Maximum of Quadratics

N. Karmitsa[1]

Abstract: In this note a definition for maximum of quadratic -functions is given as well as an algorithm for generating data for these kind of functions.

Maximum of quadratic -functions are defined as the point-wise maximum of a finite collection of quadratic functions. That is

$$f(\boldsymbol{x}) = \max\{f_j(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T A_j \boldsymbol{x} + \boldsymbol{b}_j^T \boldsymbol{x} + c_j \mid j = 1, \ldots, n_f\}, \tag{1}$$

where A_j are $n \times n$ symmetric matrices (in the convex case positive definite), $\boldsymbol{b}_j \in \mathbb{R}^n$ and $c_j \in \mathbb{R}$. With this definition, many different examples are easily created by choosing the values of n, n_f, and the sparsity parameter $0 \le p_s \le 1$ ($p_s = 0$ causes the diagonal matrix, $p_s = 1$ causes the dense matrix, and $0 < p_s < 1$ causes the sparse matrix with approximately $p_s n^2 + n$ nonzeros) and then randomly generating n_f objects A_j, $\boldsymbol{b}_j$ and c_j. Depending on the positive definiteness of matrices A_j both convex and nonconvex problems can be created.

In the following algorithm some details of the data generation are given. In the convex case, the positive definiteness of matrices A_j ($j = 1, \ldots, n_f$) is guaranteed by adding the identity matrix multiplied by one plus the absolute value of the smallest eigenvalue to each matrix A_j. In the nonconvex case, we check that the minimum eigenvalue for each matrix A_j ($j = 2, \ldots, n_f$) is negative. However, we enforce the first elemental function to be convex (i.e. A_1 to be positive definite) in order to obtain finite results. This does not restrict the overall nonconvexity of the problem (assuming $n_f > 1$) since at $\boldsymbol{x} = 0$ all the nonconvex elemental functions are bigger than the convex one (see the algorithm for more details).

A subdifferential of a maximum of quadratics is given by

$$\partial f(\boldsymbol{x}) = \text{conv}\{\partial f_i(\boldsymbol{x}) \mid f_i(\boldsymbol{x}) = f(\boldsymbol{x})\}. \tag{2}$$

Thus, at any given $\boldsymbol{x}$ any active index $i \le n_f$ (i.e. an index where the maximum of (1) is attained), we have a subgradient (generalized gradient) $A_i\boldsymbol{x} + \boldsymbol{b}_i \in \partial f(\boldsymbol{x})$ easily available.

The MatLab-file `makeproblem.m` for generating the random data as well as the Fortran subroutine `maxq.f` that reads the data-file and calculates the value of the function and subgradient are available for downloading from http://napsu.karmitsa.fi/testproblems/. In addition some ready-to-use data-files are given for $L = -10$, $U = 50$, $n = 50$, $n_f = 5$ and 10, and $p_s = 0$, 0.6 and 1.

1. Department of Mathematics, University of Turku, FI-20014 Turku, Finland. E-mail: napsu@karmitsa.fi

PROGRAM Create Data for Maximum of Quadratics

INITIALIZE Select a lower and upper bound for random number generator $L, U \in \mathbb{R}$. Fix the dimension n, the number of elemental functions n_f and the sparsity parameter p_s $(0 \leq p_s \leq 1)$. Set $i_{con} = 1$, if a convex problem is needed and $i_{con} = 0$, otherwise;

IF $i_{con} = 0$ THEN

  Set $c_1 = L$ and randomly generate $c_j \in (L, U)$ for $j = 2, \dots, n_f$;

ELSE

  Randomly generate $c_j \in (L, U)$ for $j = 1, \dots, n_f$;

END IF

FOR ALL $j = 1, \dots, n_f$ randomly generate vectors $\boldsymbol{b}_j \in (L, U)^n$;

FOR ALL $j = 1, \dots, n_f$ randomly generate symmetric matrices $A_j \in (L, U)^{n \times n}$ such that there exist approximately $p_s n^2 + n$ nonzero entries and all the diagonal elements are nonzero.

  IF $i_{con} = 1$ THEN

    Add the identity matrix multiplied by one plus the absolute value of the smallest eigenvalue to each matrix A_j to obtain positive definite matrices;

  ELSE

    Add the identity matrix multiplied by one plus the absolute value of the smallest eigenvalue to the first matrix A_1. Otherwise, check that the minimum eigenvalue is negative for each matrix A_j $(j = 2, \dots, n_f)$. Regenerate any A_j whose eigenvalue is non-negative;

  END IF

Randomly generate the starting point $\boldsymbol{x}_1 \in (L, U)^n$;

END Create Data for Maximum of Quadratics