Auxiliary-Function Methods in Iterative Optimization (4/15/15)

Charles L. Byrne[*]

April 15, 2015

Abstract

Let $C \subseteq X$ be a nonempty subset of an arbitrary set X and $f : X \to \mathbb{R}$. The problem is to minimize f over C. In auxiliary-function (AF) minimization we minimize $G_k(x) = f(x) + g_k(x)$ over x in C to get x^k, where $g_k(x) \geq 0$ for all x and $g_k(x^{k-1}) = 0$. Then the sequence $\{f(x^k)\}$ is nonincreasing. A wide variety of iterative optimization methods are either in the AF class or can be reformulated to be in that class, including forward-backward splitting, barrier-function and penalty-function methods, alternating minimization, majorization minimization (optimality transfer), cross-entropy minimization, and proximal minimization methods. In order to have the sequence $\{f(x^k)\}$ converge to β, the infimum of $f(x)$ over x in C, we need to impose additional restrictions. An AF algorithm is said to be in the SUMMA class if, for all x in C, we have the SUMMA Inequality: $G_k(x) - G_k(x^k) \geq g_{k+1}(x)$. Then $\{f(x^k)\} \downarrow \beta$. Here we generalize the SUMMA Inequality to obtain a wider class of algorithms that also contains the proximal minimization methods of Auslender and Teboulle. Algorithms are said to be in the SUMMA2 class if there are functions $h_k : X \to \mathbb{R}_+$ such that $h_k(x) - h_{k+1}(x) \geq f(x^k) - f(x)$ for all x in C. Once again, we have $\{f(x^k)\} \downarrow \beta$.

Key Words: Sequential unconstrained minimization; forward-backward splitting; proximal minimization; Bregman distances.

2000 Mathematics Subject Classification: Primary 47H09, 90C25; Secondary 26A51, 26B25.

1 Introduction

1.1 The Basic Problem

The basic problem we consider in this paper is to minimize a function $f : X \to \mathbb{R}$ over x in $C \subseteq X$, where C is an arbitrary nonempty subset of a set X. Until it is

[*] Charles_Byrne@uml.edu, Department of Mathematical Sciences, University of Massachusetts Lowell, Lowell, MA 01854

absolutely necessary, we shall not impose any structure on X or on f. One reason for avoiding structure on X and f is that we can actually achieve something interesting without it. The second reason is that when we do introduce structure, it will not necessarily be that of a metric space; for instance, cross-entropy and other Bregman distances play an important role in some of the iterative optimization algorithms to be discussed here.

1.2 Sequential Minimization

The algorithms we consider are of the sequential minimization type. For $k = 1, 2, ...$ we minimize the function

$$G_k(x) = f(x) + g_k(x) \tag{1.1}$$

over x in C to get x^k. An iterative algorithm is said to be an *interior-point* algorithm if $x^k \in C$ for all k. Many of the algorithms to be discussed here, including barrier-function methods, are interior-point algorithms. There are several ways to guarantee that x^k be in C. One way is to select $g_k(x)$ to be finite only within C. Another way is simply to minimize $G_k(x)$ only over C; if C is a proper subset of X we can replace $f(x)$ with $f(x) + \iota_C(x)$, where $\iota_C(x) = 0$, for $x \in C$, and $\iota_C(x) = +\infty$, otherwise. An iterative algorithm is said to be an *exterior-point* algorithm if x^k is outside C for all k, and only the limit, if it exists, is in C. Penalty-function methods are exterior-point methods.

While the functions $g_k(x)$ may be used to incorporate the constraint that $f(x)$ is to be minimized over $x \in C$, the $g_k(x)$ can also be selected to make the computations simpler; sometimes we select the $g_k(x)$ so that x^k can be expressed in closed form. However, in the most general, non-topological case, we are not concerned with calculational issues involved in finding x^k. Our objective is to select the $g_k(x)$ so that the sequence $\{f(x^k)\}$ converges to $\beta = \inf\{f(x), x \in C\}$.

1.3 Auxiliary-Function Methods

We shall say that the functions $g_k(x)$ are *auxiliary functions* if they have the properties $g_k(x) \geq 0$ for all $x \in X$, and $g_k(x^{k-1}) = 0$. We then say that the sequence $\{x^k\}$ has been generated by an *auxiliary-function* (AF) method. We have the following proposition.

Proposition 1.1 *If the sequence $\{x^k\}$ is generated by an AF method, then the sequence $\{f(x^k)\}$ is nonincreasing and converges to some $\beta^* \geq -\infty$.*

Proof: We have

$$G_k(x^{k-1}) = f(x^{k-1}) + g_k(x^{k-1}) = f(x^{k-1})$$
$$\geq G_k(x^k) = f(x^k) + g_k(x^k) \geq f(x^k),$$

so $f(x^{k-1}) \geq f(x^k)$. ∎

In order to have the sequence $\{f(x^k\}$ converging to $\beta = \inf\{f(x)|x \in C\}$ we need to impose additional restrictions.

Perhaps the best known examples of AF methods are the *sequential unconstrained minimization* (SUM) methods discussed by Fiacco and McCormick in their classic book [23]. They focus on barrier-function and penalty-function algorithms, which are not usually presented in AF form, but can be reformulated as members of the AF class. A wide variety of iterative optimization methods are either in the AF class or can be reformulated to be in that class, including forward-backward splitting, barrier-function and penalty-function methods, alternating minimization, majorization minimization (optimality transfer), cross-entropy minimization, and proximal minimization methods.

A barrier function has the value $+\infty$ for x not in C, while the penalty function is zero on C and positive off of C. In more general AF methods, we may or may not have $C = X$. If C is a proper subset of X, we can replace the function $f(x)$ with $f(x) + \iota_C(x)$, where $\iota_C(x)$ takes on the value zero for x in C and the value $+\infty$ for x not in C; then the $g_k(x)$ need not involve C.

1.4 The SUMMA Class

Simply asking that the sequence $\{f(x^k)\}$ be nonincreasing is usually not enough. We want $\{f(x^k)\} \downarrow \beta = \inf_{x\in C} f(x)$. This occurs in most of the examples mentioned above. In [9] it was shown that, if the auxiliary functions g_k are selected so as to satisfy the SUMMA Inequality,

$$G_k(x) - G_k(x^k) \geq g_{k+1}(x), \tag{1.2}$$

for all $x \in C$, then $\beta^* = \beta$. Although there are many iterative algorithms that satisfy the SUMMA Inequality, and are therefore in the SUMMA class, some important methods that are not in this class still have $\beta^* = \beta$; one example is the proximal minimization method of Auslender and Teboulle [1]. This suggests that the SUMMA class, large as it is, is still unnecessarily restrictive.

One consequence of the SUMMA Inequality is

$$g_k(x) - g_{k+1}(x) \geq f(x^k) - f(x), \tag{1.3}$$

for all $x \in C$. It follows from this that $\beta^* = \beta$. If this were not the case, then there would be $z \in C$ with

$$f(x^k) \geq \beta^* > f(z)$$

for all k. The sequence $\{g_k(z)\}$ would then be a nonincreasing sequence of nonnegative terms with the sequence of its successive differences bounded below by $\beta^* - f(z) > 0$. In order to widen the SUMMA class to include, among other algorithms, the proximal minimization method of Auslender and Teboulle, we shall focus on generalizing the inequality (1.3).

1.5 The SUMMA2 Class

An AF algorithm is said to be in the SUMMA2 class if, for each sequence $\{x^k\}$ generated by the algorithm, there are functions $h_k : C \to \mathbb{R}_+$ such that, for all $x \in C$, we have

$$h_k(x) - h_{k+1}(x) \geq f(x^k) - f(x). \tag{1.4}$$

Any algorithm in the SUMMA class is in the SUMMA2 class; use $h_k = g_k$. As in the SUMMA case, we must have $\beta^* = \beta$, since otherwise the successive differences of the sequence $\{h_k(z)\}$ would be bounded below by $\beta^* - f(z) > 0$. It is helpful to note that the functions h_k need not be the g_k, and we do not require that $h_k(x^{k-1}) = 0$.

As we shall show, the proximal minimization method of Auslender and Teboulle [1] is in the SUMMA2 class. It is natural to ask if there are algorithms in the SUMMA2 class that are not in SUMMA and are not in the class defined by Auslender and Teboulle. There are such algorithms; as we shall show, the *expectation maximization maximum likelihood* (EMML) [33, 4, 5, 6], as it is usually formulated, is such an algorithm.

2 Examples of AF Algorithms

In this section we consider several examples of AF methods.

2.1 Proximal Minimization

Let $d : X \times X \to \mathbb{R}_+$ be a "distance", meaning simply that $d(x, y) = 0$ if and only if $x = y$. An iterative algorithm is a *proximal minimization algorithm* (PMA) if, for each k, we minimize

$$G_k(x) = f(x) + d(x, x^{k-1}) \tag{2.1}$$

to get x^k. Clearly, any method in the PMA class is also an AF method.

2.2 Majorization Minimization

The *majorization minimization* (MM) method in statistics [26, 18], also called *optimization transfer*, is not typically formulated as an AF method, but it is one. The MM method is the following. Assume that there is a function $g(x|y) \geq f(x)$, for all x and y, with $g(y|y) = f(y)$. Then, for each k, minimize $g(x|x^{k-1})$ to get x^k. The MM methods and the PMA methods are equivalent; given $g(x|y)$, define $d(x,y) \doteq g(x|y) - f(x)$ and given $d(x,y)$, define $g(x|y) \doteq f(x) + d(x,y)$.

In [18] the authors give the following example of an MM method. Given finitely many nonempty, closed, convex subsets of $\mathbb{R}^J$, denoted $C_i, i = 1, ..., I$, the *convex feasibility problem* (CFP) is to find a member of their intersection. When the intersection is empty, we minimize the function

$$f(x) = \sum_{i=1}^{I} \|x - P_{C_i}x\|^2, \tag{2.2}$$

where $P_{C_i}x$ denotes the orthogonal projection of x onto the subset C_i. Using the MM method, they minimize instead the function

$$g(x|x^{k-1}) = \sum_{i=1}^{I} \|x - P_{C_i}x^{k-1}\|^2, \tag{2.3}$$

to get x^k. This is an MM method because

$$\|x - P_{C_i}x\| \leq \|x - P_{C_i}x^{k-1}\|,$$

for all x. The iterative algorithm generated in this way has the iterative step

$$x^k = \frac{1}{I}\sum_{i=1}^{I} P_{C_i}x^{k-1}, \tag{2.4}$$

which can also be written as a gradient-descent method,

$$x^k = x^{k-1} - \nabla f(x^{k-1}). \tag{2.5}$$

2.3 PMA with Bregman Distances

Let $\mathcal{H}$ be a Hilbert space, and $h : \mathcal{H} \to \mathbb{R}$ strictly convex and Gâteaux differentiable. The *Bregman distance* associated with h is

$$D_h(x,y) = h(x) - h(y) - \langle \nabla h(y), x - y \rangle. \tag{2.6}$$

Proximal minimization with Bregman distances (PMAB) applies to the minimization of a convex function $f : \mathcal{H} \to \mathbb{R}$. In [13, 14] Censor and Zenios discuss in detail the PMAB methods, which they call proximal minimization with D-functions.

Minimizing $G_k(x) = f(x) + D_h(x, x^{k-1})$ leads to

$$0 \in \partial f(x^k) + \nabla h(x^k) - \nabla h(x^{k-1}),$$

where

$$\partial f(x) = \{u | f(y) - f(x) - \langle \nabla u, y - x \rangle \geq 0, \text{for all}\, y\}$$

is the subdifferential of f at x. In [9] it was shown that for the PMAB methods we have $u^k \in \partial f(x^k)$ such that

$$G_k(x) - G_k(x^k) = f(x) - f(x^k) - \langle u^k, x - x^k \rangle + D_h(x, x^k) \geq g_{k+1}(x), \tag{2.7}$$

for all x. Consequently, the SUMMA Inequality holds and all PMAB algorithms are in the SUMMA class.

2.4 The Forward-Backward Splitting Methods

The *forward-backward splitting* (FBS) methods discussed by Combettes and Wajs [19] form a particular subclass of the PMAB methods. The problem now is to minimize the function $f(x) = f_1(x) + f_2(x)$, where both $f_1 : \mathcal{H} \to (-\infty, +\infty]$ and $f_2 : \mathcal{H} \to (-\infty, +\infty]$ are lower semicontinuous, proper and convex, and f_2 is Gâteaux differentiable, with L-Lipschitz continuous gradient. Before we describe the FBS algorithm we need to recall Moreau's proximity operators.

Following Combettes and Wajs [19], we say that the *Moreau envelope* of index $\gamma > 0$ of the closed, proper, convex function $f : \mathcal{H} \to (-\infty, \infty]$, or the Moreau envelope of the function γf, is the continuous, convex function

$$\text{env}_{\gamma f}(x) = \inf_{y \in \mathcal{H}} \{f(y) + \frac{1}{2\gamma} ||x - y||^2\}; \tag{2.8}$$

see also Moreau [27, 28, 29]. In Rockafellar's book [30] and elsewhere, it is shown that the infimum is attained at a unique y, usually denoted $\text{prox}_{\gamma f}(x)$. Proximity operators generalize the orthogonal projections onto closed, convex sets. Consider the function $f(x) = \iota_C(x)$, the *indicator function* of the closed, convex set C, taking the value zero for x in C, and $+\infty$ otherwise. Then $\text{prox}_{\gamma f}(x) = P_C(x)$, the orthogonal projection of x onto C. The following characterization of $x = \text{prox}_f(z)$ is quite useful: $x = \text{prox}_f(z)$ if and only if $z - x \in \partial f(x)$.

In [19] the authors show, using the characterization of $\mathrm{prox}_{\gamma f}$ given above, that x is a solution of this minimization problem if and only if

$$x = \mathrm{prox}_{\gamma f_1}(x - \gamma \nabla f_2(x)). \tag{2.9}$$

This suggests to them the following FBS iterative scheme:

$$x^k = \mathrm{prox}_{\gamma f_1}(x^{k-1} - \gamma \nabla f_2(x^{k-1})). \tag{2.10}$$

Basic properties and convergence of the FBS algorithm are then developed in [19].

In [11] we presented a simplified proof of convergence for the FBS algorithm. The basic idea used there is to formulate the FBS algorithm as a member of the PMAB class. An easy calculation shows that, if we minimize

$$G_k(x) = f_1(x) + f_2(x) + \frac{1}{2\gamma}\|x - x^{k-1}\|^2 - D_{f_2}(x, x^{k-1}), \tag{2.11}$$

we get x^k as described in Equation (2.10). The function

$$h(x) = \frac{1}{2\gamma}\|x\|^2 - f_2(x)$$

is convex and Gâteaux differentiable, when $0 < \gamma \leq \frac{1}{L}$, and

$$D_h(x, x^{k-1}) = \frac{1}{2\gamma}\|x - x^{k-1}\|^2 - D_{f_2}(x, x^{k-1}).$$

Therefore, the FBS method is in the PMAB class. A number of well known iterative algorithms are particular cases of the FBS.

2.5 Projected Gradient Descent

Let C be a nonempty, closed convex subset of $\mathbb{R}^J$ and $f_1(x) = \iota_C(x)$, the function that is $+\infty$ for x not in C and zero for x in C. Then $\iota_C(x)$ is convex, but not differentiable. We have $\mathrm{prox}_{\gamma f_1} = P_C$, the orthogonal projection onto C. The iteration in Equation (2.10) becomes

$$x^k = P_C\left(x^{k-1} - \gamma \nabla f_2(x^{k-1})\right). \tag{2.12}$$

The sequence $\{x^k\}$ converges to a minimizer of f_2 over $x \in C$, whenever such minimizers exist, for $0 < \gamma \leq 1/L$.

2.6 The CQ Algorithm and Split Feasibility

Let A be a real I by J matrix, $C \subseteq \mathbb{R}^J$, and $Q \subseteq \mathbb{R}^I$, both closed convex sets. The split feasibility problem (SFP) is to find x in C such that Ax is in Q. The function

$$f_2(x) = \frac{1}{2}\|P_Q Ax - Ax\|^2 \tag{2.13}$$

is convex, differentiable and ∇f_2 is L-Lipschitz for $L = \rho(A^T A)$, the spectral radius of $A^T A$. The gradient of f_2 is

$$\nabla f_2(x) = A^T(I - P_Q)Ax. \tag{2.14}$$

We want to minimize the function $f_2(x)$ over x in C or, equivalently, to minimize the function $f(x) = \iota_C(x) + f_2(x)$ over all x. The projected gradient descent algorithm in this case has the iterative step

$$x^k = P_C\left(x^{k-1} - \gamma A^T(I - P_Q)Ax^{k-1}\right); \tag{2.15}$$

this iterative method was called the CQ-algorithm in [7, 8]. The sequence $\{x^k\}$ converges to a solution whenever f_2 has a minimum on the set C, for $0 < \gamma \leq 1/L$.

If $Q = \{b\}$, then the CQ algorithm becomes the *projected Landweber* algorithm [3]. If, in addition, $C = \mathbb{R}^J$, then we get the Landweber algorithm [25]. In [15, 16] Yair Censor and his colleagues modified the CQ algorithm and applied it to derive protocols for intensity-modulated radiation therapy.

In the next few sections we consider several other optimization problems and iterative methods that are particular cases of the SUMMA class.

3 Barrier-Function and Penalty-Function Methods

Barrier-function methods and penalty-function methods for constrained optimization are not typically presented as AF methods [23]. However, barrier-function methods can be reformulated as AF algorithms and shown to be members of the SUMMA class. Penalty-function methods can be rewritten in the form of barrier-function methods, permitting several facts about penalty-function algorithms to be obtained from related results on barrier-function methods.

3.1 Barrier-Function Methods

The problem is to minimize $f : X \to \mathbb{R}$, subject to $x \in C$. We select $b : X \to (-\infty, +\infty]$ with $C = \{x|b(x) < +\infty\}$. For each k we minimize $B_k(x) = f(x) + \frac{1}{k}b(x)$

over all $x \in X$ to get x^k, which must necessarily lie in C. Formulated this way, the method is not yet in AF form. Nevertheless, we have the following proposition.

Proposition 3.1 *The sequence $\{b(x^k)\}$ is nondecreasing, and the sequence $\{f(x^k)\}$ is nonincreasing and converges to $\beta = \inf_{x \in C} f(x)$.*

Proof: From $B_k(x^{k-1}) \geq B_k(x^k)$ and $B_{k-1}(x^k) \geq B_{k-1}(x^{k-1})$, for $k = 2, 3, ...$, it follows easily that

$$\frac{1}{k-1}(b(x^k) - b(x^{k-1})) \geq f(x^{k-1}) - f(x^k) \geq \frac{1}{k}(b(x^k) - b(x^{k-1})).$$

Suppose that $\{f(x^k)\} \downarrow \beta^* > \beta$. Then there is $z \in C$ with

$$f(x^k) \geq \beta^* > f(z) \geq \beta,$$

for all k. Then

$$\frac{1}{k}(b(z) - b(x^k)) \geq f(x^k) - f(z) \geq \beta^* - f(z) > 0,$$

for all k. But the sequence $\{\frac{1}{k}(b(z) - b(x^k))\}$ converges to zero, which contradicts the assumption that $\beta^* > \beta$. ∎

The proof of Proposition 3.1 depended heavily on the details of the barrier-function method. Now we reformulate the barrier-function method as an AF method.

Minimizing $B_k(x) = f(x) + \frac{1}{k}b(x)$ to get x^k is equivalent to minimizing $kf(x) + b(x)$, which, in turn, is equivalent to minimizing

$$G_k(x) = f(x) + g_k(x),$$

where

$$g_k(x) = [(k-1)f(x) + b(x)] - [(k-1)f(x^{k-1}) + b(x^{k-1})].$$

Clearly, $g_k(x) \geq 0$ and $g_k(x^{k-1}) = 0$. Now we have the AF form of the method. A simple calculation shows that

$$G_k(x) - G_k(x^k) = g_{k+1}(x), \tag{3.1}$$

for all $x \in X$. Therefore, barrier-function methods are particular cases of the SUMMA class.

3.2 Penalty-Function Methods

Once again, we want to minimize $f : X \to \mathbb{R}$, subject to $x \in C$. We select a penalty function $p : X \to [0, +\infty)$ with $p(x) = 0$ if and only if $x \in C$. Then, for each k, we minimize

$$P_k(x) = f(x) + kp(x),$$

over all x, to get x^k. Here is a simple example of the use of penalty-function methods.

Let us minimize the function $f(x) = (x + 1)^2$, subject to $x \geq 0$. We let $p(x) = 0$ for $x \geq 0$, and $p(x) = x^2$, for $x < 0$. Then $x^k = -\frac{1}{k+1}$, which converges to zero, the correct answer, as $k \to +\infty$. Note that x^k is not in $C = \mathbb{R}_+$, which is why such methods are called *exterior-point methods*.

We suppose that $f(x) \geq \alpha > -\infty$, for all x. Replacing $f(x)$ with $f(x) - \alpha$ if necessary, we may assume that $f(x) \geq 0$, for all x. Clearly, it is equivalent to minimize

$$p(x) + \frac{1}{k} f(x),$$

which gives the penalty-function method the form of a barrier-function method. From Proposition 3.1 it follows that the sequence $\{p(x^k)\}$ is nonincreasing and converges to zero, while the sequence $\{f(x^k)\}$ is nondecreasing, and, as we can easily show, converges to some $\gamma \leq \beta$.

Without imposing further structure on X and f we cannot conclude that $\{f(x^k)\}$ converges to β. The reason is that, in the absence of further structure, such as the continuity of f, what f does within C can be unrelated to what it does outside C. If, for some f, we do have $\{f(x^k)\}$ converging to β, we can replace $f(x)$ with $f(x) - 1$ for x not in C, while leaving $f(x)$ unchanged for x in C. Then β remains unaltered, while the new sequence $\{f(x^k)\}$ converges to $\gamma = \beta - 1$.

4 Alternating Minimization

In later sections we discuss the *simultaneous multiplicative algebraic reconstruction technique* (SMART) and the *expectation maximization maximum likelihood* (EMML) algorithm. In [6] the SMART and the related EMML algorithm [33] were derived in tandem using the *alternating minimization* (AM) approach of Csiszár and Tusnády [20], which is the subject of this section.

Let $\Theta : A \times B \to (-\infty, +\infty]$, where A and B are arbitrary nonempty sets. In the AM approach we minimize $\Theta(a, b^{k-1})$ over $a \in A$ to get a^k and then minimize

$\Theta(a^k, b)$ over $b \in B$ to get b^k. We want

$$\{\Theta(a^k, b^k)\} \downarrow \beta = \inf\{\Theta(a, b) | a \in A, b \in B\}. \tag{4.1}$$

In [20] Csiszár and Tusnády show that, if the function Θ possesses what they call the *five-point property,*

$$\Theta(a, b) + \Theta(a, b^{k-1}) \geq \Theta(a, b^k) + \Theta(a^k, b^{k-1}), \tag{4.2}$$

for all a, b, and k, then (4.1) holds. There seemed to be no convincing explanation of why the five-point property should be used, except that it works. I was quite surprised when I discovered that the AM method can be reformulated as an AF method to minimize a function of the single variable a, and the five-point property for AM is precisely the SUMMA Inequality [10]. For each a select $b(a)$ for which $\Theta(a, b(a)) \leq \Theta(a, b)$ for all $b \in B$. Then let $f(a) = \Theta(a, b(a))$.

5 Applying Alternating Minimization

In [22] Eggermont and LaRiccia proved that alternating minimization using a Bregman distance $D_f(a, b)$ that is jointly convex, that is, convex with respect to the vector variable formed by concatenating a and b, has the five-point property.

In [2] Bauschke, Combettes and Noll consider the following problem: minimize the function

$$\Theta(a, b) = \Lambda(a, b) = \phi(a) + \psi(b) + D_f(a, b), \tag{5.1}$$

where ϕ and ψ are convex on $\mathbb{R}^J$, D_f is a Bregman distance, and $A = B$ is the interior of the domain of f. They assume that

$$\beta = \inf_{(a,b)} \Lambda(a, b) > -\infty, \tag{5.2}$$

and seek a sequence $\{(a^k, b^k)\}$ such that $\{\Lambda(a^k, b^k)\}$ converges to β. The sequence is obtained by the AM method, as in our previous discussion. They prove that, if the Bregman distance is jointly convex, then $\{\Lambda(a^k, b^k)\} \downarrow \beta$. In [12] this result was obtained by showing that $\Lambda(a, b)$ has the five-point property whenever D_f is jointly convex. The proof in [12] is related to that in [22]. From our previous discussion of AM, we conclude therefore that the sequence $\{\Lambda(a^k, b^k)\}$ converges to β; this is Corollary 4.3 of [2].

This suggests another class of proximal minimization methods for which $\beta^* = \beta$. Suppose that $D_f(x, y)$ is a jointly convex Bregman distance. For each $k = 1, 2, ...,$, we minimize

$$G_k(x) = f(x) + D_f(x^{k-1}, x) \tag{5.3}$$

to get x^k. Then using the result from [2], we may conclude that $\beta^* = \beta$.

6 Cross-Entropy Methods

For $a > 0$ and $b > 0$, let the cross-entropy or Kullback-Leibler (KL) distance [24] from a to b be

$$KL(a, b) = a \log \frac{a}{b} + b - a, \tag{6.1}$$

with $KL(a, 0) = +\infty$, and $KL(0, b) = b$. Extend to nonnegative vectors coordinate-wise, so that

$$KL(x, z) = \sum_{j=1}^{J} KL(x_j, z_j). \tag{6.2}$$

Then $KL(x, z) \geq 0$ and $KL(x, z) = 0$ if and only if $x = z$. The following lemma will be helpful later.

Lemma 6.1 *Let* $z_+ = \sum_{j=1}^{J} z_j > 0$. *Then*

$$KL(x, z) = KL(x_+, z_+) + KL(x, \frac{x_+}{z_+} z), \tag{6.3}$$

so that

$$KL(x, z) \geq KL(x_+, z_+). \tag{6.4}$$

Unlike the Euclidean distance, the KL distance is not symmetric; $KL(x, y)$ and $KL(y, x)$ are distinct. We can obtain different approximate solutions of a nonnegative system of linear equations $Px = y$ by minimizing $KL(Px, y)$ and $KL(y, Px)$ with respect to nonnegative x. The SMART minimizes $KL(Px, y)$, while the EMML algorithm minimizes $KL(y, Px)$. Both are iterative algorithms in the SUMMA class, and are best developed using the *alternating minimization* (AM) framework.

7 The SMART

The SMART and the EMML algorithm are similar in several respects, but differ in important ways that we shall consider in this and the next sections.

7.1 The SMART Iteration

The SMART [21, 32, 17, 4, 5, 6] minimizes the function $f(x) = KL(Px, y)$, over nonnegative vectors x. Here y is a vector with positive entries, and P is a matrix with nonnegative entries. For notational convenience, we shall assume that P and x have been rescaled so that $s_j = \sum_{i=1}^{I} P_{ij} = 1$. Denote by $\mathcal{X}$ the set of all nonnegative x for which the vector Px has only positive entries.

We begin with $x^0 > 0$. Having found the vector x^{k-1}, the next vector in the SMART sequence is $x^k = Sx^{k-1}$, where, for $x \in \mathcal{X}$, Sx is the vector with entries given by

$$(Sx)_j = x_j \exp\left(\sum_{i=1}^{I} P_{ij} \log \frac{y_i}{(Px)_i}\right). \tag{7.1}$$

Therefore, the SMART iterative step is

$$x_j^k = x_j^{k-1} \exp\left(\sum_{i=1}^{I} P_{ij} \log \frac{y_i}{(Px^{k-1})_i}\right). \tag{7.2}$$

In [4] the SMART was derived using the alternating minimization (AM) approach.

7.2 SMART as AM

For each $x \in \mathcal{X}$, let $r(x)$ and $q(x)$ be the I by J arrays with entries

$$r(x)_{ij} = x_j P_{ij} y_i / (Px)_i, \tag{7.3}$$

and

$$q(x)_{ij} = x_j P_{ij}. \tag{7.4}$$

In the iterative step of the SMART we get x^k by minimizing the function

$$KL(q(x), r(x^{k-1})) = \sum_{i=1}^{I} \sum_{j=1}^{J} KL(q(x)_{ij}, r(x^{k-1})_{ij})$$

over $x \geq 0$. Note that $KL(Px, y) = KL(q(x), r(x))$. The following Pythagorean identities help to reveal the properties of the SMART:

$$KL(q(x), r(z)) = KL(q(x), r(x)) + KL(x, z) - KL(Px, Pz); \tag{7.5}$$

and

$$KL(q(x), r(z)) = KL(q(Sz), r(z)) + KL(x, Sz). \tag{7.6}$$

Note that it follows from Equation (6.4) that $KL(x, z) - KL(Px, Pz) \geq 0$. Using the Pythagorean identities, we see that SMART fits into the AM framework; to get $x^k = Sx^{k-1}$ we minimize $\Theta(a, b^{k-1}) = KL(q(x), r(x^{k-1}))$ to get x^k. Then we minmize $\Theta(a^k, b) = KL(q(x^k), r(x))$ to get $x = x^k$ once again. Since KL is jointly convex, the five-point property holds and SMART is in the SUMMA class.

7.3 SMART as PMAB

In [4, 5, 6] it was shown that x^k can be obtained by minimizing

$$G_k(x) = KL(Px, y) + KL(x, x^{k-1}) - KL(Px, Px^{k-1}). \tag{7.7}$$

We have

$$KL(x, z) - KL(Px, Pz) = D_h(x, z), \tag{7.8}$$

for

$$h(x) = \sum_{j=1}^{J}(x_j \log x_j - x_j) - KL(Px, y),$$

which is convex and Gâteaux differentiable. Therefore, the SMART algorithm is a particular case of PMAB. The SMART sequence $\{x^k\}$ converges to the nonnegative minimizer of $f(x) = KL(Px, y)$ for which $KL(x, x^0)$ is minimized. If the entries of the starting vector x^0 are all one, then the sequence $\{x^k\}$ converges to the minimizer of $KL(Px, y)$ with maximum Shannon entropy [4].

7.4 SMART as SUMMA

We can show directly that the SMART is a particular case of the SUMMA, by showing that the SUMMA Inequality (1.2) holds. The inequality (6.4) is helpful in that regard. From Equation (7.5) we know that the iterative step of SMART can be expressed as follows: minimize the function

$$G_k(x) = KL(Px, y) + KL(x, x^{k-1}) - KL(Px, Px^{k-1}) \tag{7.9}$$

to get x^k. Using the inequality in (6.4), we see that the function

$$g_k(x) = KL(x, x^{k-1}) - KL(Px, Px^{k-1})$$

is nonnegative. The $g_k(x)$ are defined for all nonnegative x; that is, the set C is the closed nonnegative orthant in $\mathbb{R}^J$. Each x^k is a positive vector. From Equation (7.6) we have

$$G_k(x) = G_k(x^k) + KL(x, x^k), \tag{7.10}$$

from which it follows immediately that the SMART is in the SUMMA class.

7.5 Summarizing SMART

The following theorem summarizes the situation with regard to the SMART [4, 5, 6].

Theorem 7.1 *In the consistent case, in which $y = Px$ has a nonnegative solution, the SMART converges to the unique nonnegative solution of $y = Px$ for which the distance $\sum_{j=1}^J KL(x_j, x_j^0)$ is minimized. In the inconsistent case it converges to the unique nonnegative minimizer of the distance $KL(Px, y)$ for which $\sum_{j=1}^J KL(x_j, x_j^0)$ is minimized. If P and every matrix derived from P by deleting columns has full rank then there is a unique nonnegative minimizer of $KL(Px, y)$, say $\hat{x}$, and at most $I-1$ entries of $\hat{x}$ are nonzero.*

8 The EMML

The EMML algorithm [33, 4, 5, 6] is similar to the SMART in several respects, but different in important ways that we now discuss. The EMML minimizes the function $f(x) = KL(y, Px)$, over nonnegative vectors x.

For each nonnegative vector x in $\mathcal{X}$, we define the operator Tx by

$$(Tx)_j = x_j \sum_{i=1}^I \left(P_{i,j} \frac{y_i}{(Px)_i} \right). \tag{8.11}$$

Starting with $x^0 > 0$ and having found x^{k-1}, the next iterate is $x^k = Tx^{k-1}$. Therefore, the next vector in the EMML sequence is x^k with entries given by

$$x_j^k = x_j^{k-1} \left(\sum_{i=1}^I P_{ij} \frac{y_i}{(Px^{k-1})_i} \right). \tag{8.12}$$

The EMML algorithm is typically derived using alternating minimization.

8.1 EMML as AM

We use the same notation as in the previous section. Having found x^{k-1}, we find x^k by minimizing the function

$$KL(r(x^{k-1}), q(x)).$$

As in the case of the SMART, we have Pythagorean identities that help us discover the basic properties of the EMML algorithm:

$$KL(r(x), q(z)) = KL(r(z), q(z)) + KL(r(x), r(z)); \tag{8.13}$$

and

$$KL(r(x), q(z)) = KL(r(x), q(Tx)) + KL(Tx, z). \tag{8.14}$$

Note that $KL(y, Px) = KL(r(x), q(x))$. It follows from the inequality (6.4) that

$$KL(r(x), r(z)) \geq KL(Tx, Tz). \tag{8.15}$$

8.2 EMML as SUMMA

The EMML algorithm can be shown to be in the SUMMA class, but to do so requires a reformulation of EMML that is not entirely satisfactory.

We know from [22] that alternating minimization using the KL distance has the five-point property, and the EMML can be derived as AM using the KL distance. However, the order of the variables is not what we would like. In AM we minimize $\Theta(a, b^{k-1})$ over $a \in A$ to get a^k and then minimize $\Theta(a^k, b)$ over $b \in B$ to get b^k. In order to fit the EMML algorithm into this framework we must identify $r(x^{k-1})$ with a^k, and $q(x^k)$ with b^k. We showed that the problem in AM can be reformulated as minimizing $f(a) = \Theta(a, b(a))$. Now, this means that the function to be minimized by the EMML would be $KL(r(x), q(Tx))$, not $KL(r(x), q(x)) = KL(y, Px)$. This is unsatisfactory, in that the operator T associated with the particular algorithm being used, the EMML in this case, appears in the definition of the objective function to be minimized.

There is a second way, of course. We can let $r(x^{k-1})$ be b^{k-1} and $q(x)$ be a. However, now we can no longer rely on the result in [22], since the variables have been switched; the five-point property is not symmetric in the two vector variables.

We could try to show directly that the EMML is in SUMMA, by showing that the SUMMA Inequality (1.2) holds. Using the Pythagorean identities, (8.13) and (8.14), we see that we obtain x^k by minimizing

$$G_k(x) = KL(y, Px) + KL(r(x^{k-1}), q(x)) - KL(r(x^{k-1}), q(x^k)), \tag{8.16}$$

or

$$G_k(x) = KL(y, Px) + KL(r(x^{k-1}), r(x)) = f(x) + g_k(x). \tag{8.17}$$

From

$$G_k(x) - G_k(x^k) = KL(x^k, x), \tag{8.18}$$

the SUMMA Inequality (1.2) becomes

$$KL(x^k, x) \geq KL(r(x^k), r(x)) = g_{k+1}(x). \tag{8.19}$$

I have not been able to prove that this inequality holds, and I doubt that it is true.

We also know that the EMML fits into the PMA framework:

$$G_k(x) = f(x) + KL(r(x^{k-1}), r(x)) = f(x) + d(x, x^{k-1}). \tag{8.20}$$

However, the distance

$$d(x, z) = KL(r(z), r(x)) \tag{8.21}$$

is not a Bregman distance in x and z. Therefore, we do not have a PMAB algorithm. It is possible, of course, that this distance fits into the induced-proximal-distance approach of [1], which we shall discuss later, but that seems unlikely. We can, however, show that the EMML algorithm is in the SUMMA2 class.

8.3 The EMML as SUMMA2

When we try to exhibit the EMML algorithm as a member of the SUMMA class we encounter some difficulties. With $f(x) = KL(y, Px)$ and $g_k(x) = KL(r(x^{k-1}), r(x))$, it seems that the SUMMA Inequality (1.2) does not hold. We know, though, that Eggermont and LaRiccia tell us that AM with the KL distance has the five-point property, and therefore is in the SUMMA class. Since the EMML algorithm can be derived using AM with the KL distance, the EMML algorithm must be in the SUMMA class. However, exhibiting the EMML algorithm as a SUMMA algorithm in this way is artificial and unsatisfactory. In this section we show that the EMML algorithm is more naturally expressed as a member of the SUMMA2 class.

Using the Pythagorean identities (8.13) and (8.14), we write $KL(r(z), q(x^k))$ in two ways:

$$KL(r(z), q(x^k)) = KL(r(x^k), q(x^k)) + KL(r(z), r(x^k)), \tag{8.22}$$

and

$$KL(r(z), q(x^k)) = KL(y, Pz) - KL(Tz, z) + KL(Tz, x^k). \tag{8.23}$$

From the inequality (8.15) we have

$$KL(r(z), r(x^k)) \geq KL(Tz, Tx^k) = KL(Tz, x^{k+1}). \tag{8.24}$$

Therefore,

$$KL(Tz, x^k) - KL(Tz, x^{k+1}) \geq f(x^k) - f(z) = KL(y, Px^k) - KL(y, Pz). \tag{8.25}$$

With $h_k(z) = KL(Tz, x^k)$ we have

$$h_k(z) - h_{k+1}(z) \geq f(x^k) - f(z), \tag{8.26}$$

for all positive z. We conclude that the EMML algorithm is in the SUMMA2 class. Note that the functions $h_k(z)$ are not the $g_k(z)$.

8.4 Summarizing the EMML Algorithm

The following theorem summarizes the situation with regard to the EMML algorithm [4, 5, 6].

Theorem 8.1 *In the consistent case, in which $y = Px$ has a nonnegative solution, the EMML algorithm converges to a nonnegative solution $\hat{x}$ of $y = Px$. In the inconsistent case it converges to a nonnegative minimizer $\hat{x}$ of the distance $KL(y, Px)$. If P and every matrix derived from P by deleting columns has full rank then there is a unique nonnegative minimizer of $KL(y, Px)$, $\hat{x}$, and at most $I - 1$ entries of $\hat{x}$ are nonzero.*

In contrast to the SMART, we have no further information about the EMML limit in either the consistent or inconsistent cases. When the limit $\hat{x}$ is not unique, it will certainly depend on the choice of x^0, but how it depends on x^0 is unknown.

9 The PMA of Auslender and Teboulle

In [1] Auslender and Teboulle take C to be a closed, nonempty, convex subset of $\mathbb{R}^J$, with interior U. At the kth step of their method one minimizes a function

$$G_k(x) = f(x) + d(x, x^{k-1}) \tag{9.1}$$

to get x^k. Their distance $d(x, y)$ is defined for x and y in U, and the gradient with respect to the first variable, denoted $\nabla_1 d(x, y)$, is assumed to exist. The distance $d(x, y)$ is not assumed to be a Bregman distance. Instead, they assume that the distance d has an associated *induced proximal distance* $H(a, b) \geq 0$, finite for a and b in U, with $H(a, a) = 0$ and

$$\langle \nabla_1 d(b, a), c - b \rangle \ \leq H(c, a) - H(c, b), \tag{9.2}$$

for all c in U.

If $d = D_h$, that is, if d is a Bregman distance, then from the equation

$$\langle \nabla_1 d(b, a), c - b \rangle \ = D_h(c, a) - D_h(c, b) - D_h(b, a) \tag{9.3}$$

we see that D_h has $H = D_h$ for its associated induced proximal distance, so D_h is *self-proximal*, in the terminology of [1]. The method of Auslender and Teboulle seems not to be a particular case of SUMMA. However, it is in the SUMMA2 class, as we now show.

Since x^k minimizes $f(x) + d(x, x^{k-1})$, it follows that

$$0 \in \partial f(x^k) + \nabla_1 d(x^k, x^{k-1}),$$

so that

$$-\nabla_1 d(x^k, x^{k-1}) \in \partial f(x^k).$$

We then have

$$f(x^k) - f(x) \leq \langle \nabla_1 d(x^k, x^{k-1}), x - x^k \rangle.$$

Using the associated induced proximal distance H, we obtain

$$f(x^k) - f(x) \leq H(x, x^{k-1}) - H(x, x^k).$$

Therefore, this method is in the SUMMA2 class, with the choice of $h_k(x) = H(x, x^{k-1})$. Consequently, we have $\beta^* = \beta$ for these algorithms.

It is interesting to note that the Auslender-Teboulle approach places a restriction on the function $d(x, y)$, the existence of the induced proximal distance H, that is unrelated to the objective function $f(x)$, but this condition is helpful only for convex $f(x)$. In contrast, the SUMMA approach requires that

$$0 \leq g_{k+1}(x) \leq G_k(x) - G_k(x^k),$$

which involves the $f(x)$ being minimized, but does not require that $f(x)$ be convex; it does not even require any structure on X. The SUMMA2 approach is general enough to include both classes.

10 Summary

We have considered the problem of minimizing $f : X \to \mathbb{R}$ over x in C, a nonempty subset of the arbitrary set X. For $k = 1, 2, ...$ we minimize $G_k(x) = f(x) + g_k(x)$ to get x^k. For a sequence $\{x^k\}$ generated by an AF algorithm the sequence $\{f(x^k)\}$ is nonincreasing and converges to some $\beta^* \geq -\infty$. In addition, for AF algorithms in the SUMMA class we have $\{f(x^k)\} \downarrow \beta = \inf_{x \in C} f(x)$; so $\beta^* = \beta$.

The SUMMA class of algorithms is quite large, but there are algorithms not in the SUMMA class for which $\beta^* = \beta$; the proximal minimization method of Auslender and Teboulle [1] is an example. The SUMMA Inequality is sufficient to guarantee that $\beta^* = \beta$, but it is clearly overly restrictive. We extend the SUMMA class to the SUMMA2 class by generalizing the SUMMA Inequality and show that the methods of [1] are members of the larger SUMMA2 class.

Although the EMML algorithm is in the SUMMA class, to exhibit it as such requires us to reformulate the problem being solved in an artificial manner. The EMML can be shown, in a quite natural way, to be a member of the SUMMA2 class. The $h_k(z)$ are not the $g_k(z)$, nor do they have the form $h_k(z) = H(z, x^{k-1})$ for some induced proximal distance H, since the $h_k(z)$ involve the operator T.

References

1. Auslender, A., and Teboulle, M. (2006) “Interior gradient and proximal methods for convex and conic optimization.” *SIAM Journal on Optimization*, **16(3)**, pp. 697–725.

2. Bauschke, H., Combettes, P., and Noll, D. (2006) “Joint minimization with alternating Bregman proximity operators.” *Pacific Journal of Optimization*, **2**, pp. 401–424.

3. Bertero, M., and Boccacci, P. (1998) *Introduction to Inverse Problems in Imaging*, Bristol, UK: Institute of Physics Publishing.

4. Byrne, C. (1993) “Iterative image reconstruction algorithms based on cross-entropy minimization.” *IEEE Transactions on Image Processing* **IP-2**, pp. 96–103.

5. Byrne, C. (1995) “Erratum and addendum to ‘Iterative image reconstruction algorithms based on cross-entropy minimization’.” *IEEE Transactions on Image Processing* **IP-4**, pp. 225–226.

6. Byrne, C. (1996) “Iterative reconstruction algorithms based on cross-entropy minimization.” in *Image Models (and their Speech Model Cousins)*, S.E. Levinson and L. Shepp, editors, IMA Volumes in Mathematics and its Applications, Volume 80, pp. 1–11. New York: Springer-Verlag.

7. Byrne, C. (2002) “Iterative oblique projection onto convex sets and the split feasibility problem.” *Inverse Problems* **18**, pp. 441–453.

8. Byrne, C. (2004) “A unified treatment of some iterative algorithms in signal processing and image reconstruction.” *Inverse Problems* **20**, pp. 103–120.

9. Byrne, C. (2008) “Sequential unconstrained minimization algorithms for constrained optimization.” *Inverse Problems*, **24(1)**, article no. 015013.

10. Byrne, C. (2013) “Alternating minimization as sequential unconstrained minimization: a survey.” *Journal of Optimization Theory and Applications*, electronic **154(3)**, DOI 10.1007/s1090134-2, (2012), and hardcopy **156(3)**, February, 2013, pp. 554–566.

11. Byrne, C. (2014) “An elementary proof of convergence of the forward-backward splitting algorithm.” *Journal of Nonlinear and Convex Analysis* **15(4)**, pp. 681–691.

12. Byrne, C. (2014) *Iterative Optimization in Inverse Problems.* Boca Raton, FL: CRC Press.

13. Censor, Y., and Zenios, S.A. (1992) “Proximal minimization algorithm with D-functions.” *Journal of Optimization Theory and Applications*, **73(3)**, pp. 451–464.

14. Censor, Y. and Zenios, S.A. (1997) *Parallel Optimization: Theory, Algorithms and Applications*. New York: Oxford University Press.

15. Censor, Y., Bortfeld, T., Martin, B., and Trofimov, A. "A unified approach for inversion problems in intensity-modulated radiation therapy." *Physics in Medicine and Biology* 51 (2006), 2353-2365.

16. Censor, Y., Elfving, T., Kopf, N., and Bortfeld, T. (2005) "The multiple-sets split feasibility problem and its application for inverse problems." *Inverse Problems*, **21** , pp. 2071-2084.

17. Censor, Y. and Segman, J. (1987) "On block-iterative maximization." *J. of Information and Optimization Sciences* **8**, pp. 275–291.

18. Chi, E., Zhou, H., and Lange, K. (2014) "Distance Majorization and Its Applications." Mathematical Programming, **146 (1-2)**, pp. 409–436.

19. Combettes, P., and Wajs, V. (2005) "Signal recovery by proximal forward-backward splitting." *Multiscale Modeling and Simulation*, **4(4)**, pp. 1168–1200.

20. Csiszár, I. and Tusnády, G. (1984) "Information geometry and alternating minimization procedures."*Statistics and Decisions* **Supp. 1**, pp. 205–237.

21. Darroch, J. and Ratcliff, D. (1972) "Generalized iterative scaling for log-linear models."*Annals of Mathematical Statistics* **43**, pp. 1470–1480.

22. Eggermont, P., and LaRiccia, V. (2001) *Maximum Penalized Likelihood Estimation.* New York: Springer.

23. Fiacco, A., and McCormick, G. (1990) *Nonlinear Programming: Sequential Unconstrained Minimization Techniques.* Philadelphia, PA: SIAM Classics in Mathematics (reissue).

24. Kullback, S. and Leibler, R. (1951) "On information and sufficiency."*Annals of Mathematical Statistics* **22**, pp. 79–86.

25. Landweber, L. (1951) "An iterative formula for Fredholm integral equations of the first kind."*Amer. J. of Math.* **73**, pp. 615–624.

26. Lange, K., Hunter, D., and Yang, I. (2000) "Optimization transfer using surrogate objective functions (with discussion)." *J. Comput. Graph. Statist.*, **9**, pp. 1–20.

27. Moreau, J.-J. (1962) "Fonctions convexes duales et points proximaux dans un espace hilbertien." *C.R. Acad. Sci. Paris Sér. A Math.*, **255**, pp. 2897–2899.

28. Moreau, J.-J. (1963) “Propriétés des applications ‘prox’.” *C.R. Acad. Sci. Paris Sér. A Math.*, **256**, pp. 1069–1071.

29. Moreau, J.-J. (1965) “Proximité et dualité dans un espace hilbertien.” *Bull. Soc. Math. France*, **93**, pp. 273–299.

30. Rockafellar, R. (1970) *Convex Analysis.* Princeton, NJ: Princeton University Press.

31. Rockafellar, R.T. and Wets, R. J-B. (2009) *Variational Analysis* (3rd printing), Berlin: Springer-Verlag.

32. Schmidlin, P. (1972) “Iterative separation of sections in tomographic scintigrams.” *Nuklearmedizin* **11**, pp. 1–16.

33. Vardi, Y., Shepp, L.A. and Kaufman, L. (1985) “A statistical model for positron emission tomography.”*Journal of the American Statistical Association* **80**, pp. 8–20.