EE364a Homework 5 solutions

5.1 *A simple example.* Consider the optimization problem

$$\begin{array}{ll} \text{minimize} & x^2+1 \\ \text{subject to} & (x-2)(x-4) \leq 0, \end{array}$$

with variable $x \in \mathbf{R}$.

(a) *Analysis of primal problem.* Give the feasible set, the optimal value, and the optimal solution.

(b) *Lagrangian and dual function.* Plot the objective x^2+1 versus x. On the same plot, show the feasible set, optimal point and value, and plot the Lagrangian $L(x,\lambda)$ versus x for a few positive values of λ. Verify the lower bound property ($p^\star \geq \inf_x L(x,\lambda)$ for $\lambda \geq 0$). Derive and sketch the Lagrange dual function g.

(c) *Lagrange dual problem.* State the dual problem, and verify that it is a concave maximization problem. Find the dual optimal value and dual optimal solution $\lambda^\star$. Does strong duality hold?

(d) *Sensitivity analysis.* Let $p^\star(u)$ denote the optimal value of the problem

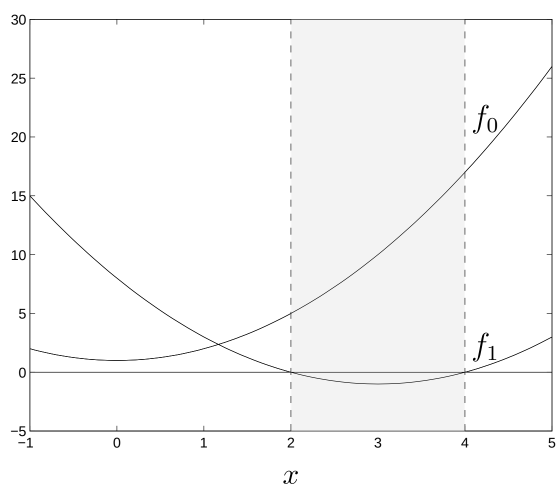
$$\begin{array}{ll} \text{minimize} & x^2+1 \\ \text{subject to} & (x-2)(x-4) \leq u, \end{array}$$

as a function of the parameter u. Plot $p^\star(u)$. Verify that $dp^\star(0)/du = -\lambda^\star$.

Solution.

(a) The feasible set is the interval $[2,4]$. The (unique) optimal point is $x^\star = 2$, and the optimal value is $p^\star = 5$.

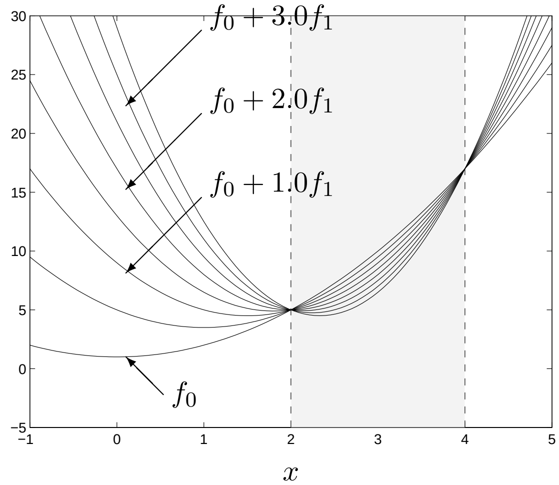
The plot shows f_0 and f_1.



(b) The Lagrangian is

$$L(x,\lambda) = (1+\lambda)x^2 - 6\lambda x + (1+8\lambda).$$

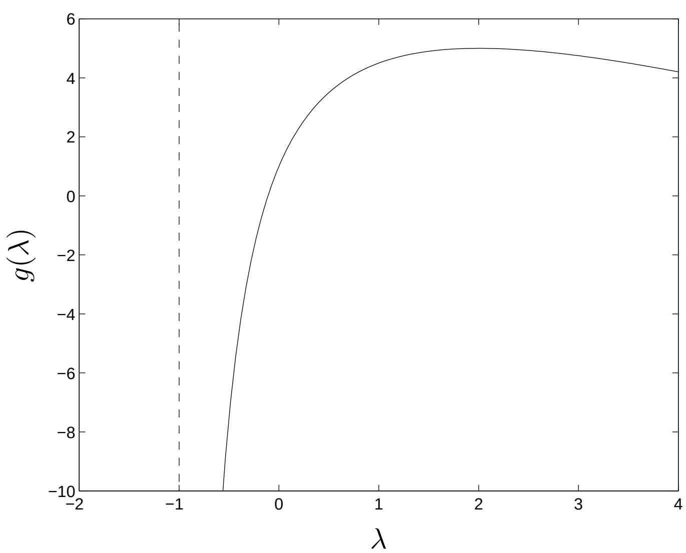
The plot shows the Lagrangian $L(x,\lambda) = f_0 + \lambda f_1$ as a function of x for different values of $\lambda \geq 0$. Note that the minimum value of $L(x,\lambda)$ over x (*i.e.*, $g(\lambda)$) is always less than $p^\star$. It increases as λ varies from 0 toward 2, reaches its maximum at $\lambda = 2$, and then decreases again as λ increases above 2. We have equality $p^\star = g(\lambda)$ for $\lambda = 2$.



For $\lambda > -1$, the Lagrangian reaches its minimum at $\tilde{x} = 3\lambda/(1+\lambda)$. For $\lambda \leq -1$ it is unbounded below. Thus

$$g(\lambda) = \begin{cases} -9\lambda^2/(1+\lambda) + 1 + 8\lambda & \lambda > -1 \\ -\infty & \lambda \leq -1 \end{cases}$$

which is plotted below.



We can verify that the dual function is concave, that its value is equal to $p^\star = 5$ for $\lambda = 2$, and less than $p^\star$ for other values of λ.

(c) The Lagrange dual problem is

$$\begin{array}{ll} \text{maximize} & -9\lambda^2/(1+\lambda) + 1 + 8\lambda \\ \text{subject to} & \lambda \geq 0. \end{array}$$

The dual optimum occurs at $\lambda = 2$, with $d^\star = 5$. So for this example we can directly observe that strong duality holds (as it must — Slater's constraint qualification is satisfied).

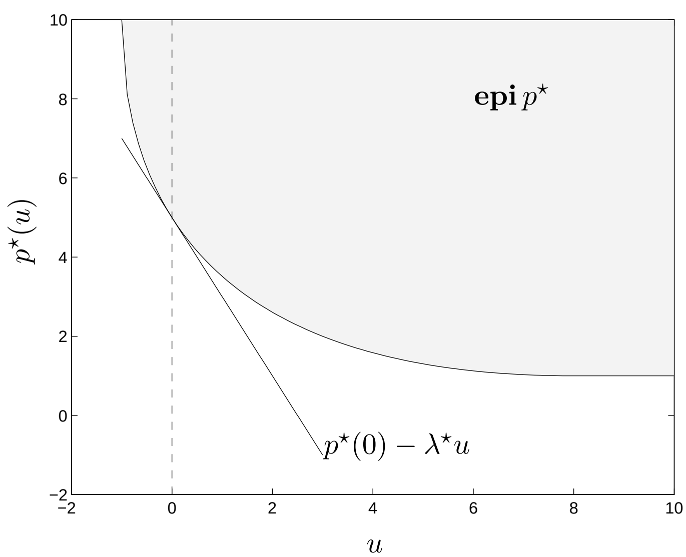
(d) The perturbed problem is infeasible for $u < -1$, since $\inf_x(x^2 - 6x + 8) = -1$. For $u \geq -1$, the feasible set is the interval

$$[3 - \sqrt{1+u}, 3 + \sqrt{1+u}],$$

given by the two roots of $x^2 - 6x + 8 = u$. For $-1 \leq u \leq 8$ the optimum is $x^\star(u) = 3 - \sqrt{1+u}$. For $u \geq 8$, the optimum is the unconstrained minimum of f_0, *i.e.*, $x^\star(u) = 0$. In summary,

$$p^\star(u) = \begin{cases} \infty & u < -1 \\ 11 + u - 6\sqrt{1+u} & -1 \leq u \leq 8 \\ 1 & u \geq 8. \end{cases}$$

The figure shows the optimal value function $p^\star(u)$ and its epigraph.



Finally, we note that $p^\star(u)$ is a differentiable function of u, and that

$$\frac{dp^\star(0)}{du} = -2 = -\lambda^\star.$$

5.13 *Lagrangian relaxation of Boolean LP.* A *Boolean linear program* is an optimization problem of the form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \\ & x_i \in \{0,1\}, \quad i=1,\ldots,n, \end{array}$$

and is, in general, very difficult to solve. In exercise 4.15 we studied the LP relaxation of this problem,

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \\ & 0 \le x_i \le 1, \quad i=1,\ldots,n, \end{array} \tag{1}$$

which is far easier to solve, and gives a lower bound on the optimal value of the Boolean LP. In this problem we derive another lower bound for the Boolean LP, and work out the relation between the two lower bounds.

(a) *Lagrangian relaxation.* The Boolean LP can be reformulated as the problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \\ & x_i(1-x_i) = 0, \quad i=1,\ldots,n, \end{array}$$

which has quadratic equality constraints. Find the Lagrange dual of this problem. The optimal value of the dual problem (which is convex) gives a lower bound on the optimal value of the Boolean LP. This method of finding a lower bound on the optimal value is called *Lagrangian relaxation.*

(b) Show that the lower bound obtained via Lagrangian relaxation, and via the LP relaxation (1), are the same. *Hint.* Derive the dual of the LP relaxation (1).

Solution.

(a) The Lagrangian is

$$\begin{array}{rcl} L(x,\mu,\nu) & = & c^T x + \mu^T(Ax-b) - \nu^T x + x^T \mathbf{diag}(\nu) x \\ & = & x^T \mathbf{diag}(\nu) x + (c + A^T\mu - \nu)^T x - b^T\mu. \end{array}$$

Minimizing over x gives the dual function

$$g(\mu,\nu) = \begin{cases} -b^T\mu - (1/4)\sum_{i=1}^n (c_i + a_i^T\mu - \nu_i)^2/\nu_i & \nu \succeq 0 \\ -\infty & \text{otherwise} \end{cases}$$

where a_i is the ith column of A, and we adopt the convention that $a^2/0 = \infty$ if $a \neq 0$, and $a^2/0 = 0$ if $a = 0$.

The resulting dual problem is

$$\begin{array}{ll} \text{maximize} & -b^T\mu - (1/4)\sum_{i=1}^n (c_i + a_i^T\mu - \nu_i)^2/\nu_i \\ \text{subject to} & \mu \succeq 0, \end{array}$$

with implicit constraint $\nu \succeq 0$.

In order to simplify this dual, we optimize analytically over ν, by noting that

$$\begin{aligned} \sup_{\nu_i \geq 0} \left(-\frac{(c_i + a_i^T\mu - \nu_i)^2}{\nu_i} \right) &= \begin{cases} 4(c_i + a_i^T\mu) & c_i + a_i^T\mu \leq 0 \\ 0 & c_i + a_i^T\mu \geq 0 \end{cases} \\ &= \min\{0, 4(c_i + a_i^T\mu)\}. \end{aligned}$$

This allows us to eliminate ν from the dual problem, and simplify it as

$$\begin{array}{ll} \text{maximize} & -b^T\mu + \sum_{i=1}^n \min\{0, c_i + a_i^T\mu\} \\ \text{subject to} & \mu \succeq 0. \end{array}$$

(b) We follow the hint. The Lagrangian and dual function of the LP relaxation are

$$\begin{aligned} L(x,u,v,w) &= c^Tx + u^T(Ax - b) - v^Tx + w^T(x - \mathbf{1}) \\ &= (c + A^Tu - v + w)^Tx - b^Tu - \mathbf{1}^Tw \\ g(u,v,w) &= \begin{cases} -b^Tu - \mathbf{1}^Tw & A^Tu - v + w + c = 0 \\ -\infty & \text{otherwise.} \end{cases} \end{aligned}$$

The dual problem is

$$\begin{array}{ll} \text{maximize} & -b^Tu - \mathbf{1}^Tw \\ \text{subject to} & A^Tu - v + w + c = 0 \\ & u \succeq 0, v \succeq 0, w \succeq 0, \end{array}$$

which is equivalent to the Lagrange relaxation problem derived above. We conclude that the two relaxations give the same value.

5.31 *Supporting hyperplane interpretation of KKT conditions.* Consider a convex problem with no equality constraints,

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m. \end{array}$$

Assume that $x^\star \in \mathbf{R}^n$ and $\lambda^\star \in \mathbf{R}^m$ satisfy the KKT conditions

$$\begin{aligned} f_i(x^\star) &\leq 0, \quad i = 1, \ldots, m \\ \lambda_i^\star &\geq 0, \quad i = 1, \ldots, m \\ \lambda_i^\star f_i(x^\star) &= 0, \quad i = 1, \ldots, m \\ \nabla f_0(x^\star) + \sum_{i=1}^m \lambda_i^\star \nabla f_i(x^\star) &= 0. \end{aligned}$$

Show that

$$\nabla f_0(x^\star)^T(x - x^\star) \geq 0$$

for all feasible x. In other words the KKT conditions imply the simple optimality criterion of §4.2.3.

Solution. Suppose x is feasible. Since f_i are convex and $f_i(x) \leq 0$ we have

$$0 \geq f_i(x) \geq f_i(x^\star) + \nabla f_i(x^\star)^T(x - x^\star), \quad i = 1, \ldots, m.$$

Using $\lambda_i^\star \geq 0$, we conclude that

$$\begin{aligned} 0 &\geq \sum_{i=1}^m \lambda_i^\star \left(f_i(x^\star) + \nabla f_i(x^\star)^T(x - x^\star) \right) \\ &= \sum_{i=1}^m \lambda_i^\star f_i(x^\star) + \sum_{i=1}^m \lambda_i^\star \nabla f_i(x^\star)^T(x - x^\star) \\ &= -\nabla f_0(x^\star)^T(x - x^\star). \end{aligned}$$

In the last line, we use the complementary slackness condition $\lambda_i^\star f_i(x^\star) = 0$, and the last KKT condition. This shows that $\nabla f_0(x^\star)^T(x - x^\star) \geq 0$, *i.e.*, $\nabla f_0(x^\star)$ defines a supporting hyperplane to the feasible set at $x^\star$.

5.39 *SDP relaxations of two-way partitioning problem.* We consider the two-way partitioning problem (5.7), described on page 219,

$$\begin{array}{ll} \text{minimize} & x^T W x \\ \text{subject to} & x_i^2 = 1, \quad i = 1, \ldots, n, \end{array} \tag{2}$$

with variable $x \in \mathbf{R}^n$. The Lagrange dual of this (nonconvex) problem is given by the SDP

$$\begin{array}{ll} \text{maximize} & -\mathbf{1}^T \nu \\ \text{subject to} & W + \mathbf{diag}(\nu) \succeq 0 \end{array} \tag{3}$$

with variable $\nu \in \mathbf{R}^n$. The optimal value of this SDP gives a lower bound on the optimal value of the partitioning problem (5.113). In this exercise we derive another SDP that gives a lower bound on the optimal value of the two-way partitioning problem, and explore the connection between the two SDPs.

(a) *Two-way partitioning problem in matrix form.* Show that the two-way partitioning problem can be cast as

$$\begin{array}{ll} \text{minimize} & \mathbf{tr}(WX) \\ \text{subject to} & X \succeq 0, \quad \mathbf{rank}\, X = 1 \\ & X_{ii} = 1, \quad i = 1, \ldots, n, \end{array}$$

with variable $X \in \mathbf{S}^n$. *Hint.* Show that if X is feasible, then it has the form $X = xx^T$, where $x \in \mathbf{R}^n$ satisfies $x_i \in \{-1, 1\}$ (and vice versa).

(b) *SDP relaxation of two-way partitioning problem.* Using the formulation in part (a), we can form the relaxation

$$\begin{array}{ll} \text{minimize} & \mathbf{tr}(WX) \\ \text{subject to} & X \succeq 0 \\ & X_{ii} = 1, \quad i = 1, \ldots, n, \end{array} \tag{4}$$

with variable $X \in \mathbf{S}^n$. This problem is an SDP, and therefore can be solved efficiently. Explain why its optimal value gives a lower bound on the optimal value of the two-way partitioning problem (2). What can you say if an optimal point $X^\star$ for this SDP has rank one?

(c) We now have two SDPs that give a lower bound on the optimal value of the two-way partitioning problem (2): the SDP relaxation (4) found in part (b), and the Lagrange dual of the two-way partitioning problem, given in (3). What is the relation between the two SDPs? What can you say about the lower bounds found by them? *Hint:* Relate the two SDPs via duality.

Solution.

(a) Follows from $\mathbf{tr}(Wxx^T) = x^T W x$ and $(xx^T)_{ii} = x_i^2$.

(b) It gives a lower bound because we minimize the same objective over a larger set. If X is rank one, it is optimal.

(c) We write the problem as a minimization problem

$$\begin{array}{ll} \text{minimize} & \mathbf{1}^T \nu \\ \text{subject to} & W + \mathbf{diag}(\nu) \succeq 0. \end{array}$$

Introducing a Lagrange multiplier $X \in \mathbf{S}^n$ for the matrix inequality, we obtain the Lagrangian

$$\begin{aligned} L(\nu, X) &= \mathbf{1}^T \nu - \mathbf{tr}(X(W + \mathbf{diag}(\nu))) \\ &= \mathbf{1}^T \nu - \mathbf{tr}(XW) - \sum_{i=1}^n \nu_i X_{ii} \\ &= -\mathbf{tr}(XW) + \sum_{i=1}^n \nu_i (1 - X_{ii}). \end{aligned}$$

This is bounded below as a function of ν only if $X_{ii} = 1$ for all i, so we obtain the dual problem

$$\begin{array}{ll} \text{maximize} & -\mathbf{tr}(WX) \\ \text{subject to} & X \succeq 0 \\ & X_{ii} = 1, \quad i = 1, \ldots, n. \end{array}$$

Changing the sign again, and switching from maximization to minimization, yields the problem in part (a).

A3.18 *Heuristic suboptimal solution for Boolean LP.* This exercise builds on exercises 4.15 and 5.13 in *Convex Optimization*, which involve the Boolean LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \\ & x_i \in \{0,1\}, \quad i = 1, \ldots, n, \end{array}$$

with optimal value $p^\star$. Let x^{rlx} be a solution of the LP relaxation

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \\ & 0 \preceq x \preceq \mathbf{1}, \end{array}$$

so $L = c^T x^{\text{rlx}}$ is a lower bound on $p^\star$. The relaxed solution x^{rlx} can also be used to guess a Boolean point $\hat{x}$, by rounding its entries, based on a threshold $t \in [0,1]$:

$$\hat{x}_i = \begin{cases} 1 & x_i^{\text{rlx}} \geq t \\ 0 & \text{otherwise}, \end{cases}$$

for $i = 1, \ldots, n$. Evidently $\hat{x}$ is Boolean (*i.e.*, has entries in $\{0,1\}$). If it is feasible for the Boolean LP, *i.e.*, if $A\hat{x} \preceq b$, then it can be considered a guess at a good, if not optimal, point for the Boolean LP. Its objective value, $U = c^T \hat{x}$, is an upper bound on $p^\star$. If U and L are close, then $\hat{x}$ is nearly optimal; specifically, $\hat{x}$ cannot be more than $(U - L)$-suboptimal for the Boolean LP.

This rounding need not work; indeed, it can happen that for all threshold values, $\hat{x}$ is infeasible. But for some problem instances, it can work well.

Of course, there are many variations on this simple scheme for (possibly) constructing a feasible, good point from x^{rlx}.

Finally, we get to the problem. Generate problem data using

```
rand('state',0);
n=100;
m=300;
A=rand(m,n);
b=A*ones(n,1)/2;
c=-rand(n,1);
```

You can think of x_i as a job we either accept or decline, and $-c_i$ as the (positive) revenue we generate if we accept job i. We can think of $Ax \preceq b$ as a set of limits on m resources. A_{ij}, which is positive, is the amount of resource i consumed if we accept job j; b_i, which is positive, is the amount of resource i available.

Find a solution of the relaxed LP and examine its entries. Note the associated lower bound L. Carry out threshold rounding for (say) 100 values of t, uniformly spaced over $[0, 1]$. For each value of t, note the objective value $c^T \hat{x}$ and the maximum constraint violation $\max_i(A\hat{x} - b)_i$. Plot the objective value and the maximum violation versus t. Be sure to indicate on the plot the values of t for which $\hat{x}$ is feasible, and those for which it is not.

Find a value of t for which $\hat{x}$ is feasible, and gives minimum objective value, and note the associated upper bound U. Give the gap $U - L$ between the upper bound on $p^\star$ and the lower bound on $p^\star$. If you define vectors `obj` and `maxviol`, you can find the upper bound as `U=min(obj(find(maxviol<=0)))`.

Solution. The following Matlab code finds the solution

```
% generate data for boolean LP relaxation & heuristic
rand('state',0);
n=100;
m=300;
A=rand(m,n);
b=A*ones(n,1)/2;
c=-rand(n,1);

% solve LP relaxation
cvx_begin
   variable x(n)
   minimize (c'*x)
   subject to
       A*x <= b
       x>=0
       x<=1
cvx_end
xrlx = x;
L=cvx_optval;

% sweep over threshold & round
thres=0:0.01:1;
maxviol = zeros(length(thres),1);
obj = zeros(length(thres),1);
for i=1:length(thres)
   xhat = (xrlx>=thres(i));
   maxviol(i) = max(A*xhat-b);
   obj(i) = c'*xhat;
end
```

```
% find least upper bound and associated threshold
i_feas=find(maxviol<=0);
U=min(obj(i_feas))
%U=min(obj(find(maxviol <=0)))
t=min(i_feas);
min_thresh=thres(t)

% plot objective and max violation versus threshold
subplot(2,1,1)
plot(thres(1:t-1),maxviol(1:t-1),'r',thres(t:end),maxviol(t:end),'b','linewidth',2)
xlabel('threshold');
ylabel('max violation');
subplot(2,1,2)
hold on; plot(thres,L*ones(size(thres)),'k','linewidth',2);
plot(thres(1:t-1),obj(1:t-1),'r',thres(t:end),obj(t:end),'b','linewidth',2);
xlabel('threshold');
ylabel('objective');
```

The lower bound found from the relaxed LP is $L = -33.1672$. We find that the threshold value $t = 0.6006$ gives the best (smallest) objective value for feasible $\hat{x}$: $U = -32.4450$. The difference is 0.7222. So $\hat{x}$, with $t = 0.6006$, can be no more than 0.7222 suboptimal, *i.e.*, around 2.2% suboptimal.

In figure 1, the red lines indicate values for thresholding values which give infeasible $\hat{x}$, and the blue lines correspond to feasible $\hat{x}$. We see that the maximum violation decreases as the threshold is increased. This occurs because the constraint matrix A only has nonnegative entries. At a threshold of 0, all jobs are selected, which is an infeasible solution. As we increase the threshold, projects are removed in sequence (without adding new projects), which monotonically decreases the maximum violation. For a general boolean LP, the corresponding plots need not exhibit monotonic behavior.

A4.1 *Numerical perturbation analysis example.* Consider the quadratic program

$$\begin{array}{ll} \text{minimize} & x_1^2 + 2x_2^2 - x_1x_2 - x_1 \\ \text{subject to} & x_1 + 2x_2 \leq u_1 \\ & x_1 - 4x_2 \leq u_2, \\ & 5x_1 + 76x_2 \leq 1, \end{array}$$

with variables x_1, x_2, and parameters u_1, u_2.

(a) Solve this QP, for parameter values $u_1 = -2$, $u_2 = -3$, to find optimal primal variable values $x_1^\star$ and $x_2^\star$, and optimal dual variable values $\lambda_1^\star$, $\lambda_2^\star$ and $\lambda_3^\star$. Let $p^\star$ denote the optimal objective value. Verify that the KKT conditions hold for the optimal primal and dual variables you found (within reasonable numerical accuracy).

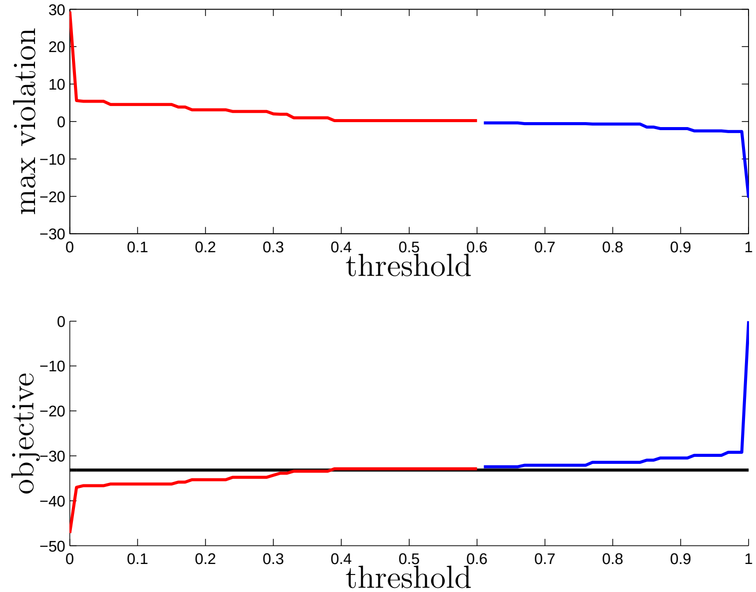


Figure 1 Plots of violation and objective vs threshold rule.

Hint: See §3.7 of the CVX users' guide to find out how to retrieve optimal dual variables. To specify the quadratic objective, use `quad_form()`.

(b) We will now solve some perturbed versions of the QP, with

$$u_1 = -2 + \delta_1, \qquad u_2 = -3 + \delta_2,$$

where δ_1 and δ_2 each take values from $\{-0.1, 0, 0.1\}$. (There are a total of nine such combinations, including the original problem with $\delta_1 = \delta_2 = 0$.) For each combination of δ_1 and δ_2, make a prediction $p^\star_{\rm pred}$ of the optimal value of the perturbed QP, and compare it to $p^\star_{\rm exact}$, the exact optimal value of the perturbed QP (obtained by solving the perturbed QP). Put your results in the two righthand columns in a table with the form shown below. Check that the inequality $p^\star_{\rm pred} \leq p^\star_{\rm exact}$ holds.

δ_1	δ_2	$p^\star_{\rm pred}$	$p^\star_{\rm exact}$
0	0		
0	-0.1		
0	0.1		
-0.1	0		
-0.1	-0.1		
-0.1	0.1		
0.1	0		
0.1	-0.1		
0.1	0.1		

Solution.

(a) The following Matlab code sets up the simple QP and solves it using CVX:

```
Q = [1 -1/2; -1/2 2];
f = [-1 0]';
A = [1 2; 1 -4; 5 76];
b = [-2 -3 1]';

cvx_begin
    variable x(2)
    dual variable lambda
    minimize(quad_form(x,Q)+f'*x)
    subject to
        lambda: A*x <= b
cvx_end
p_star = cvx_optval
```

When we run this, we find the optimal objective value is $p^\star = 8.22$ and the optimal point is $x_1^\star = -2.33$, $x_2^\star = 0.17$. (This optimal point is unique since the objective is strictly convex.) A set of optimal dual variables is $\lambda_1^\star = 2.13$, $\lambda_2^\star = 3.31$ and $\lambda_3^\star = 0.08$. (The dual optimal point is unique too, but it's harder to show this, and it doesn't matter anyway.)

The KKT conditions are

$$\begin{array}{lll} x_1^\star + 2x_2^\star \leq u_1, & x_1^\star - 4x_2^\star \leq u_2, & 5x_1^\star + 76x_2^\star \leq 1 \\ \lambda_1^\star \geq 0, & \lambda_2^\star \geq 0, & \lambda_3^\star \geq 0 \\ \lambda_1^\star(x_1^\star + 2x_2^\star - u_1) = 0, & \lambda_2^\star(x_1^\star - 4x_2^\star - u_2) = 0, & \lambda_3^\star(5x_1^\star + 76x_2^\star - 1) = 0, \\ 2x_1^\star - x_2^\star - 1 + \lambda_1^\star + \lambda_2^\star + 5\lambda_3^\star = 0, & & \\ 4x_2^\star - x_1^\star + 2\lambda_1^\star - 4\lambda_2^\star + 76\lambda_3^\star = 0. & & \end{array}$$

We check these numerically. The dual variable $\lambda_1^\star$, $\lambda_2^\star$ and $\lambda_3^\star$ are all greater than zero and the quantities

```
A*x-b
2*Q*x+f+A'*lambda
```

are found to be very small. Thus the KKT conditions are verified.

(b) The predicted optimal value is given by

$$p_{\rm pred}^\star = p^\star - \lambda_1^\star \delta_1 - \lambda_2^\star \delta_2.$$

The following Matlab code fills in the table

```
arr_i = [0 -1 1];
delta = 0.1;
pa_table = [];
for i = arr_i
    for j = arr_i
        p_pred = p_star - [lambda(1) lambda(2)]*[i; j]*delta;
        cvx_begin
            variable x(2)
            minimize(quad_form(x,Q)+f'*x)
            subject to
                A*x <= b+[i;j;0]*delta
        cvx_end
        p_exact = cvx_optval;

        pa_table = [pa_table; i*delta j*delta p_pred p_exact]
    end
end
```

The values obtained are

δ_1	δ_2	$p^\star_{\text{pred}}$	$p^\star_{\text{exact}}$
0	0	8.22	8.22
0	-0.1	8.55	8.70
0	0.1	7.89	7.98
-0.1	0	8.44	8.57
-0.1	-0.1	8.77	8.82
-0.1	0.1	8.10	8.32
0.1	0	8.01	8.22
0.1	-0.1	8.34	8.71
0.1	0.1	7.68	7.75

The inequality $p^\star_{\text{pred}} \leq p^\star_{\text{exact}}$ is verified to be true in all cases.

A4.21 *Robust LP with polyhedral cost uncertainty.* We consider a robust linear programming problem, with polyhedral uncertainty in the cost:

$$\begin{array}{ll} \text{minimize} & \sup_{c\in\mathcal{C}} c^T x \\ \text{subject to} & Ax \succeq b, \end{array}$$

with variable $x \in \mathbf{R}^n$, where $\mathcal{C} = \{c \mid Fc \preceq g\}$. You can think of x as the quantities of n products to buy (or sell, when $x_i < 0$), $Ax \succeq b$ as constraints, requirements, or limits on the available quantities, and $\mathcal{C}$ as giving our knowledge or assumptions about the product prices at the time we place the order. The objective is then the worst possible (*i.e.*, largest) possible cost, given the quantities x, consistent with our knowledge of the prices.

In this exercise, you will work out a tractable method for solving this problem. You can assume that $\mathcal{C} \neq \emptyset$, and the inequalities $Ax \succeq b$ are feasible.

(a) Let $f(x) = \sup_{c \in \mathcal{C}} c^T x$ be the objective in the problem above. Explain why f is convex.

(b) Find the dual of the problem

$$\begin{array}{ll} \text{maximize} & c^T x \\ \text{subject to} & Fc \preceq g, \end{array}$$

with variable c. (The problem data are x, F, and g.) Explain why the optimal value of the dual is $f(x)$.

(c) Use the expression for $f(x)$ found in part (b) in the original problem, to obtain a single LP equivalent to the original robust LP.

(d) Carry out the method found in part (c) to solve a robust LP with data

```
rand('seed',0);
A = rand(30,10);
b = rand(30,1);
c_nom = 1+rand(10,1);  % nominal c values
```

and $\mathcal{C}$ described as follows. Each c_i deviates no more than 25% from its nominal value, *i.e.*, $0.75c_{\text{nom}} \preceq c \preceq 1.25c_{\text{nom}}$, and the average of c does not deviate more than 10% from the average of the nominal values, *i.e.*, $0.9(\mathbf{1}^T c_{\text{nom}})/n \leq \mathbf{1}^T c/n \leq 1.1(\mathbf{1}^T c_{\text{nom}})/n$.

Compare the worst-case cost $f(x)$ and the nominal cost $c_{\text{nom}}^T x$ for x optimal for the robust problem, and for x optimal for the nominal problem (*i.e.*, the case where $\mathcal{C} = \{c_{\text{nom}}\}$). Compare the values and make a brief comment.

Solution.

(a) For each $c \in \mathcal{C}$, $c^T x$ is linear, therefore convex. $f(x)$ is the supremum of convex functions, and therefore convex.

(b) The dual is

$$\begin{array}{ll} \text{minimize} & \lambda^T g \\ \text{subject to} & F^T \lambda = x, \quad \lambda \succeq 0, \end{array}$$

with variable λ. Since the primal problem is feasible (we know this since we assume $\mathcal{C} \neq \emptyset$), we are guaranteed there is zero duality gap, so $f(x)$, the optimal value of the primal problem, is also the optimal value of the dual above.

(c) Substituting our expression for $f(x)$ into the original problem, we arrive at

$$\begin{array}{ll} \text{minimize} & \inf\{\lambda^T g \mid F^T\lambda = x,\ \lambda \succeq 0\} \\ \text{subject to} & Ax \succeq b. \end{array}$$

We can just as well minimize over x and λ at the same time, which gives the problem

$$\begin{array}{ll} \text{minimize} & \lambda^T g \\ \text{subject to} & Ax \succeq b, \quad F^T\lambda = x, \quad \lambda \succeq 0, \end{array}$$

which is an LP in the variables x and λ. Solving this single LP gives us the optimal x for the original robust LP.

(d) We define x_{nom} and x_{rob} to be the nominal and robust solutions respectively. The numerical results are given in the table below.

	x_{nom}	x_{rob}
$c_{\mathrm{nom}}^T x$	1.50	1.94
$f(x)$	4.07	2.31

The code below formulates and solve the robust diet problem.

```
rand('seed',0);
n=10;m=30;
A = rand(m,n);
b = rand(m,1);
c_nom = 1+1*rand(n,1); % nominal c values

F = [eye(n);-eye(n);ones(1,n)/n;-ones(1,n)/n];
g = [1.25*c_nom;-0.75*c_nom;1.1*sum(c_nom)/n;-0.9*sum(c_nom)/n];
k = length(g);

% robust LP
cvx_begin
variables x_rob(n) lambda(k)
minimize(lambda'*g)
A*x_rob>=b
lambda>=0
F'*lambda==x_rob
cvx_end

% nominal cost of x_rob
c_nom'*x_rob

% nominal LP
cvx_begin
```

```
variables x_nom(n)
minimize(c_nom'*x_nom)
A*x_nom>=b
cvx_end

% worst case cost of x_nom
cvx_begin
variables c_wc(n)
maximize(c_wc'*x_nom)
F*c_wc<=g
cvx_end
```

A16.9 *Energy storage trade-offs.* We consider the use of a storage device (say, a battery) to reduce the total cost of electricity consumed over one day. We divide the day into T time periods, and let p_t denote the (positive, time-varying) electricity price, and u_t denote the (nonnegative) usage or consumption, in period t, for $t = 1, \ldots, T$. Without the use of a battery, the total cost is $p^T u$.

Let q_t denote the (nonnegative) energy stored in the battery in period t. For simplicity, we neglect energy loss (although this is easily handled as well), so we have $q_{t+1} = q_t + c_t$, $t = 1, \ldots, T-1$, where c_t is the charging of the battery in period t; $c_t < 0$ means the battery is discharged. We will require that $q_1 = q_T + c_T$, *i.e.*, we finish with the same battery charge that we start with. With the battery operating, the net consumption in period t is $u_t + c_t$; we require this to be nonnegative (*i.e.*, we do not pump power back into the grid). The total cost is then $p^T(u + c)$.

The battery is characterized by three parameters: The capacity Q, where $q_t \leq Q$; the maximum charge rate C, where $c_t \leq C$; and the maximum discharge rate D, where $c_t \geq -D$. (The parameters Q, C, and D are nonnegative.)

(a) Explain how to find the charging profile $c \in \mathbf{R}^T$ (and associated stored energy profile $q \in \mathbf{R}^T$) that minimizes the total cost, subject to the constraints.

(b) Solve the problem instance with data p and u given in `storage_tradeoff_data.m`, $Q = 35$, and $C = D = 3$. Plot u_t, p_t, c_t, and q_t versus t.

(c) *Storage trade-offs.* Plot the minimum total cost versus the storage capacity Q, using p and u from `storage_tradeoff_data.m`, and charge/discharge limits $C = D = 3$. Repeat for charge/discharge limits $C = D = 1$. (Put these two trade-off curves on the same plot.) Give an interpretation of the endpoints of the trade-off curves.

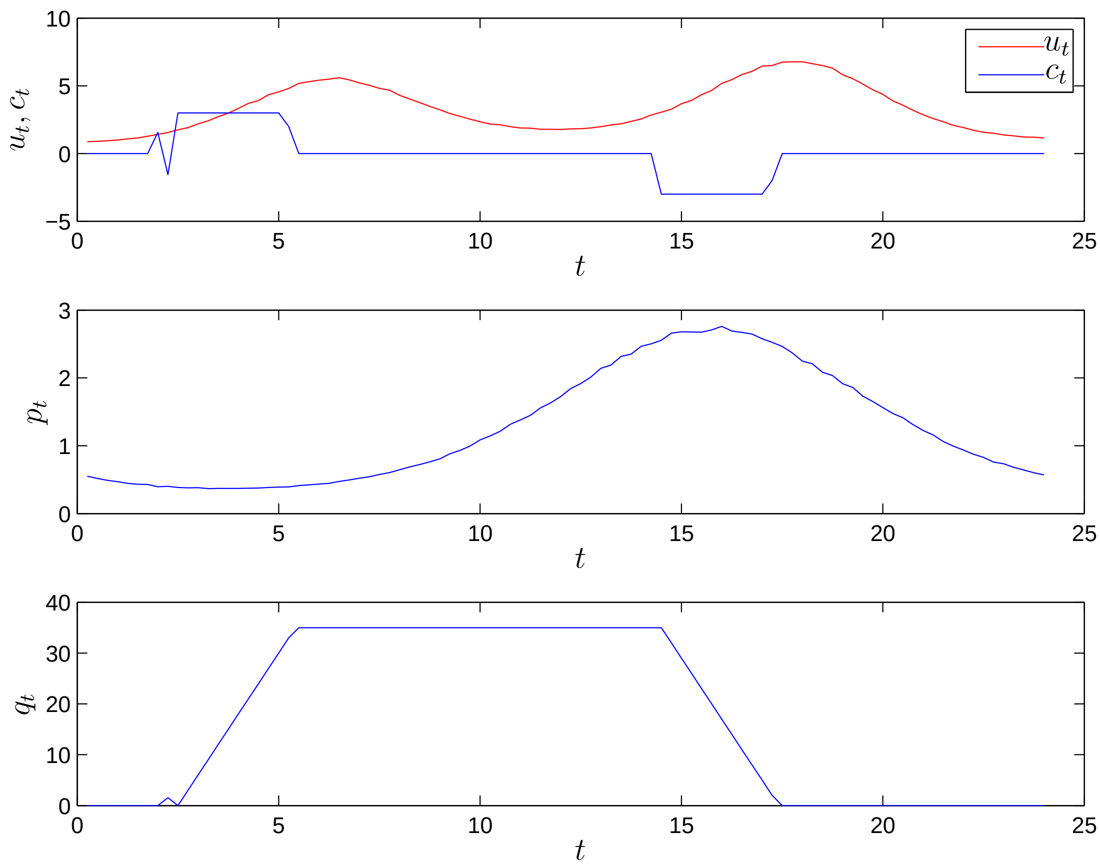
Solution.

(a) The problem is an LP,

$$
\begin{array}{ll}
\text{minimize} & p^T(u+c) \\
\text{subject to} & -D\mathbf{1} \preceq c \preceq C\mathbf{1}, \quad u+c \succeq 0, \quad 0 \preceq q \preceq Q\mathbf{1} \\
 & q_{t+1} = q_t + c_t, \quad t = 1, \ldots, T \\
q_1 = q_T + c_T &
\end{array}
$$

with variables $c,\ q \in \mathbf{R}^T$.

(b) The code for the problem is given in part (c). The results are shown below.



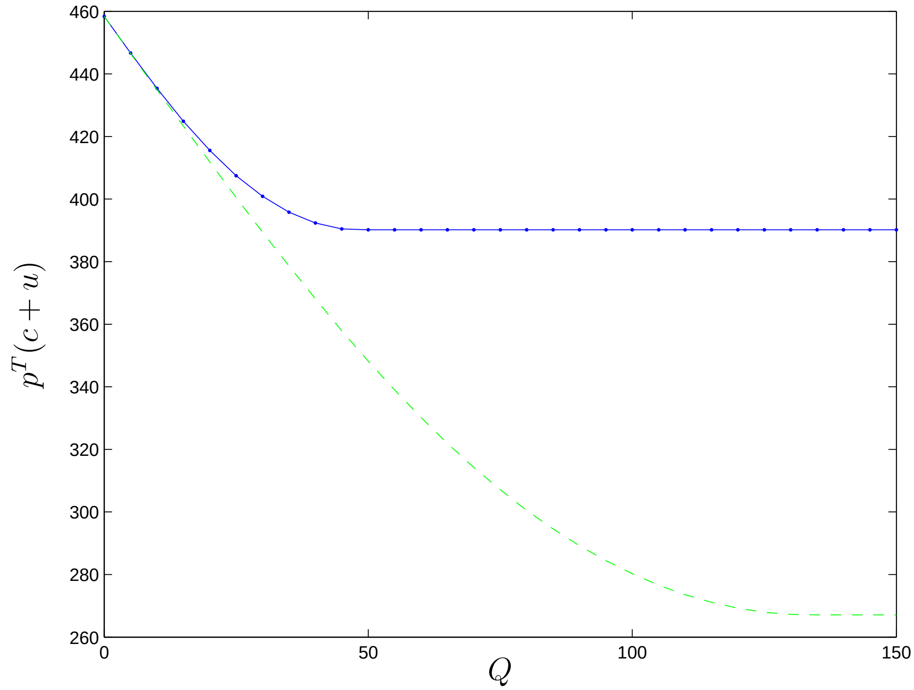
(c) We vary the range of Q from 1 to 150 and solve the LP for the two cases of high charge/discharge limit and low charge/discharge limit. The matlab code is given below.

```
clear all; close all; storage_tradeoff_data;
Q = 35;
C = 3; D = 3;
cvx_quiet(true);
cvx_begin
```

```
    variables q(T,1) c(T,1);
    minimize(p'*(u+c));
    subject to
    c >= -D;  c<= C;
    q >= 0;  q <= Q;
    q(2:T) == q(1:T-1) + c(1:T-1);
    q(1) == q(T) + c(T);
    u+c >= 0;
cvx_end

figure;
ts = (1:T)/4;
subplot(3,1,1);
plot(ts,u, 'r'); hold on
plot(ts,c,'b');
legend('u','c');
xlabel('t');
ylabel('uc');
subplot(3,1,2);
plot(ts, p);
ylabel('pt');
xlabel('t');
subplot(3,1,3);
plot(ts,q);
ylabel('qt');
xlabel('t');
print -depsc storage_tradeoff_time_trace.eps

%% Plot the trade-off curves
N = 31; Qs = linspace(0, 150,N);
C = 1; D = 1;
cvx_quiet(true);
for i=1:N
    Q = Qs(i);
    cvx_begin
        variables q(T,1) c(T,1);
        minimize(p'*(u+c));
        subject to
        c >= -D;         c <= C;
        q >= 0;          q <= Q;
        q(2:T) == q(1:T-1) + c(1:T-1);
        q(1) == q(T) + c(T);
```

```
        u + c >= 0;
    cvx_end
    qStore1(:,i) = q;
    cStore1(:,i) = c;
    cost1(i) = cvx_optval;
end
figure;
plot(Qs,cost1, 'b.-');
hold on

C = 3; D = 3;
for i=1:N
    Q = Qs(i);
    cvx_begin
        variables q(T,1) c(T,1);
        minimize(p'*(u+c));
        subject to
        c >= -D;        c <= C;
        q >= 0;         q <= Q;
        q(2:T) == q(1:T-1) + c(1:T-1);
        q(1) == q(T) + c(T);
        u + c >= 0;
    cvx_end
    qStore2(:,i) = q;
    cStore2(:,i) = c;
    cost2(i) = cvx_optval;
end
plot(Qs,cost2, 'g--');
xlabel('Q');
ylabel('cost');
print -depsc storage_tradeoff_curve.eps
```

The trade-off curves are shown below, where the blue solid curve corresponds to $C = D = 1$ and the green dashed curve corresponds to $C = D = 3$. The intersection of the trade-off curves with the y axis (corresponding to $Q = 0$) gives the total cost if there were no battery, $p^T u$. On the right end of the trade-off curves, the battery capacity constraint is no longer active, so no further reduction in total cost is obtained. The total cost reduction here is limited by the charge/discharge limits.



460
440
420
400
380
360
340
320
300
280
260
0
50
100
150
Q
$p^T(c+u)$