Numerical Optimization - Final Project

Active-Set Methods for Convex Quadratic Optimization
Problems with Inequality Constraints

Branch Vincent

May 3, 2017

Abstract

An optimization problem with a convex, quadratic objective function and linear constraints is called a convex quadratic program (QP). The main challenge in convex QPs with inequality constraints is determining the optimal active set. In this paper, we present an active-set method that generates feasible iterates, while decreasing the objective function. We apply this method to an example problem, varying the choice of the initial point. Results show that the algorithm usually converges quickly, but may struggle at boundary points.

Contents

1 Theory

In this section, we formally express the convex quadratic optimization problem. Then, we summarize the optimality conditions for equality and inequality-constrained problems. Lastly, we present one class of algorithms, the active-set method, for solving convex quadratic programs that contain both equality and inequality constraints.

1.1 Convex Quadratic Program

In general, a convex quadratic program (QP) can be expressed as

$$\min_{x} \quad q(x) = \frac{1}{2}x^T G x + x^T c \tag{1a}$$
$$\text{subject to} \quad a_i^T x = b_i \qquad i \in \mathcal{E} \tag{1b}$$
$$a_i^T x \geq b_i \qquad i \in \mathcal{I} \tag{1c}$$

where G is symmetric, positive semidefinite and $\mathcal{E}$ and $\mathcal{I}$ are finite sets of indices, referring to the equality and inequality constraints, respectively. The difficulty of this problem depends on the objective function, q, and the number of inequality constraints.

1.2 Optimality Conditions

The KKT conditions can easily be applied to a convex quadratic problem with both equality and inequality constraints. By doing so, we arrive at the following first-order, sufficient conditions for x^* to be a global solution:

$$Gx^* + c - \sum_{i \in \mathcal{A}(x^*)} \lambda_i^* a_i = 0 \tag{2a}$$
$$a_i^T x^* = b_i \qquad \forall i \in \mathcal{A}(x^*) \tag{2b}$$
$$a_i^T x^* \geq b_i \qquad \forall i \in \mathcal{I} \setminus \mathcal{A}(x^*) \tag{2c}$$
$$\lambda_i^* \geq 0 \qquad \forall i \in \mathcal{I} \cap \mathcal{A}(x^*) \tag{2d}$$

where λ_i^* are Lagrange multipliers and $\mathcal{A}(x^*)$ is the active set, which contains the indices of the constraints for which equality holds at x^*. In other words,

$$\mathcal{A}(x^*) = \{i \in \mathcal{E} \cup \mathcal{I} \mid a_i^T x^* = b_i\} \tag{3}$$

These sufficient conditions are formally proved in Theorem 16.4 from Nocedal & Wright [1]. Furthermore, when G is positive definite (as opposed to positive semidefinite), x^* is the unique global solution. Note that we can also derive second-order sufficient conditions, which is not done here for sake of brevity.

Complications may arise when G is not positive definite, as there could be more than one strict local solution. Another issue is degeneracy, which occurs when:

(a) gradients a_i for $i \in \mathcal{A}(x^*)$ are linearly dependent at x^*

(b) $\exists\, i \in \mathcal{A}(x^*)$ such that all Lagrange multipliers satisfying Eqn. (2) have $\lambda_i^* = 0$

Degeneracy in the form of (a) can create difficulties in the step computation, because certain matrices become rank deficient, while (b) can cause problems in determining the active constraints at the solution, possibly leading iterates to zigzag.

1.3 Active-Set Methods

The main challenge of this problem is determining the optimal active set, $\mathcal{A}(x^*)$. If this set was known, then we could reduce the problem to the following equality-constrained QP:

$$\min_{x} \quad q(x) = \frac{1}{2}x^T G x + x^T c \tag{4a}$$

$$\text{subject to} \quad a_i^T x = b_i \qquad i \in \mathcal{A}(x^*) \tag{4b}$$

However, this is rarely the case. Instead, active-set methods can be used to solve such problems. They are effective with small and medium sized problems and can accurately estimate the optimal active set.

Primal active-set methods are a specific type, and the focus of this paper, that generates feasible iterates with respect to Eqn. (1), while decreasing the objective function q. More specifically, the method utilizes a working set, W_k, defined as a set containing some of the inequality constraints and all the equality constraints. Now given an iterate x_k and $\mathcal{W}_k$, we check whether x_k minimizes the quadratic q in the subspace defined by $\mathcal{W}_k$. If not, we compute a step p by solving an equality-constrained QP subproblem in which the constraints corresponding to $\mathcal{W}_k$ are regarded as equalities while all other constraints are temporarily disregarded. By substituting $x = x_k + p$ into the objective function q and manipulating the result, we can express this subproblem as

$$\min_{p} \quad \frac{1}{2}p^T G p + p^T g_k \tag{5a}$$

$$\text{subject to} \quad a_i^T p = 0 \qquad i \in \mathcal{W}_k \tag{5b}$$

where

$$p = x - x_k \tag{6}$$

$$g_k = G x_k + c \tag{7}$$

This subproblem can be solved using any equality-constrained QP technique (see Section 16.2 in Nocedal & Wright [1]). Now if the solution p_k is nonzero, we set

$$x_{k+1} = x_k + \alpha_k p_k \tag{8}$$

where α_k is the largest value in $[0,1]$ for which all constraints are satisfied. From this, we can derive the following expression for α_k.

$$\alpha_k \equiv \min\left(1, \min_{i \notin \mathcal{W}_k, a_i^T p_k < 0} \frac{b_i - a_i^T x_k}{a_i^T p_k}\right) \tag{9}$$

Note that if $\alpha_k < 1$, the step was blocked by some constraint not in $\mathcal{W}_k$. Thus, $\mathcal{W}_{k+1}$ is constructed by adding the blocking constraint to $\mathcal{W}_k$. The above process is continued until we reach an $\hat{x}$ that minimizes q over the current working set $\hat{\mathcal{W}}$. Thus, using the optimality conditions for equality-constrained QPs (see Equation 16.5 in Nocedal & Wright [1]), we have

$$\sum_{i \in \hat{\mathcal{W}}} a_i \hat{\lambda}_i = g \tag{10}$$

If we define the multipliers corresponding to the inequality constraints not in $\hat{\mathcal{W}}$ to be zero, then $\hat{x}$ and $\hat{\lambda}$ satisfy the first three KKT conditions from Eqn. (2). Furthermore, if the constraints $i \in \hat{\mathcal{W}} \cap \mathcal{I}$ are all nonnegative, then the fourth KKT is also satisfied and thus $\hat{x}$ is the global solution. Otherwise, q may be decreased by dropping one the negative constraints.

Nocedal & Wright [1] prove, in Theorems 16.5 & 16.6, that this strategy produces a feasible direction p_k and that q is strictly decreasing along this p_k.

2 Implementation

2.1 Algorithm

Elaborating on Section (1.3), we can formally specify the active-set method into the following algorithm, which was used to obtain this paper's results. Note that this is Algorithm 16.3 from Nocedal & Wright [1].

Algorithm 1 Active-Set Method for Convex QP

$x_0 \leftarrow$ feasible starting point
$\mathcal{W}_0 \leftarrow$ subset of $\mathcal{A}(x_0)$
for $k = 0, 1, ...$
    $p_k \leftarrow$ solve Eqn. (5)
    if $p_k = 0$
        $\hat{\lambda}_i \leftarrow$ solve Eqn. (10)
        if $\hat{\lambda}_i \geq \forall i \in \mathcal{W}_k \cap \mathcal{I}$
            stop with $x^* \leftarrow x_k$
        else
            $j \leftarrow \text{argmin}_{j \in \mathcal{W}_k \cap \mathcal{I}} \hat{\lambda}_j$
            $x_{k+1} \leftarrow x_k$
            $\mathcal{W}_{k+1} \leftarrow \mathcal{W}_k \setminus \{j\}$
    else
        $\alpha_k \leftarrow$ solve Eqn. (9)
        $x_{k+1} \leftarrow x_k + \alpha_k p_k$
        if $\exists$ blocking constraint, c_j
            $\mathcal{W}_{k+1} \leftarrow \mathcal{W}_k \cup j$
        else
            $\mathcal{W}_{k+1} \leftarrow \mathcal{W}_k$

This algorithm identifies x^* in a finite number of iterations, assuming that the method takes a nonzero α_k whenever p_k is nonzero. In other words, we assume cycling is avoided.

2.2 Problem Statement

In this paper, we examine the following problem:

$$\begin{aligned} \min_x \quad & q(x) = x_1^2 + 2x_2^2 - 2x_1x_2 - 2x_1 - 6x_2 && \text{(11a)} \\ \text{subject to} \quad & -2.41x_1 - x_2 \geq -6.83 && \text{(11b)} \\ & -0.41x_1 - x_2 \geq -4 && \text{(11c)} \\ & 0.41x_1 - x_2 \geq -4 && \text{(11d)} \\ & 2.41x_1 - x_2 \geq -6.83 && \text{(11e)} \\ & 2.41x_1 + x_2 \geq -2.83 && \text{(11f)} \\ & 0.41x_1 + x_2 \geq 0 && \text{(11g)} \\ & -0.41x_1 + x_2 \geq 0 && \text{(11h)} \\ & -2.41x_1 + x_2 \geq -2.83 && \text{(11i)} \end{aligned}$$

Note this is the same objective function as Problem 16.11 in Nocedal & Wright [1], but with different constraints. We can formulate the problem into the standard form from Eqn. (1), with

$$
\begin{aligned}
G &= \begin{bmatrix} 2 & -2 \\ -2 & 2 \end{bmatrix} & c &= \begin{bmatrix} -2 \\ -6 \end{bmatrix} \\
a^T &= \begin{bmatrix} -2.41 & -1 \\ -0.41 & -1 \\ 0.41 & -1 \\ 2.41 & -1 \\ 2.41 & 1 \\ 0.41 & 1 \\ -0.41 & 1 \\ -2.41 & 1 \end{bmatrix} & b &= \begin{bmatrix} -6.83 \\ -4 \\ -4 \\ -6.83 \\ -2.83 \\ 0 \\ 0 \\ -2.83 \end{bmatrix} \\
\mathcal{E} &= \emptyset & \mathcal{I} &= \{1,\ 2,\ 3,\ 4,\ 5,\ 6,\ 7,\ 8\}
\end{aligned}
\tag{12}
$$

These constraints were specifically formulated to give an octagonal feasible region. This was chosen in order to challenge the algorithm with a relatively large number of inequality constraints. Alternatively, a higher dimensional problem would have also presented a challenge, but this was not chosen in order to visualize the results.

3 Results and Discussion

3.1 Results

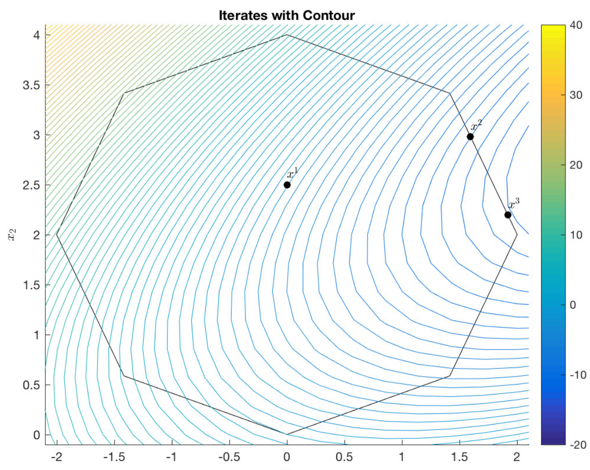
Using the problem defined in Section (2.2), we explore the active-set algorithm's behavior for several initial points.

3.1.1 Interior Point

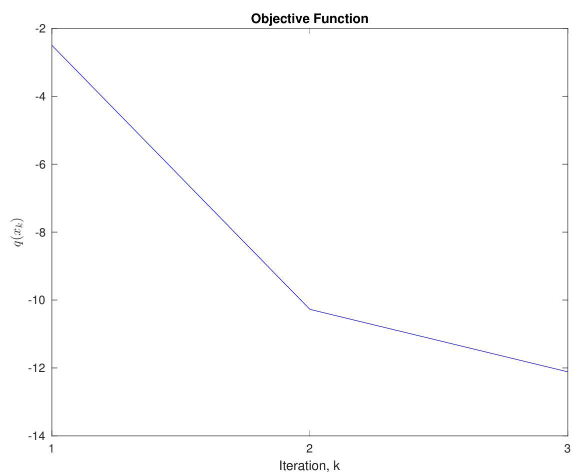
First, consider an interior point of the feasible region as the initial point. Table 1 reports important computed values for each iteration of the algorithm while Figure 1 helps visualize the objective function's descent.

Table 1: Interior point iterations

Iteration, k	$\mathbf{x_k}$	$\mathbf{W_k}$	$\mathbf{p_k}$	$\alpha_\mathbf{k}$
1	(0, 2.5)	{}	(5, 1.5)	0.32
2	(1.595, 2.978)	{1}	(0.324, -0.783)	1.00
3	(1.919, 2.195)	{1}	(0, 0)	



(a) Contour



(b) Objective function

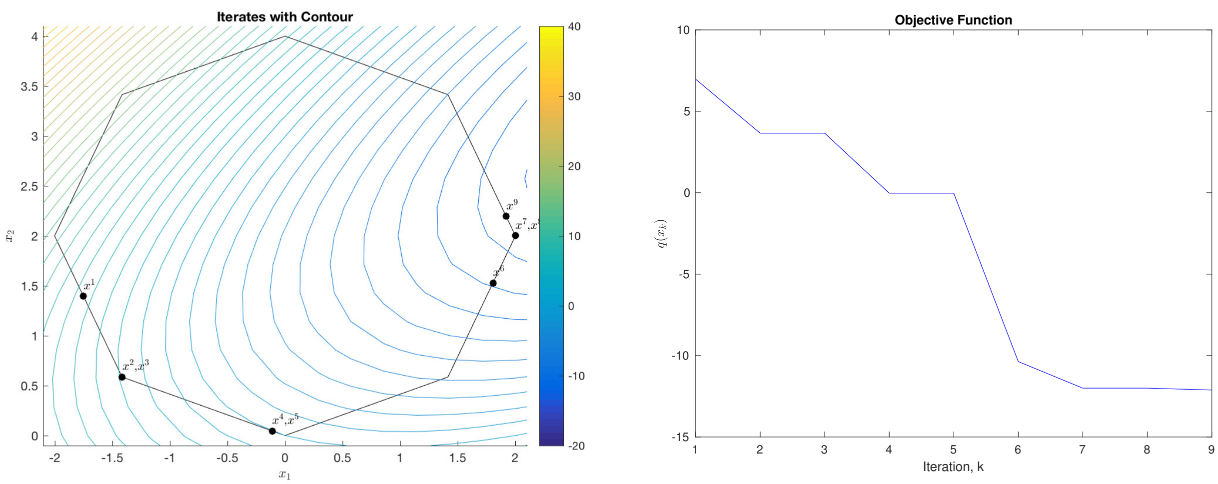
Figure 1: Tracking the objective function's descent

3.1.2 Boundary Point

Next, consider a boundary point of the feasible region as the initial point. Table 2 reports important computed values for each iteration of the algorithm while Figure 2 helps visualize the objective function's descent.

Table 2: Boundary point iterations

Iteration, k	$\mathbf{x_k}$	$\mathbf{W_k}$	$\mathbf{p_k}$	$\alpha_\mathbf{k}$
1	(-1.75, 1.396)	{5}	(0.45, -1.087)	0.75
2	(-1.414, 0.586)	{5, 6}	(0, 0)	
3	(-1.414, 0.586)	{6}	(1.302, -0.54)	1.00
4	(-0.112, 0.046)	{6}	(0, 0)	
5	(-0.112, 0.046)	{}	(5.112, 3.954)	0.37
6	(1.805, 1.529)	{8}	(0.631, 1.524)	0.31
7	(2, 2)	{1, 8}	(0, 0)	
8	(2, 2)	{1}	(-0.081, 0.195)	1.00
9	(1.919, 2.195)	{1}	(0, 0)	



(a) Contour

(b) Objective function

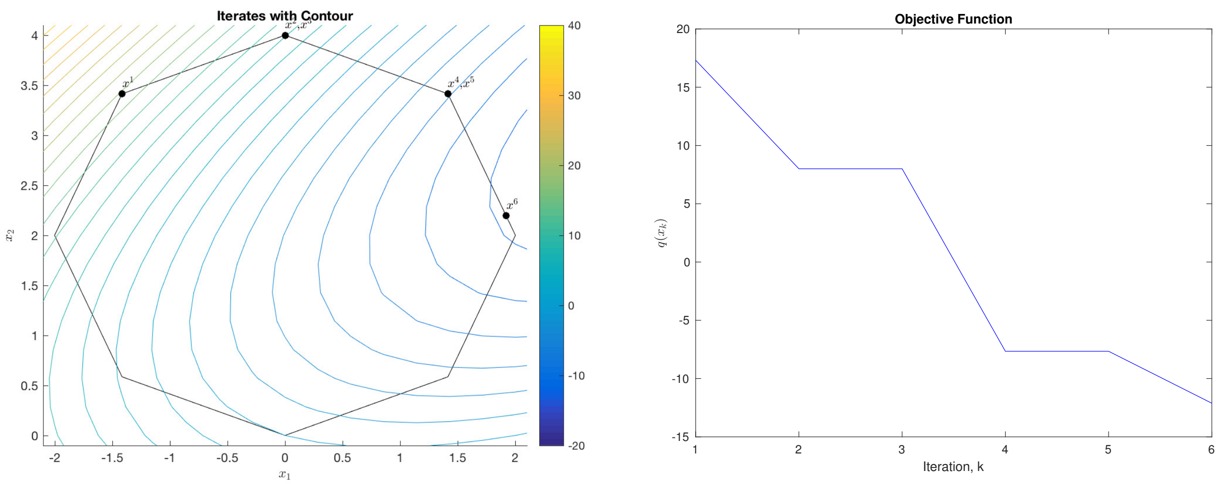
Figure 2: Tracking the objective function's descent

3.1.3 Vertex Point

Now, consider a vertex point of the feasible region as the initial point. Table 3 reports important computed values for each iteration of the algorithm while Figure 3 helps visualize the objective function's descent.

Table 3: Vertex point iterations

Iteration, k	$\mathbf{x_k}$	$\mathbf{W_k}$	$\mathbf{p_k}$	$\alpha_\mathbf{k}$
1	(-1.414, 3.414)	{3}	(7.105, 2.943)	0.20
2	(0, 4)	{2, 3}	(0, 0)	
3	(0, 4)	{2}	(3.256, -1.349)	0.43
4	(1.414, 3.414)	{1, 2}	(0, 0)	
5	(1.414, 3.414)	{1}	(0.505, -1.219)	1.00
6	(1.919, 2.195)	{1}	(0, 0)	



(a) Contour

(b) Objective function

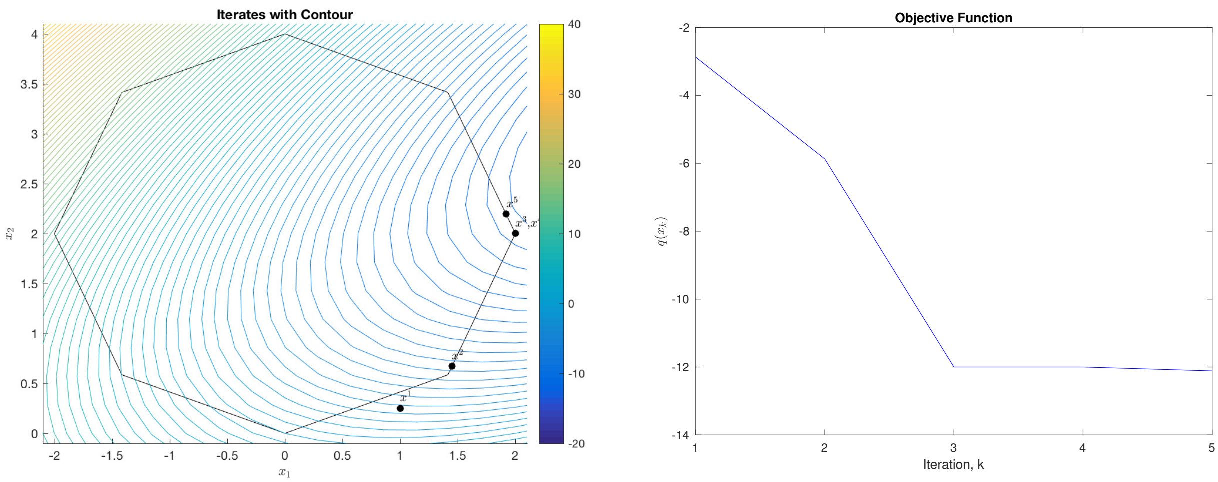
Figure 3: Tracking the objective function's descent

3.1.4 Exterior Point

Lastly, consider an exterior point of the feasible region as the initial point. Table 4 reports important computed values for each iteration of the algorithm while Figure 4 helps visualize the objective function's descent.

Table 4: Exterior point iterates.

Iteration, k	$\mathbf{x_k}$	$\mathbf{W_k}$	$\mathbf{p_k}$	$\alpha_\mathbf{k}$
1	(1, 0.25)	{}	(4, 3.75)	0.11
2	(1.45, 0.672)	{8}	(0.986, 2.381)	0.56
3	(2, 2)	{1, 8}	(0, 0)	
4	(2, 2)	{1}	(-0.081, 0.195)	1.00
5	(1.919, 2.195)	{1}	(0, 0)	



(a) Contour

(b) Objective function

Figure 4: Tracking the objective function's descent

3.2 Discussion

As the results show, the algorithm was able to quickly converge to the correct global solution, x^*. The objective function plots illustrate that for each iteration, p_k was indeed a feasible descent direction leading to the minimization of q. The algorithm performed the best with the interior point, since the working set was smaller. However, it struggled at boundary points, having to traverse the outer edge of the feasibility region to reach the solution. This illustrates the importance of choosing an appropriate initial point. The cost may not seem very high for this simple problem, but this is likely not the case for more complex problems.

It is important to note that these points were specifically chosen because they took several iterations to converge. More commonly, many interior and nearby exterior points converged in about 3 or 4 iterations. In fact, initial points were sampled uniformly for $x_1 \in [-2, 2]$ and $x_2 \in [0, 4]$. For 1000 runs, the algorithm took an average of 3.8 iterations and found x^* every time.

4 Conclusion

In this paper, we have defined the convex quadratic program (QP) and summarized its optimality conditions. Then, we discussed a method for solving convex QPs with inequality constraints, known as active-set methods. A specific algorithm was then introduced, which generates feasible iterates while decreasing the objective function. The algorithm was implemented on a particular problem with 8 inequality constraints. Results showed that the choice of the initial point does impact performance, especially when the initial point lies on the boundary of the feasible region. In the future, we can test the algorithm on problems of higher dimension and/or compare the computational time to other implementations.

References

[1] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006.