References

[1] Yu.M. Ermoliev, "Methods of stochastic Programming", Nauka, 1976, p.150 (in Russian).

[2] H. Robbins and S. Monro, "A stochastic approximation method", *Ann. Math. Statist.* **22**(1951), 400–407.

[3] H. Kesten, "Accelerated stochastic approximation", *Ann. Math. Statist.* **29**(1958), 41–59.

[4] G. Pflug, "On the determination of the step size in stochastic quasigradient methods". Collaborative Paper, CP-83-25, International Institute for Applied Systems Analysis, Laxenburg, Austria, 1983.

[5] S.P. Uriasiev, "Step regulation for direct methods of stochastic programming", *Kibernetika* **6**(1980), 85–87 (in Russian).

[6] S.P. Uriasiev, "Adaptive stepsize rules for stochastic optimization methods", Ph.D. thesis. Institute of Cybernetics, Kiev, 1983.

[7] F. Mirzoakhmedov and Urjasév, "Adaptive Step Size Control for Stochastic Optimization Algorithm", *Ih urn. vych.mat.i mat. fisiki,* **6**(1983), 1314–1325 (in Russian).

[8] R.E. Bruck, "On weak convergence of an Ergodic iteration the solution of variational inequalities for Monotone Operators in Hilbert Space".

CHAPTER 19

A NOTE ABOUT PROJECTIONS IN THE IMPLEMENTATION OF STOCHASTIC QUASIGRADIENT METHODS

*R.T. Rockafellar and R.J-B Wets**

Given a stochastic optimization problem find $x \in X \subset R^n$ that minimizes $F(x) = E\{f(x,\xi)\}$ where $f : R^n x \Xi \rightarrow R$ is a real-valued function, the quasi-gradient algorithm generates a sequence $\{x^1, x^2, \ldots\}$ of points of X (converging to the optimal solution with probability 1) through the recursion:

$$x^{\nu+1} := \underset{X}{\text{prj}}(x^\nu - \rho_\nu z^\nu)$$

where prj_X denotes the projection on X, $\{\rho_\nu, \nu = 1, \ldots\}$ is a sequence of positive scalars that tend to 0, and z^ν is a stochastic quasi-gradient of F at x^ν; see Chapter 5.

Unless X is a simple convex set, e.g. a rectangle or a ball, the projection operation may be too onerous to allow for a straightforward implementation of the iterative step; one would have to find at each step

$$x^{\nu+1} = \text{argmin}[\text{dist}^2(x^\nu - \rho_\nu z^n, x) | x \in X],$$

which means solving a mathematical program with quadratic objective function. Therefore the implementations of the stochastic quasi-gradient method rely usually on various schemes to bypass this projection operation, through penalization or primal-dual methods, for example. There are however a few cases when it is possible to design a very effective subroutine to perform the projection operation.

We describe a simple method for projecting a point $\hat{y} \in R^n_+$ on a convex set X, assumed to be nonempty, that is the intersection of a rectangle $C \subset R^n$ and a set determined by a single linear or more generally by a separable nonlinear constraint of the type:

$$\sum_{j=1}^{n} a_j(x_j) \leq b, \tag{19.1}$$

where the a_j are convex differentiable functions such that for every $j = 1, \ldots, n$, the derivative a'_j of a $a_j(\cdot)$ is positive and bounded away from zero on C where

$$C := \{x \in R^n | \ell_j \leq x_j \leq u_j, \quad j = 1, \ldots, n\} \tag{19.2}$$

* *Supported in part by grants of the National Science Foundation.*

with $\ell_j = -\infty$ and $u_j = +\infty$ if x_j is not bounded below or above. We had to deal with such a case in connection with the model described in Chapter 22. (For related work, cf. [**2**]-[**6**].) Since the derivative of a convex function is a monotone nondecreasing function, the preceding condition on the derivative is satisfied if (and only if)

$$a_j'(l_j) > 0 \text{ if } \ell_j \text{ is finite} \tag{19.3}$$

or if $\ell_j = -\infty$

$$\lim_{\tau \to -\infty} a_j'(\tau) = a_j'(\ell_j) > 0.$$

Set $a_j'(u_j) = \lim_{\tau \to +\infty} a_j'(\tau)$ if $u_j = +\infty$. In the special case when $a_j(\cdot)$ is linear, in which case we write

$$a_j(x_j) = a_j x_j, \tag{19.4}$$

this condition boils down to having $a_j > 0$.

The projection $\mathrm{prj}_X \hat{y}$ of $\hat{y}$ on X is the optimal solution of the (convex) nonlinear program

$$\begin{aligned} &\text{find} && x \in C \subset R^n \\ &\text{such that} && \sum_{j=1}^{n} a_j(x_j) \le b \\ &\text{and} && z = \frac{1}{2}\mathrm{dist}^2(\hat{y}, x) \text{ is minimized.} \end{aligned} \tag{19.5}$$

Here "dist" is the Euclidean distance, i.e. the objective is the quadratic form

$$\mathrm{dist}^2(\hat{y}, x) = \sum_{j=1}^{n} x_j^2 - 2\sum_{j=1}^{n} \hat{y}_j x_j + \sum_{j=1}^{n} \hat{y}_j^2. \tag{19.6}$$

Since the feasible region

$$X = C \cap \{x | \sum_{j=1}^{n} a_j(x_j) \le b\} \tag{19.7}$$

is a closed convex set, and the objective is an inf-compact (closed and bounded level sets) strictly convex function, the projection problem (19.5) is always solvable and it has a *unique* solution which is $\mathrm{prj}_X \hat{y}$.

Of course, it would be very easy to find the optimal solution of such a problem if there were no additional constraints besides $x \in C$. Our purpose is to show that with a single additional constraint it is possible to devise an algorithmic procedure for solving (19.5) that requires only marginally more work. This is achieved by constructing a (partial) dual to (19.5) whose solution gives us the (optimal) Lagrange multiplier λ^* to associate to the constraint $\Sigma_j a_j(x) \le b$.

When this multiplier λ^* is known, then the theory of convex optimization allows us to replace (19.5) by the following separable convex optimization problem:

$$\begin{aligned} &\text{find} && x \in C \subset R^n \\ &\text{such that} && \sum_{j=1}^{n} [\tfrac{1}{2}(x_j - \hat{y}_j)^2 + \lambda' a_j(x_j)] \text{ is minimized.} \end{aligned} \tag{19.8}$$

The solution to such a problem yields $x^* = \mathrm{prj}_X \hat{y}$, with

$$x_j^* = \begin{cases} \ell_j & \text{if } (\ell_j - \hat{y}_j) + \lambda^* a_j'(\ell_j) \ge 0, \\ u_j & \text{if } (u_j - \hat{y}_j) + \lambda^* a_j'(u_j) \le 0, \\ x_j & \text{where } x_j + \lambda^* a_j'(x_j) = \hat{y}_j, \text{ otherwise.} \end{cases} \tag{19.9}$$

In particular if $a_j(\cdot)$ is linear (19.4), then (19.9) becomes

$$x_j^* = \begin{cases} \ell_j & \text{if } (\ell_j - \hat{y}_j) + \lambda^* a_j \ge 0, \\ u_j & \text{if } (u_j - \hat{y}_j) + \lambda^* a_j \le 0, \\ \hat{y}_j - \lambda^* a_j & \text{otherwise.} \end{cases} \tag{19.10}$$

Thus all that is needed is an efficient procedure for finding λ^*. To do so let us consider the following convex optimization problem:

$$\begin{aligned} &\text{find } \lambda \in R_+ \\ &\text{such that} \quad g(\lambda) \text{ is maximized,} \end{aligned} \tag{19.11}$$

where

$$g(\lambda) = \min_{x \in C} \sum_{j=1}^{n} \frac{1}{2}(x_j - \hat{y}_j)^2 + \lambda a_j(x_j)] - \lambda b. \tag{19.12}$$

In fact this problem is *dual* to our original problem (19.8). This claim can be substantiated by appealing to the general duality theory for convex optimization problems, cf. [**7**]; the Lagrangian generating (19.5) and (19.11) as a dual pair of problems is the function:

$$L(x, \lambda) = \begin{cases} \sum_{j=1}^{n} [\frac{1}{2}(x_j - \hat{y}_j)^2 + \lambda a_j(x_j)] - \lambda b & \text{if } x \in C, y \ge 0, \\ +\infty & \text{if } x \notin C, \lambda \ge 0, \\ -\infty & \text{if } \lambda < 0. \end{cases}$$

We can also argue directly as follows: define

$$\varphi(\eta) = \sup[\eta\lambda + g(\lambda) | \lambda \in R_+].$$

Note that $\varphi(0)$ is then the optimal value of (19.11). From (19.12) it follows that

$$\varphi(\eta) = \sup_{\lambda \ge 0} \lambda \left[\sum_{j=1}^{n} a_j(x_j) - b+\right] + \min_{x \in C} \tfrac{1}{2}\mathrm{dist}^2(x, \hat{y})$$

and in particular for $\eta = 0$, since $X = C \cap \{x | \sum_{j=1}^{n} a_j(x_j) \leq b\}$ is nonempty, we obtain

$$\varphi(0) = \min_{x \in C} \frac{1}{2}[\mathrm{dist}^2(x, \hat{y})] \text{ if } \sum_{j=1}^{n} a_j(x_j) \leq b$$

which is the optimal value of the projection problem (19.5). The equality of the optimal values implies in turn that if x^0 solves (19.5) and λ^0 solves (19.11) then from definition (19.12), we have

$$\lambda^0 \left(\sum_{j=1}^{n} a_j(x_j^0) - b \right) = 0 \tag{19.13}$$

Thus the multiplier λ^* that we seek, to substitute in (19.9), is the optimal solution of (19.11), the 1-dimensional optimization problem (on R). For any $\lambda \in R_+$, we can find an explicit expression, that yields the argmin of (19.11), similar to (19.9), namely

$$x_j(\lambda) = \begin{cases} \ell_j & \text{if } \lambda \geq \eta_j^+ = (\hat{y}_j - \ell_j)/a_j'(\ell_j), \\ u_j & \text{if } \lambda \leq \eta_j^- = (\hat{y}_j - u_j)/a_j'(u_j'), \\ x_j & \text{if } \eta_j^- \leq \lambda \leq n_j^+ \\ & \text{where } x_j + \lambda a_j'(x_j) = \hat{y}_j. \end{cases} \tag{19.14}$$

Note that we have used the facts that a_j' is nonnegative and nondecreasing, so that $a_j'(\ell_j) \leq a_j'(u_j)$ and hence $\eta_j^- \leq \eta_j^+$ for all j. With

$$\begin{aligned} J^-(\lambda) &= \{j | \lambda < \eta_j^-\}, \\ J^+(\lambda) &= \{j | \lambda \geq \eta_j^+\}, \end{aligned} \tag{19.15}$$

and

$$J(\lambda) = \{j | \eta_j^- \leq \lambda < \eta_j^+\},$$

we have that

$$\begin{aligned} g(\lambda) = & \sum_{j \in J^-(\lambda)} [\frac{1}{2}(u_j - \hat{y}_j)^2 + \lambda a_j(u_j)] \\ & + \sum_{j \in J^+(\lambda)} left[\frac{1}{2}(\ell_j - \hat{y}_j)^2 + \lambda a_j(\ell_j)] \\ & + \sum_{j \in J(\lambda)} [\frac{1}{2}(x_j(\lambda) - \hat{y}_j)^2 + \lambda a_j(x_j(\lambda))] - \lambda b. \end{aligned} \tag{19.16}$$

The function g is concave: expression (19.12) gives us g as the sum of a linear function $(-b)\lambda$ and a min-function (of a collection of linear functions in λ). Thus the derivative, if it exists, is a monotone nonincreasing function of λ. Finding the maximum of g on R_+ corresponds to finding λ^* such that $g'(\lambda^*) = 0$, unless $g(0) \leq 0$ in which case $\lambda^* = 0$. Here, unless a_j' is pathological, we have that

$$\begin{aligned} g'(\lambda) = & \sum_{j \in J^-(\lambda)} a_j(u^j) + \sum_{j \in J^+(\lambda)} a_j(\ell_j) - b \\ & + \sum_{j \in J(\lambda)} [(x_j(\lambda) - \hat{y}_j)x_j'(\lambda) + a_j(x_j(\lambda)) + \lambda a_j'(x_j(\lambda))x_j'(\lambda)] \end{aligned}$$

and using the definition of $x_j(\lambda)$ when $j \in J(\lambda)$ this simplifies to

$$g'(\lambda) = \sum_{j \in J^-(\lambda)} a_j(u_j) + \sum_{j \in J^+(\lambda)} a_j(\ell_j) + \sum_{j \in J(\lambda)} a_j(x_j(\lambda)) - b. \tag{19.17}$$

In the linear case, this becomes

$$g'(\lambda) = \sum_{j \in J^-(\lambda)} a_j u_j + \sum_{j \in J^+(\lambda)} a_j \ell_j + \sum_{j \in J(\lambda)} [a_j \hat{y}_j - a_j^2 \lambda] - b. \tag{19.18}$$

To find $\lambda^* \in \mathrm{argmax}[g(\lambda) | \lambda \in R_+]$, we propose the following procedure:

Step 0. Order $\{\eta_j^-, \eta_j^+, j = 1, \ldots, \eta\}$, say as $(\theta_1, \ldots, \theta_{2n})$, recording for each θ_i the corresponding label $(j, -)$ or $(j, +)$. (Ties correspond to an entry in the θ-vector repeated the appropriate number of times.)
Set $\theta^- = 0, \theta^+ = \theta_p$ with $p = \min(j | \theta_j > 0.)$
Construct $J^-(\theta^- = 0), J^+(0), J(0)$.
Compute

$$g'(0) = \sum_{j \in J^-(0)} a_j(u_j) + \sum_{j \in J^+(0)} a_j(\ell_j) + \sum_{j \in J(0)} a_j(\hat{y}_j) - b.$$

If $g'(0) \leq 0$, stop. Set $\lambda^* = 0$ and exit.
If $g'(0) > 0$, continue.

Step 1. Compute $g'(\theta^+)$ using (19.17) or (19.18).
If $g'(\theta^+) \leq 0$, then find $\lambda^* \in [\theta^-, \theta^+]$ such that $g'(\lambda^*) = 0$, exit.
If $g'(\theta^+) > 0$, continue.

Step 2. Set $p := p + 1, \theta^- := \theta^+, \theta^+ := \theta_p$
Adjust $J^-(\theta^-), J^+(\theta^-), J(\theta^-)$
Return to Step 1.

The algorithm clearly converges since it is a systematic search of a monotone nonincreasing function that eventually must reach the interval $[\alpha_p, \alpha_{p+1}]$ in which g' takes on the value 0; the problem is known to have a solution, see the preceding comments about duality.

In the linear case, all operations prescribed by the algorithm are simple and straightforward. The derivative $g'(\lambda)$ is given by (19.18). In Step 1, when $g'(\alpha^+) \leq 0, \lambda^*$ is given by the expression

$$\lambda^* = \beta/\gamma$$

where

$$\beta = \sum_{j\in J^-} a_j u_j + \sum_{j\in J^+} a_j \ell_j + \sum_{j\in J} a_j \widehat{y}_j - b,$$

and

$$\gamma = \sum_{j\in J} a_j^2.$$

When the $a_j(\cdot)$ are nonlinear, the evaluation of $g'(\lambda)$ requires first the evaluation of $x_j(\lambda)$ for all $j \in J(\lambda)$. Also in Step 1 there may be difficulties in finding λ^* when $g'(\theta^+) \leq 0$. To begin with, let us consider the equations

$$x_j + \lambda a_j'(x_j) = \widehat{y}_j. \tag{19.19}$$

Usually, there are many situations when it is easy to find a closed form expression for x_j as a function of λ. For example, if $a_j(z) = \alpha z^2 + z + \gamma$ with $\alpha > 0$ (recall that $a_j(\cdot)$ is convex), then

$$x_j(\lambda) = (\widehat{y}_j - \lambda\beta)/(1 + 2\alpha\lambda).$$

In general, however, even when an explicit expression for the derivative is available, we may have to resort to a numerical procedure for finding $x_j(\lambda)$. But here we are greatly aided by the following observation. For $\lambda \in [{}_j^-, \eta_j^+]$ the function

$$z \mapsto (z + \lambda a_j'(z) - \widehat{y}_j)$$

is monotone nondecreasing between ℓ_j and u_j with

$$(\ell_j - \widehat{y}_j) + \lambda a_j'(\ell_j) < 0$$

and

$$(u_j - \widehat{y}_j) + \lambda a_j'(u_j) \geq 0,$$

as follows from the definition of η_j^- and η_j^+, see (19.14). Thus a secant method [1], that we used in our implementation, is a very efficient procedure to find $x_j(\lambda)$.

We now turn to finding λ^* with $g'(\lambda^*) = 0$, knowing that

$$g'(\theta^-) > 0 \text{ and } g'(\theta^+) \leq 0,$$

where g' is given by (19.17). The sets $J^-(\lambda)$, $J^+(\lambda)$ and $J(\lambda)$ remain fixed on this interval. Let

$$\beta = b - \sum_{j\in J^-} a_j(u_j) - \sum_{j\in J^+} a_j(\ell_j),$$

and

$$\gamma(\lambda) = \sum_{j\in J(\lambda)} a_j(x_j(\lambda)).$$

Note that from the definition of θ^- and θ^+ it follows that

$$\eta_j^- \leq \theta^- \leq \theta^+ \leq \eta_j^+, \text{ for all } j \in J.$$

Moreover, $\lambda \mapsto \gamma(\lambda)$ is a decreasing function with

$$\gamma(\theta^-) > \beta \text{ and } \gamma(\theta^+) \leq \beta.$$

We need to find λ^* such that $\gamma(\lambda^*) = \beta$. Unless we have some expression for $a_j(x_j(\lambda))$ that can be handled easily, we again need to rely on a numerical procedure, and in this case too the secant method suggests itself [**1**]. That is what we have used in our own implementation of the procedure.

This projection method is extremely efficient in the linear case but also produces very good results in the nonlinear case, in which case its efficiency is that of the secant method used in finding λ^* and $x_j(\lambda)$.

If there is more than one constraint, in addition to the upper and lower bounds, it may still be possible to use the procedure outlined here. For example it is possible to keep track of the active constraints, and when only one (or no) extra constraint is violated then we could use this procedure to obtain the projection, provided the projected point does not violate some other constraint. We should thus be able to cope with two or three extra constraints, resorting only once in a while to a general optimization procedure for solving (19.5).

Acknowledgment: We very much appreciate the comments of Dr. Richard Cottle (Stanford University) as well as his help on bibliographical questions.

References

[1] R.P. Brent, *Algorithms for Minimization with Derivatives*, Series in Automatic Computation, Prentice-Hall, New Jersey, 1978.

[2] G.R. Bitran and A.C. Hax, "Disaggregation and resource allocation using convex knapsack problems with bounded variables", *Management Science* **27**(1981) 431–441.

[3] R.W. Cottle and S.G. Duvall, *A Lagrangean relaxation algorithm for the constrained matrix problem*, Technical Report SOL 82-10, Systems Optimization Laboratory, Department of Operations Research, Stanford University, 1982.

[4] J.A. Ferland, B. Lemaire, and P. Robert, *Analytic solutions for nonlinear programs with one or two equality constraints*, Technical Report # 285, Departement d'Informatique et de Recherche Operationnelle, Université de Montréal, 1978.

[5] R. Helgason, J. Kennington and H. Lall, "A polynomial algorithm for a singly constrained quadratic program", *Mathematical Programming* **18** (1980) 338–343.

[6] R.K. McCord, "Minimization with one equality constraint and bounds on the variables", Ph.D. Thesis, Department of Operations Research, Stanford University, 1979.

[7] R.T. Rockafellar, *Conjugate Duality and Optimization*, Conference Series in Applied Mathematics 16, SIAM Publications, Philadelphia, 1974.