithm for multidimensional scaling contemporary with the classic paper on EM algorithms (Dempster et al., 1977). For more information, historical developments in MM algorithms is presented in the tutorial on MM algorithms (Hunter and Lange, 2004).

The remainder of this paper is organized as follows. The idea of MM algorithm is presented in Section 2. The Section 3 presents the specific construction method of MM algorithms for some typical statistical problems. Then the simulation results are reported in Section 4. Finally, we conclude this paper with brief remarks in Section 5. In addition, we implement our proposed algorithms in an efficient and user-friendly Matlab toolbox called MM, which is available at `https://github.com/feiyoung/MM-algorithm`.

2 Idea of MM Algorithm

In this section, we mainly introduce the basic principles and common techniques on MM algorithm.

2.1 Basic Principle

As for minimization problem $\min_x f(x)$, we usually need to find a majorization function $g(x|x^{(k)})$ such that for any $x \in R^p$

$$\begin{aligned} f(x) &\leqslant g(x|x^{(k)}), && (2.1)\\ f(x^{(k)}) &= g(x^{(k)}|x^{(k)}), && (2.2) \end{aligned}$$

where $x^{(k)}$ is the value of x in k-th iteration. Under these conditions, we can convert the optimization problem $\min f(x)$ to the optimization of $x^{(k+1)} = arg\min g(x|x^{(k)})$, from which we obtain

$$g(x^{(k+1)}|x^{(k)}) \leqslant g(x|x^{(k)}). \quad (2.3)$$

Thus, combined with the conditions, we have

$$\begin{aligned} f(x^{(k+1)}) &\leqslant g(x^{(k+1)}|x^{(k)}) \\ &\leqslant g(x^{(k)}|x^{(k)}) \\ &= f(x^{(k)}), \end{aligned}$$

where the first inequality is followed by (2.1), the second inequality is followed by (2.3), the last equality is followed by (2.2). This result ensures the algorithm implemented in the direction of decreasing the objective function f, so it is valid.

Correspondingly, for maximization problem $\max_x f(x)$, we usually want to find a minorization function $g(x|x^{(k)})$ such that for any $x \in R^p$

$$f(x) \geqslant g(x|x^{(k)}), \tag{2.4}$$

$$f(x^{(k)}) = g(x^{(k)}|x^{(k)}). \tag{2.5}$$

2.2 Common Technique

There are some general methods of constructing optimization algorithm by the principle of MM algorithm. We mainly introduce six methods, including Jensen's inequality based method, supporting hyperplane property based method, quadratic upper bound principle based method, arithmetic-geometric mean inequality based method, Cauchy-Schwartz inequality based method and upper bounded function matrix of second derivative function matrix based method.

Jensen's inequality For minimization problem, assume $f(x)$ is a convex function on $x = (x_1, \cdots, x_p)' \in R^p$, then

$$f(x) = f(\sum_{j=1}^{p} \omega_j x_j) \leqslant \sum_{j=1}^{p} \omega_j f(x_j) = g(x|x^{(k)})$$

when $x = x^{(k)}$, we have $\omega_j = \omega_k$.

For maximization problem, if $f(x)$ is concave, then

$$f(x) = f(\sum_{j=1}^{p} \omega_j x_j) \geqslant \sum_{j=1}^{p} \omega_j f(x_j) = g(x|x^{(k)})$$

when $x = x^{(k)}$, we have $\omega_j = \omega_k$.

Next, we consider the function generated by the composition of a concave function $h(x)$

and a linear function $\langle l, x\rangle$. Assume $f(x) = h(\langle l, x\rangle)$. By concavity of h, we have

$$\begin{aligned} f(x) &= h(\langle l, x\rangle) \\ &= f(\sum_{j=1}^{p} \alpha_{ij}^{(k)} \frac{x_j}{x_j^{(k)}} \langle l, x^{(k)}\rangle) \\ &\geqslant \sum_{j=1}^{p} \alpha_{ij}^{(k)} f(\frac{x_j}{x_j^{(k)}} \langle l, x^{(k)}\rangle) \\ &= g(x|x^{(k)}), \end{aligned}$$

where $\alpha_{ij}^{(k)} = \frac{l_j x_j^{(k)}}{\langle l, x^{(k)}\rangle}$. Thus, we get the separation of each element of x, which leads to EM algorithm. We consider

$$\max_x f(x) = \max_x log \int_Z exp\{u(x, z)\} dz,$$

which is the maximum likelihood estimation of exponential family model. Noting

$$\begin{aligned} f(x) - f(y) &= log \frac{\int_Z exp\{u(y, z)\} \frac{exp\{u(x,z)\}}{exp\{u(y,z)\}} dz}{\int_Z exp\{u(y, z)\} dz} \\ &\geqslant \frac{\int_Z exp\{u(y, z)\} \log \frac{exp\{u(x,z)\}}{exp\{u(y,z)\}} dz}{\int_Z exp\{u(y, z)\} dz}, \end{aligned}$$

where the last inequality is followed by Jensen's inequality. In the following, we obtain

$$f(x) \geqslant g(x|x^{(k)}) = f(x^{(k)}) + \frac{\int_Z exp\{u(x^{(k)}, z)\} \log \frac{exp\{u(x,z)\}}{exp\{u(x^{(k)},z)\}} dz}{\int_Z exp\{u(x^{(k)}, z)\} dz}.$$

Finally, we give the specific derivation of the last inequality. Let $h(y, z) = exp\{u(x, z)\}$, $f(z) = \frac{h(y,z)}{\int_Z h(y,z) dz}$, then

$$\begin{aligned} \log(E_Z(\frac{h(x, z)}{h(y, z)})) &= \log \int_Z \frac{h(x, z)}{h(y, z)} f(z) dz \\ &\geqslant E_Z(\log \frac{h(x, z)}{h(y, z)}) \\ &= \int_Z \log \frac{h(x, z)}{h(y, z)} f(z) dz \\ &= \int_Z \log \frac{h(x, z)}{h(y, z)} \frac{h(y, z)}{\int_Z h(y, z) dz} dz \\ &= \frac{\int_Z \log \frac{h(x,z)}{h(y,z)} h(y, z) dz}{\int_Z h(y, z) dz}. \end{aligned}$$

Supporting hyperplane property The so-called supporting hyperplane property is that if $f(x)$ is a convex function, then we have

$$f(x) \geqslant \Delta f(x^{(k)})^T(x - x^{(k)}) + f(x^{(k)}), \forall x^{(k)} \in R^p,$$

where $\Delta f(x)$ is the derivative function of $f(x)$. We give two typical examples here. The first specific example is

$$ln(1 - x_1x_2) \geqslant ln(1 - x_1^{(k)}x_2^{(k)}) - \frac{1}{1 - x_1^{(k)}x_2^{(k)}}(x_1x_2 - x_1^{(k)}x_2^{(k)}),$$

where $x_1, x_2 > 0$. In addition, separation can be achieved by invoking

$$-x_1x_2 \geqslant \frac{-1}{2}(\frac{x_2^{(k)}x_1^2}{x_1^{(k)}} + \frac{x_1^{(k)}x_2^2}{x_2^{(k)}}),$$

Thus, we have

$$g(x|x^{(k)}) = ln(1 - x_1^{(k)}x_2^{(k)}) - \frac{1}{1 - x_1^{(k)}x_2^{(k)}}(\frac{1}{2}(\frac{x_2^{(k)}x_1^2}{x_1^{(k)}} + \frac{x_1^{(k)}x_2^2}{x_2^{(k)}}) - x_1^{(k)}x_2^{(k)}),$$

When $f(x)$ is concave, we have

$$f(x) \leqslant \Delta f(x^{(k)})^T(x - x^{(k)}) + f(x^{(k)}),$$

The second example is the C-D algorithm. C-D algorithm is also a special case induced by support hyperplane property. We consider

$$\min_x f(x) = \min_x f_1(x) + f_2(x),$$

where $f_1(x)$ is convex, $f_2(x)$ is concave. Then

$$f(x) \leqslant f_1(x) + f_2'(x^{(k)})(x - x^{(k)}) + f_2(x^{(k)}) = g(x|x^{(k)}),$$

where $g(x|x^{(k)})$ is the majorization function.

Quadratic upper bound principle If $f(x) = |x|$, then we have

$$f(x) \leqslant \frac{1}{2}\frac{x^2}{x^{(k)}} + \frac{1}{2}|x^{(k)}| = g(x|x^{(k)}),$$

which is followed by

$$\begin{aligned} |x| &= \frac{\sqrt{x^2x^{2(k)}}}{|x^{(k)}|} \\ &\leqslant \frac{x^2 + x^{2(k)}}{2|x^{(k)}|} \\ &= \frac{1}{2}\frac{x^2}{|x^{(k)}|} + \frac{1}{2}|x^{(k)}|, \end{aligned}$$

where the second inequality is followed by arithmetic-geometric mean inequality. We can extend to the function of $f(x) = |h(x)|$,

$$f(x) \leqslant \frac{1}{2}\frac{h(x)^2}{|h(x)^{(k)}|} + \frac{1}{2}|h(x)^{(k)}| = g(x|x^{(k)}).$$

Arithmetic-geometric mean inequality If $f(x) = \sqrt{x_1 x_2}$, then we have

$$\begin{aligned} f(x) &= \sqrt{x_1 x_1^{(k)} x_2 x_2^{(k)}} / \sqrt{x_1^{(k)} x_2^{(k)}} \\ &\leqslant \frac{x_2^{(k)} x_1 + x_1^{(k)} x_2}{2\sqrt{x_1^{(k)} x_2^{(k)}}} = g(x|x^{(k)}), \end{aligned}$$

where $x_1, x_2 >= 0$. Using the fact that $(a+b)^2 \leqslant 2(a^2+b^2)$, we obtain

$$\tilde{f}(x) = x_1 x_2 \leqslant \frac{1}{2}(\frac{x_2^{(k)} x_1^2}{x_1^{(k)}} + \frac{x_1^{(k)} x_2^2}{x_2^{(k)}}) = \tilde{g}(x|x^{(k)}).$$

Assume $\|.\|$ is a norm, then

$$\|x\| \leqslant \frac{1}{2\|x^{(k)}\|}(\|x\|^2 + \|x^{(k)}\|^2).$$

Especially, if $\|x\| = \sqrt{x'Ax}$, then

$$\sqrt{x'Ax} \leqslant \frac{1}{2\|x^{(k)}\|}(x'Ax + x'^{(k)}Ax^{(k)}),$$

A is a positive semi-definite matrix.

Cauchy-Schwartz inequality Assume $d(.,.)$ is a metric on $x \in \mathcal{X}$,then

$$d(x, x^{(k)})^2 \leqslant d(x,x)d(x^{(k)}, x^{(k)}).$$

Thus, we have

$$\frac{d(x, x^{(k)})^2}{d(x^{(k)}, x^{(k)})} \leqslant d(x,x) = \|x\|.$$

Such as $d(x, x^{(k)}) = \sqrt{x'Ax^{(k)}}$, A is a positive semi-definite matrix. Particularly, if $A = I$, d is Euclidean distance.

Upper bounded function matrix of second derivative function matrix Assume $f(x)$ is convex. If we use support hyperplane property getting the majorization function which is a linear function on x, then we need to use the second derivative matrix, a positive definite matrix. By the Taylor expansion, we have

$$f(x) = f(x^{(k)}) + \Delta f(x^{(k)})^T(x - x^{(k)}) + (x - x^{(k)})^T \Delta^2 f(\xi)(x - x^{(k)}).$$

Further, if we can find a positive matrix B which satisfies $B \geqslant \Delta^2 f(x) \forall x$, then we can obtain a majorization function,

$$g(x|x^{(k)}) = f(x^{(k)}) + \Delta f(x^{(k)})^T (x - x^{(k)}) + (x - x^{(k)})^T B (x - x^{(k)}).$$

It is worthwhile that the majorization function holds for any minimized function.

3 Some Statistical Problems

In this section, we apply the principle of MM algorithm to solve some common statistical problems. Here, we only present the specific content on lasso regression, group lasso regression, least absolute deviation regression and logistic regression. Of course, MM algorithm has many other applications in statistical computation not involved here.

3.1 Linear Regression

Given observation sample $(Y_i, X_i), i = 1, \ldots, n$, from a population $(Y, X), Y \in R, X \in R^p$, we consider the regression relationship of Y and X. For linear regression model

$$Y_i = X_i^T \beta_0 + \varepsilon_i, i = 1, \ldots, n, \tag{3.1}$$

where $\beta_0 = (\beta_{0,1}, \cdots, \beta_{0,p})'$ and ε_i is the error term.

3.1.1 MM Algorithm of Lasso Regression

For variable selection, we assume only small subset of X is related with Y, that is sparse. And we minimize the following objective function to obtain sparse solution,

$$f(\beta) = \|Y - X\beta\|_2^2 + \lambda \|\beta\|_1.$$

By the quadratic upper bound principle, we have

$$|\beta_j| \leqslant \frac{\beta_j^2}{2|\beta_j^{(k)}|} + \frac{1}{2}|\beta_j^{(k)}|,$$

which satisfies the majorization function conditions. Then, our surrogate objective function is

$$g(\beta|\beta^{(k)}) = \|Y - X\beta\|_2^2 + \lambda \sum_{j=1}^{p} (\frac{\beta_j^2}{2|\beta_j^{(k)}|} + \frac{1}{2}|\beta_j^{(k)}|).$$

Doing partial derivative w.r.t. β_j, we obtain

$$\begin{aligned} \frac{\partial g(\beta|\beta^{(k)})}{\partial \beta_j} &= -2\sum_{i=1}^{n} x_{ij}y_i + 2\sum_{i=1}^{n} x_{ij}x_i^T\beta + \lambda\frac{\beta_j}{|\beta_j^{(k)}|} \\ &= -c_j + \sum_{k=1}^{p} a_{jk}\beta_k + d_j\beta_j, j = 1, \ldots, p \\ &= 0, \end{aligned}$$

where $c_j = 2\sum_{i=1}^{n} x_{ij}y_i, a_{jk} = 2\sum_{i=1}^{n} x_{ij}x_{ik} = 2X_j^T X_k, d_j = \frac{\lambda}{|\beta_j^{(k)}|}$. We rewrite with matrix notation as follows.

$$(2X^T X + \Lambda(\beta^{(k)}))\beta = C,$$

where $X = (X_1^T, \ldots, X_n^T)^T, \Lambda(\beta^{(k)})) = diag(d_1, \ldots, d_p), C = (c_1, \ldots, c_p)^T$. Thus, we can easily get the close form solution of $(k+1)th$ iteration,

$$\beta^{(k+1)} = (2X^T X + \Lambda(\beta^{(k)})))^{-1}C.$$

When p is not large, such as $p \leqslant 2000$, we can conduct the iteration rapidly. However, when p is large($p \gg n$), the $p \times p$ matrix inversion is not available. Fortunately, with the help of Matrix Inverse Lemma, we can get the inversion by solve a $n \times n$ matrix.

Lemma 1. *For any matrices $A \in R^{p\times p}, B \in R^{p\times n}, C \in R^{n\times n}, D \in R^{n\times p}$, we have*

$$(A + BCD)^{-1} = A^{-1} - A^{-1}D^T(DA^{-1}B + C^{-1})^{-1}DA^{-1},$$

if all the inverses exist.

For simplicity, we write $\Lambda(\beta^{(k)}))$ as Λ. By Lemma 1, we have

$$(2X^T X + \Lambda)^{-1} = \Lambda^{-1} + \Lambda^{-1}X^T(X\Lambda^{-1}X^T + \frac{1}{2}I_n)^{-1}X\Lambda^{-1}$$

where $\Lambda^{-1} = diag(\frac{|\beta_1|}{\lambda}, \ldots, \frac{|\beta_p|}{\lambda})$ by the diagonal property. So we only need to evaluate an inversion of $n \times n$ matrix, $(X\Lambda^{-1}X^T + \frac{1}{2}I_n)^{-1}$.

3.1.2 Group Lasso Regression

Assume $X_j \in R^{p_j}$ is a group of same type covariates, whose regression coefficients have same sparsity, where $j = 1, \ldots, d$. Thus, we need to conduct group variables selection. Denote sample as $Y = (y_1, \ldots, y_n)', X_i = (x'_{i1}, \ldots, x'_{id}), X = (X_1^T, \ldots, X_n^T)^T, \beta_j = (\beta_{j1}, \ldots, \beta_{jp_j})', \beta = (\beta'_1, \ldots, \beta'_d)'$. Given the following objective function,

$$\min_{\beta} f(\beta) = \|Y - X\beta\|_2^2 + \lambda \sum_{j=1}^{d} (\sum_{k=1}^{p_j} \beta_{jk}^2)^{1/2}$$

By Cauchy-Switchz inequality, we obtain

$$f(\beta) \leqslant \|Y - X\beta\|_2^2 + \frac{1}{2}\lambda \sum_{j=1}^{d} (\|\beta_j\|_2^2 + \|\beta_j^{(k)}\|_2^2)/\|\beta_j^{(k)}\|_2 = g(\beta|\beta^{(k)}).$$

Similarly, we can get the iterative estimating equation with matrix form,

$$(2X^T X + \Lambda(\beta^{(k)}))\beta = C,$$

where $\Lambda(\beta^{(k)})) = diag(b_1^{(k)} I_{p_1}, \ldots, b_d^{(k)} I_{p_d}), C = (c_1, \ldots, c_d)^T, b_j^{(k)} = \frac{\lambda}{\|\beta_j^{(k)}\|_2}, c_j = 2\sum_{i=1}^{n} X'_{ij} y_i, C = 2X'y$. Thus, we can easily get the close form solution of $(k+1)th$ iteration.

$$\beta^{(k+1)} = (2X^T X + \Lambda(\beta^{(k)})))^{-1} C.$$

3.1.3 Least Absolute Deviation Regression

Under the same model (3.1), we solve the following minimization problem to carry out least absolute deviation regression,

$$\min_{\beta} f(\beta) = \min_{\beta} |Y_i - X_i^T \beta|.$$

Using the same technique as Lasso, we obtain

$$f(\beta) \leqslant \sum_{i=1}^{n} [\frac{(Y_i - X_i^T \beta)^2}{2|Y_i - X_i^T \beta^{(k)}|} + \frac{1}{2}|Y_i - X_i^T \beta^{(k)}|] = g(\beta|\beta^{(k)})$$

Further, we have

$$\frac{\partial g}{\partial \beta_j} = \sum_{i=1}^{n} \frac{X_{ij} X_i^T \beta}{|Y_i - X_i^T \beta^{(k)}|} - \sum_{i=1}^{n} \frac{Y_i X_{ij}}{|Y_i - X_i^T \beta^{(k)}|} = 0, j = 1, \ldots, p$$

For simplicity, we use the notation of matrix. And denote $h_{jk} = \sum_{i=1}^{n} \frac{X_{ij}X_{ik}}{|Y_i - X_i^T\beta^{(k)}|}, c_j = \sum_{i=1}^{n} \frac{Y_i X_{ij}}{|Y_i - X_i^T\beta^{(k)}|}, H(\beta^{(k)}) = (h_{j,k})_{j,k=1,\ldots,p}, C(\beta^{(k)}) = (c_1, \ldots, c_p)^T$, then we have

$$H(\beta^{(k)})\beta = C(\beta^{(k)}).$$

Thus, we obtain

$$\beta^{(k+1)} = H^{-1}C(\beta^{(k)}).$$

Here, we need to note a small trick. Because $\beta^{(k)}$ is in the denominator in the iteration process, the algorithm is not stable. To solve this problem, we replace $|Y_i - X_i^T\beta^{(k)}|$ with $|Y_i - X_i^T\beta^{(k)}| + \epsilon$, where, typically, $\epsilon = 1e-6$.

3.2 Generalized Linear Regression

Given observation sample $(Y_i, X_i), i = 1, \ldots, n$ from a population $(Y, X), Y \in R, X \in R^p$, we consider the regression relationship of Y and X. We consider the generalized linear regression model

$$h(\mathbf{E}(Y|X)) = X'\beta, \tag{3.2}$$

where $h(\cdot)$ is a link function. We only take the logistic regression as the example, and the other generalized linear regression is similar.

3.2.1 MM Algorithm of Logistic Regression

When $h(t) = \ln(\frac{t}{1-t})$, it is logistic model. And corresponding negative log likelihood function is

$$f(\beta) = -l(\beta) = \sum_{i=1}^{n} \ln(1 + e^{X_i^T\beta}) - \sum_{i=1}^{n} Y_i X_i^T\beta.$$

Let $h(t) = \ln(1 + e^t)$, then it is obvious that $h'(t) = \frac{e^t}{1+e^t}, 1/4 \geqslant h''(t) = \frac{e^t}{(1+e^t)^2} \geqslant 0$. Thus, $h(t)$ is convex. By Taylor expansion and the upper bound of $h''(t)$, we have

$$h(t) \leqslant h'(t^{(k)})(t - t^{(k)}) + h(t^{(k)}) + (t - t^{(k)})^2.$$

Thus, we obtain the optimization problem

$$\min_{\beta} g(\beta|\beta^{(k)}) = \sum_{i=1}^{n} [h_i^{(k)} X_i^T\beta + (X_i^T\beta - X_i^T\beta^{(k)})^2 - Y_i X_i^T\beta]$$

which is just a quadratic program, where $h_i^{(k)} = h'(X_i^T\beta^{(k)})$. We can easily obtain the close form solution of iteration.

$$\beta^{(k+1)} = S(\beta^{(k)})^{-1}C(\beta^{(k)}).$$

where the $(j,r)th$ entry of $S(\beta^{(k)}), s_{jr} = \sum_{i=1}^{n} 2X_{ij}X_{ir}$, the jth entry of $C(\beta^{(k)})$, $c_j = \sum_{i=1}^{n}(Y_i + 2X_i^T\beta^{(k)} - h_i^{(k)})X_{ij}$. We briefly denote this MM algorithm as **Convexsupport**.

In addition, we introduce another construction method, which is called **Upbound**. By $f(\beta)$, we have

$$\Delta^2 f(\beta) = \sum_{i=1}^{n} h(X_i^T\beta)(1 - h(X_i^T\beta))X_iX_i^T.$$

Since $h(X_i^T\beta)(1 - h(X_i^T\beta)) \leqslant 1/4$, we may define the positive matrix $B = \frac{1}{4}X^TX$ and conclude that $B - \Delta^2 f(\beta)$ is positive definite matrix. Thus

$$g(\beta|\beta^{(k)}) = f(\beta^{(k)}) + \Delta f(\beta^{(k)})^T(\beta - \beta^{(k)}) + 1/2(\beta - \beta^{(k)})^T B(\beta - \beta^{(k)}).$$

Finally, we obtain

$$\beta^{(k+1)} = \beta^{(k)} + B^{-1}\Delta f(\beta^{(k)}),$$

where $\Delta f(\beta^{(k)}) = X^T(Y - h(X\beta^{(k)}))$.

4 Simulation Study

In this section, to investigate the performance of the proposed algorithm, we consider three examples corresponding to lasso regression, least absolute deviation regression and logistic regression. Because the algorithm of group lasso regression is very similar to lasso regression, we omit the example about it.

4.1 Example 1: Lasso Regression

We consider the specific form of model (3.1). $X_i \in R^p$ is generated from multivariate normal distribution with mean 0 and covariance matrix $\Sigma = (\sigma_{jk}) = (0.5^{|j-k|}), j, k = 1, \cdots, p$. Sample size $n = 300, 500$, dimension of covariates $p = 50, 100$, $\varepsilon_i \sim N(0,1), \beta = (1, -1, 1, -1, 1, 0, \cdots, 0)^T$ is sparse regression coefficients, namely, only the first five components of β are not zero.

We use TPR (sensitivity), TNR (specificity) and F-measure (Powers, 2011) to assess the performance of variable selection, where F-measure is defined as aggregation of TPR and TNR,

$$F\text{-}measure = 2 \times \frac{TPR \times TNR}{TPR + TNR}.$$

TPR represents the proportion of the number of important variables identified by our algorithm being true important variables to the number of true important variables. If TPR is closer to 1, it indicates that our algorithm can select the important variables precisely. On the contrary, TNR represents the proportion of the number of unimportant variables identified by our algorithm being true unimportant variables to the number of true unimportant variables. If TNR is closer to 1, it indicates that our algorithm can drop out the unimportant variables precisely. F-measure is a composite index of TPR and TNR, which represents the comprehensive performance of identifying the important variables and eliminating the unimportant variables.

The results presented in Table 1 are obtained from 100 repetitions. We observe the following results from Table 1. First, our proposed MM algorithm for lasso regression can precisely select the important variables, but it may identify some unimportant variables as important variables and this phenomenon is called over-selection. Second, the performance becomes better as sample size increases because information is more as sample size increases. Third, the performance is worse as dimension of covariates increases because the number of parameters increases. In general, our proposed algorithm works well in lasso regression.

Table 1: Performance of variable selection.

(n,p)	TPR	TNR	F-measure
(300, 50)	1.0000	0.9816	0.9906
(500, 50)	1.0000	0.9980	0.9990
(300, 100)	1.0000	0.9636	0.9813
(500, 100)	1.0000	0.9972	0.9986

4.2 Example 2: Least Absolute Deviation Regression

We consider another specific form of model (3.1). $X_i \in R^p$ is also generated from multivariate normal distribution with mean 0 and covariance matrix $\Sigma = (\sigma_{jk}) = (0.5^{|j-k|}), j, k = 1, \cdots, p$. $n = 50, 100, p = 5, \beta = (-2.3, 1.2, 1.7, -1.5, 2.4)^T$, where each component of β is draw from i.i.d. Uniform(-3, 3). We evaluate the algorithm described in Section 3.1.3 by the bias (Bias), empirical standard deviation (SD) and root mean squared error (RMSE) of the regression coefficients.

Table 2 summarizes the results obtained from 100 repetitions. From Table 2, we obtain the following conclusion. First, our algorithm can achieve the unbiased property of the estimates of regression coefficients. Second, we can obtain better estimates as sample size increases from 50 to 100.

Table 2: Performance of estimation.

n		β_1	β_2	β_3	β_4	β_5
50	Bias	-0.0016	-0.0003	-0.0186	-0.0051	0.0035
	SD	0.2147	0.2452	0.2359	0.2586	0.2112
	RMSE	0.2147	0.2452	0.2366	0.2587	0.2113
100	Bias	-0.0039	-0.0005	-0.0261	-0.0040	-0.0167
	SD	0.1389	0.1704	0.1885	0.2008	0.1350
	RMSE	0.1390	0.1704	0.1903	0.2008	0.1360

4.3 Example 3: Logistic Regression

We consider a specific form of model (3.2) which is called logistic model. $X_i \in R^p$ is also generated from multivariate normal distribution with mean 0 and covariance matrix $\Sigma = (\sigma_{jk}) = (0.5^{|j-k|}), j, k = 1, \cdots, p$. $n = 200, 300, p = 5, \beta_0 = (-2.3, 1.2, 1.7, -1.5, 2.4)^T$. And Y_i is from a Bernoulli distribution with probability $P_i = \frac{1}{1+\exp(-X_i'\beta_0)}$. Here, we consider the sample size $n = 200, 300$ because the signal of response variable is very weak.

We not only report the results from our proposed algorithms, simplified as Upbound

and Convexsupport, introduced in Section 3.2.1, but also the results from the iterative re-weighted least square (IRW-LS) algorithm, which is the standard algorithm in generalized linear model. All results , obtained from 100 repetitions, are summarized in Table 3. From Table 3, we conclude that Upbound performs best and our proposed two algorithms works a little better than IRW-LS but without significant difference. In general, our proposed algorithms are satisfactory.

Table 3: Performance of estimation.

n	Method		β_1	β_2	β_3	β_4	β_5
200	Upbound	Bias	0.1301	0.0942	0.0423	0.0974	0.1256
		SD	0.5689	0.5239	0.5146	0.5024	0.6426
		RMSE	0.5836	0.5323	0.5163	0.5118	0.6547
	Convexsupport	Bias	0.1338	0.0959	0.0453	0.0991	0.1297
		SD	0.5860	0.5274	0.5273	0.5053	0.6617
		RMSE	0.6011	0.5360	0.5293	0.5150	0.6743
	IRW-LS	Bias	0.1338	0.0959	0.0453	0.0991	0.1297
		SD	0.5890	0.5300	0.5300	0.5079	0.6651
		RMSE	0.6040	0.5386	0.5319	0.5175	0.6776
300	Upbound	Bias	0.1147	0.0326	0.0930	0.1258	0.1119
		SD	0.4414	0.3732	0.4227	0.3879	0.4692
		RMSE	0.4560	0.3746	0.4328	0.4078	0.4823
	Convexsupport	Bias	0.1147	0.0326	0.0931	0.1259	0.1120
		SD	0.4415	0.3732	0.4228	0.3880	0.4693
		RMSE	0.4562	0.3747	0.4329	0.4079	0.4824
	IRW-LS	Bias	0.1147	0.0326	0.0931	0.1259	0.1120
		SD	0.4437	0.3751	0.4249	0.3900	0.4716
		RMSE	0.4583	0.3765	0.4350	0.4098	0.4847

5 Discussion

In this paper, we mainly introduce the idea and construction method of a popular algorithm—MM algorithm. In addition, we use MM algorithm to solve some typical statistical problems, including lasso regression, group lasso regression, LAD regression and logistic regression. Finally, the designed simulation study illustrates that our constructed MM algorithm works very well to the corresponding problems.

In recent years, another popular algorithm is ADMM algorithm which divides one optimization problem into two sub-optimization problems in each iteration. If the two sub-optimization problems have closed-form solution, then the ADMM algorithm has a tremendous advantage. If not, ADMM algorithm is also not efficient. As further direction of research, we can try to combine ADMM algorithm and MM algorithm if two sub-optimization problems in ADMM iteration do not have closed-form solution. The convergence and efficiency of the algorithm is a good research topic to be investigated.

References

De Leeuw, J. and Heiser, W. J. (1977). Convergence of correction matrix algorithms for multidimensional scaling. *Geometric representations of relational data*, pages 735–752.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society Series B (methodological*, pages 1–38.

Hunter, D. R. (2004). Mm algorithms for generalized bradley-terry models. *The annals of statistics*, 32(1):384–406.

Hunter, D. R. and Lange, K. (2000). Quantile regression via an mm algorithm. *Journal of Computational and Graphical Statistics*, 9(1):60–77.

Hunter, D. R. and Lange, K. (2002). Computing estimates in the proportional odds model. *Annals of the Institute of Statistical Mathematics*, 54(1):155–168.

Hunter, D. R. and Lange, K. (2004). A tutorial on mm algorithms. *The American Statistician*, 58(1):30–37.

Hunter, D. R. and Li, R. (2002). A connection between variable selection and em-type algorithms. *Dept. of Statistics, Pennsylvania State University*, 201.

Hunter, D. R. and Li, R. (2005). Variable selection using mm algorithms. *Annals of statistics*, 33(4):1617.

Nguyen, H. D. (2016). An introduction to mm algorithms for machine learning and statistical. *arXiv1611.03969*.

Ortega, J. M. and Rheinboldt, W. C. (1970). *Iterative solution of nonlinear equations in several variables (Vol. 30)*. Siam.

Powers, D. M. W. (2011). Evaluation: From precision, recall and f-measure to roc, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 2(1):37–63.

Sabatti, C. and Lange, K. (2002). Genomewide motif identification using a dictionary model. *Proceedings of the IEEE*, 90(11):1803–1810.