Brian Knott

EE364A

Homework #5

Problem 6.9

For each i:

$\frac{p(t_i)}{q(t_i)} - y_i \leq \lambda$ is a sublevel set of $f(t_i)$

for $f(t_i) = \frac{p(t_i)}{q(t_i)} - y_i$.

This sublevelset is equivalent to the sublevelset

$g(t_i) \leq \lambda + y_i$ for $g(t_i) = \frac{p(t_i)}{q(t_i)}$.

Since y_i is constant in this problem, we can consider the value of the sublevelset, $\lambda + y_i$, as another constant, c.

$\rightarrow \quad g(t_i) \leq c \rightarrow p(t_i) \leq c \cdot q(t_i)$

$p(t_i) = a^T \vec{t}$ for $a = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{bmatrix}$, $\vec{t} = \begin{bmatrix} 1 \\ t \\ t^2 \\ \vdots \\ t^m \end{bmatrix}$,

$q(t_i) = b^T \vec{t}$ for $b = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_m \end{bmatrix}$, $b_0 = 1$.

This simplifies to the inequality:

$$g(t_i) \le c \Rightarrow \vec{a}^T \vec{t} \le c \cdot \vec{b}^T \vec{t}$$

$$\Rightarrow (a - c \cdot b)^T \vec{t} \le 0.$$

Since exponentiation is a monotonic operation, increases in t cannot lead to decreases in any argument of $\vec{t} = \begin{bmatrix} t^0 \\ t^1 \\ \vdots \\ t^m \end{bmatrix}$. Therefore this linear inequality implies that the sublevel sets of $g(t_i)$ are convex sets and therefore $g(t_i)$ is quasiconvex. Because $f(t_i) = g(t_i) - y_i$ this implies $f(t_i)$ is quasiconvex. Because the maximum of quasiconvex functions is quasiconvex and $f(t_i)$ is quasiconvex for each $i = 1, \dots, k$:

$(\vec{a} - c \cdot \vec{b})^T \vec{t} \le 0$ convex in t

$\Rightarrow g(t_i)$ quasiconvex $\forall i$

$\Rightarrow f(t_i)$ quasiconvex $\forall i$

$\Rightarrow \max_{i=1,\dots,k} |f(t_i)| = \max_{i=1,\dots,k} \left| \frac{p(t_i)}{q(t_i)} - y_i \right|$ quasiconvex

Note: $|f(t_i)|$ is always quasiconvex when $f(t_i)$ is quasilinear since sublevel sets of $|f(t_i)| \le \lambda \Leftrightarrow -\lambda \le f(t_i) \le \lambda$ so both sublevelsets and superlevelsets ($f(t_i) \le \lambda$ and $f(t_i) \ge -\lambda$) being convex sets will imply $|f(t_i)|$ quasiconvex. $f(t_i)$ is quasilinear for each i since

$$g(t_i) \le c \quad , \quad g(t_i) \ge c$$

$$\Rightarrow a^T\vec{t} \le c b^T \vec{t} \quad , \quad \Rightarrow a^T\vec{t} \ge c b^T\vec{t}$$

$$(a-cb)^T\vec{t} \le 0 \quad , \quad \rightarrow (a-cb)^T\vec{t} \ge 0$$

These are both convex sets in t since they are linear inequalities so $g(t_i)$ is quasilinear so

$$\Rightarrow f(t_i) \text{ is quasilinear.}$$

$$\rightarrow |f(t_i)| \quad \text{quasiconvex}$$

$$\Rightarrow \boxed{\max_{i=1,k} |f(t_i)| \quad \text{quasiconvex}}$$

Problem A3.91

(a) To solve the complex, least l_2-norm linearly constrained problem

$$\begin{aligned} &\text{minimize} \quad \|x\|_2 \\ &\text{sub to} \quad Ax = b \end{aligned}$$

with $x \in \mathbb{C}^n$, $A \in \mathbb{C}^{m\times n}$, $b \in \mathbb{C}^m$, we can split x into its real and imaginary components:

$$X_c = \begin{bmatrix} | \\ \text{real}(x) \\ | \\ | \\ \text{imag}(x) \\ | \end{bmatrix} \in \mathbb{R}^{2n}.$$

Since x_c is now real and we know for complex vectors

$$a^T x = b \Rightarrow \text{real}(a)^T \text{real}(x) - \text{imag}(a)^T \text{imag}(x) = \text{real}(b)$$
$$\text{and } \text{real}(a)^T \text{imag}(x) + \text{imag}(a)^T \text{real}(x) = \text{imag}(b).$$

Therefore, for $x_c = \begin{bmatrix} \mathcal{R}(x) \\ \vdots \\ \mathcal{I}(x) \end{bmatrix}$, $b_c = \begin{bmatrix} \mathcal{R}(b) \\ \vdots \\ \mathcal{I}(b) \end{bmatrix}$

$$b_c = \begin{bmatrix} \mathcal{R}(A) & -\mathcal{I}(A) \\ \mathcal{I}(A) & \mathcal{R}(A) \end{bmatrix} x_c$$

Let this be denoted $b_c = A_c x_c$.

Now b_c, A_c, and x_c are all real valued:

$b_c \in \mathbb{R}^{2m}$, $x_c \in \mathbb{R}^{2n}$, $A \in \mathbb{R}^{2m \times 2n}$

Additionally, for complex x, $\|x\|_2 = \sqrt{x^* x} = \sqrt{x_c^T x_c} = \|x_c\|_2$

Therefore the problem reduces to

$$\begin{array}{ll} \text{minimize} & \|x_c\|_2 \\ \text{sub to} & A_c x_c = b_c \end{array} \Rightarrow \begin{array}{ll} \text{min.} & x_c^T x_c \\ \text{s.t} & A_c x_c = b_c \end{array}$$

We can then construct x from x_c by:

x = xc(1:n) + i*xc(n+1:2*n);

Note: I happened to use the notation x_c instead of the suggested notation z. I will use z from now on

(b) The same constraint must hold for the ∞-norm case: $A_c z = b_c$. However calculation of the norm is not as straightforward.

The problem now is:

$$\begin{aligned} &\min.\ \|x\|_\infty \\ &\text{s.t. } A_c z = b_c \\ &(z^T = [Rx, Ix]) \end{aligned} \Rightarrow \begin{aligned} &\min.\ \max_i |x_i| \\ &\text{s.t. } A_c z = b_c \\ &(z^T = [Rx, Ix]) \end{aligned} \Rightarrow$$

$$\Rightarrow \begin{aligned} &\min.\ \max_i |x_i|^2 \\ &\text{s.t. } A_c z = b_c \\ &(z^T = [Rx, Ix]) \end{aligned} \Rightarrow \begin{aligned} &\min.\ \max_i (z_i^2 + z_{n+i}^2) \text{ for } i=1,2,\dots,n \\ &\text{s.t. } A_c z = b_c \end{aligned}$$

$$\Rightarrow \begin{aligned} &\min.\ t \\ &\text{s.t. } z_i^2 + z_{n+i}^2 \le t \quad \forall i=1,\dots,n \\ &A_c z = b_c \end{aligned} \Rightarrow \begin{aligned} &\min.\ t \\ &\text{s.t. } z^T \begin{bmatrix} e_i & 0 \\ 0 & e_i \end{bmatrix} \begin{bmatrix} -e_i^T- & -0- \\ -0- & -e_i^T- \end{bmatrix} z \le t \\ &A_c z = b_c \quad \forall i=1,\dots,n \end{aligned}$$

$$\Rightarrow \begin{aligned} &\min.\ t \\ &\text{s.t. } \left\| \begin{bmatrix} -e_i^T- & -0- \\ -0- & -e_i^T- \end{bmatrix} z \right\|_2^2 \le t \\ &A_c z = b_c \quad \forall i=1,\dots,n \end{aligned} \Rightarrow \begin{aligned} &\min.\ y \\ &\text{s.t. } \left\| \begin{bmatrix} -e_i^T- & -0- \\ -0- & -e_i^T- \end{bmatrix} z \right\|_2 \le y \quad \forall i=1,\dots,n \\ &A_c z = b_c \end{aligned}$$

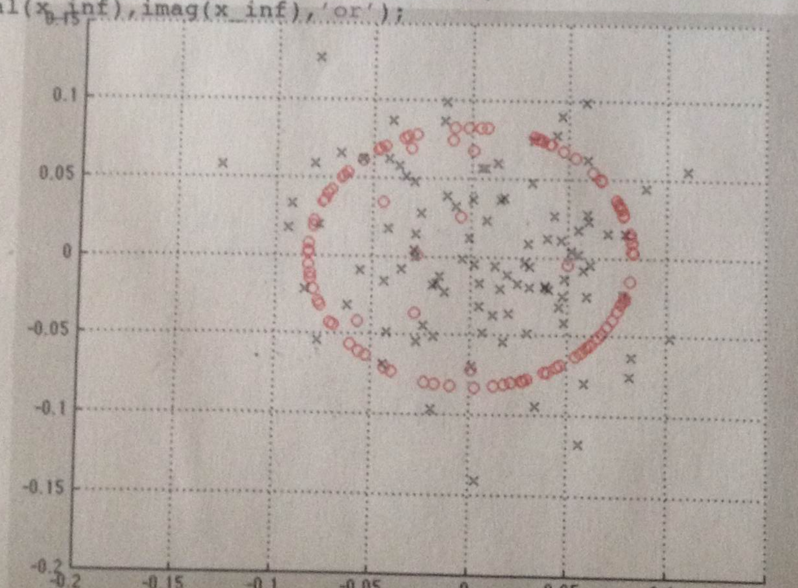
$(y = \sqrt{t})$

The SOCP problem is then:

$$\boxed{\begin{aligned} &\min.\ y \\ &\text{s.t. } \|B_i z\|_2 \le y \ \forall i=1,\dots,n \qquad \text{where } B_i = \begin{bmatrix} -e_i^T- & -0- \\ -0- & -e_i^T- \end{bmatrix} \\ &A_c z = b_c \end{aligned}}$$

```
% EE 364A Homework 5 Problem A3.9 %
close all; clear all;
% (c) %
m = 30; n = 100;

% complex least L2 norm %
A = randn(m,n) + i*randn(m,n);  % setup
b = randn(m,1) + i*randn(m,1);

% solution %
Ar = real(A); Ai = imag(A);
br = real(b); bi = imag(b);

bc = [br; bi];
Ac = [Ar -Ai; Ai Ar];

cvx_begin
    variable z(2*n)
    minimize (z'*z)            % Square of 2-norm
    subject to
        Ac*z == bc;
cvx_end
x_2 = z(1:n) + i*z(n+1:2*n);

% complex least inf norm %
% Note: SOCP method had too many second-order cone constraints to
% easily enter them all.
cvx_begin
    variable x(n) complex
    minimize (norm(x,inf))
    subject to
        A*x == b;
cvx_end
x_inf = x;      % This line is only to have clearly named variables

axis square;
scatter(real(x_2),imag(x_2),'xk');  hold on;
scatter(real(x_inf),imag(x_inf),'or');
grid on;
```

← Note: axis equal looks better.



x = X-2
o = x-inf

The ℓ-2 optimum solution is widely scattered while the least ∞-norm has most points very close to the circle of radius $r = \max_i |x_i| = \|x\|_\infty$. This makes sense since moving a single point closer (decreasing its magnitude) does not decrease $\|x\|_\infty$ at all except at $i = \arg\max_i |x_i|$ and even then it only decreases $\|x\|_\infty$ a small amount.

Problem A5.21

We proved in P6.9 that $\max_{i=1..k} |f(t_i) - y_i|$ is quasiconvex for $1 + b_1 t_i + b_2 t_i^2 > 0 \ \forall i = 1, \dots, k$, which is a domain on f for this problem. Because the function we are minimizing is quasiconvex, we can use bisection to the find the global minimum of:

$$\begin{array}{ll} \text{min.} & \max_{i=1..k} |f(t_i) - y_i| \\ \text{s.t} & (1 + b_1 t_i + b_2 t_i > 0) \end{array} \longrightarrow \begin{array}{ll} \text{min.} & x \\ \text{s.t} & |f(t_i) - y_i| \leq x \end{array} \quad T = \begin{bmatrix} - & 1 & - \\ - & t & - \\ - & t^2 & - \end{bmatrix}$$

$$\Longrightarrow \begin{array}{ll} \text{min.} & x \\ \text{s.t} & |a^T T_i + y_i (b^T T_i)| \leq x (b^T T_i) \ \forall i \end{array} \qquad a = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix}, \ b = \begin{bmatrix} 1 \\ b_1 \\ b_2 \end{bmatrix}$$

To solve this by bisection, we will set a lower band $l_b = e^{t_1}$ and upper bound $U_b = e^{t_k}$ and repeating the following algorithm until convergence (with tolerance .001):

- Set mid $= \frac{1}{2}(l_b + u_b)$
- check if mid is a feasible value for x
- if so, set U_b = mid
- otherwise set l_b = mid
- repeat until $u_b - l_b = .001$

See code Problem A5.2 for solution.

optimal parameters:

$a_0 = 1.0096$ $b_1 = -0.4144$

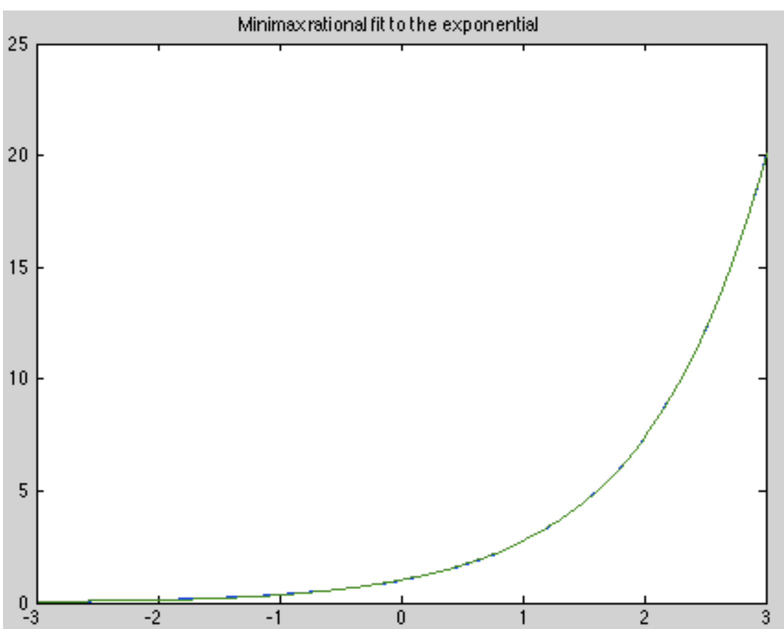
$a_1 = 0.6120$ $b_2 = 0.0485$

$a_2 = 0.1139$

so constraint is satisfied

Note: $1 - .4144x + .0485x^2 > 0 \ \forall x \in \mathbb{R}$

```
% EE 364A Homework 5 Problem A5.2 %
close all; clear all;

k= 201;
i = 1:k;
ti = -3 + 6*i/k;
yi = exp(ti);

x = [ones(1,k); ti; ti.^2];
ub = ti(end);     lb = ti(1);
% Bisection Algorithm loop %
astar = []; bstar = []; pstar = 0;
while ub-lb >= .001,
    p = .5*(ub+lb);
    cvx_begin quiet
        variable a(3)
        variable b(2)
        subject to
            abs(x'*a - yi'.*(x'*[1;b])) <= p*x'*[1;b];
    cvx_end

    if strcmp(cvx_status,'Solved'),
        ub = p;
        astar = a;
        bstar = b;
        pstar = p;
    else,
        lb = p;
    end
end

exp_approx = (x'*astar)./(x'*[1;bstar]);
exp_approx = exp_approx';

plot(ti,yi,ti,exp_approx);
```



Problem 5.15

(a) Let $z_i = x_i - y_i$, $z = x - y$

$\rightarrow d(z_i) = (z_i^T P z)^{1/2}$

Since $P \in S_+^n \rightarrow \vec{P} \in \mathbb{R}^{\frac{1}{2}n(n+1)}$ ← let $s = \frac{1}{2}n(n+1)$ contains all the information in P.

Let $P = \begin{bmatrix} P_1 & P_2 & P_4 & \\ P_2 & P_3 & P_5 & \\ P_4 & P_5 & P_6 & \\ & & & \ddots \\ & & & P_s \end{bmatrix} \longrightarrow \vec{P} = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ \vdots \\ P_s \end{bmatrix}$

$\rightarrow d(z_i)^2 = z_i^T P z_i = P_1 z_{i1}^2 + 2P_2 z_{i1} z_{i2} + P_3 z_{i2}^2 + \ldots \overset{?}{=} a_i^T \vec{P}$

for $a_i^T = [z_{i1}^2 \;\; 2z_{i1}z_{i2} \;\; z_{i2}^2 \;\; \ldots \;\; z_s^2]$

Let $A = \begin{bmatrix} - & a_1^T & - \\ - & a_2^T & - \\ & \vdots & \\ - & a_N^T & - \end{bmatrix}$ Then $\frac{1}{N}\sum_{i=1}^{N}(d_i - d_i(z_i))^2$

$= \frac{1}{N} \| d - \sqrt{(A\vec{P})} \|_2^2$ where the square root is element-wise.

Since CVX will not accept this square root minimizing the differences of square distances will suffice since d_i and $d(z_i)$ are both positive. Therefore the problem becomes:

minimize $\| d^2 - A\vec{P} \|_2$

s.t. $P \succeq 0$

where d^2 is element-wise square

```
% EE 364A Homework 5 Problem A5.15 %
close all; clear all;

quad_metric_data;
s = .5*n*(n+1);      % dimension of P

Z = X - Y;  A = [];
for i = 1:N,
    z = Z(:,i);
    row = [];
    for j = 1:n,
        for k = 1:j,
            entry = z(j)*z(k);
            if i~=j, entry = 2*entry;    end
            row = [row entry];
        end
    end
    A = [A; row];
end
B = A.^.5;
cvx_begin
    variable p(s)
    variable q
    minimize q
    subject to
        for i = 1:N,
            norm(d'- B*p) <= q;
        end
        [p(1) p(2) p(4) p(7) p(11); p(2) p(3) p(5) p(8) p(12); ...
            p(4) p(5) p(6) p(9) p(13); p(7) p(8) p(9) p(10) p(14);...
            p(11) p(12) p(13) p(14) p(15)] == semidefinite(n);
cvx_end


P = [p(1) p(2) p(4) p(7) p(11); p(2) p(3) p(5) p(8) p(12); ...
            p(4) p(5) p(6) p(9) p(13); p(7) p(8) p(9) p(10) p(14);...
            p(11) p(12) p(13) p(14) p(15)];

Ztest = X_test-Y_test;
MSE = 0;
for i = 1:N_test,
    z = Ztest(:,i);
    dist = (d_test(i) - (z'*P*z)^.5);
    MSE = MSE + dist;
end
MSE = MSE/N_test;
```

Test Mean Square Error = 5.02

Problem A6.61

(a) We know $y(t) = \sum_{\tau=1}^{k} h(\tau)x(t-\tau) + v(t) \quad \forall t = 2, \dots, N+1$

Therefore $y(t) - \sum_{\tau=1}^{k} h(\tau)x(t-\tau) = v(t)$ To get a maximum likelihood estimate, we then want to find $x(t)$ to minimize $\|y(t) - h(t) * x(t)\|_2$

Since ~~v(t) is Gaussian.~~ Since $v(t)$ is Gaussian, the least norm solution is the best estimator. We then estimate $\hat{x}_{ml}$ by solving the least ℓ-2 norm problem with constraints $x_i \le x_j \quad \forall \{(i,j), i \le j\}$:

$$\begin{aligned} &\text{Minimize} \quad \|y(t) - (h * x)(t)\|_2 \\ &\text{Subject to} \quad x_i \le x_j \quad \forall \{(i,j) \mid i \le j\}. \end{aligned}$$

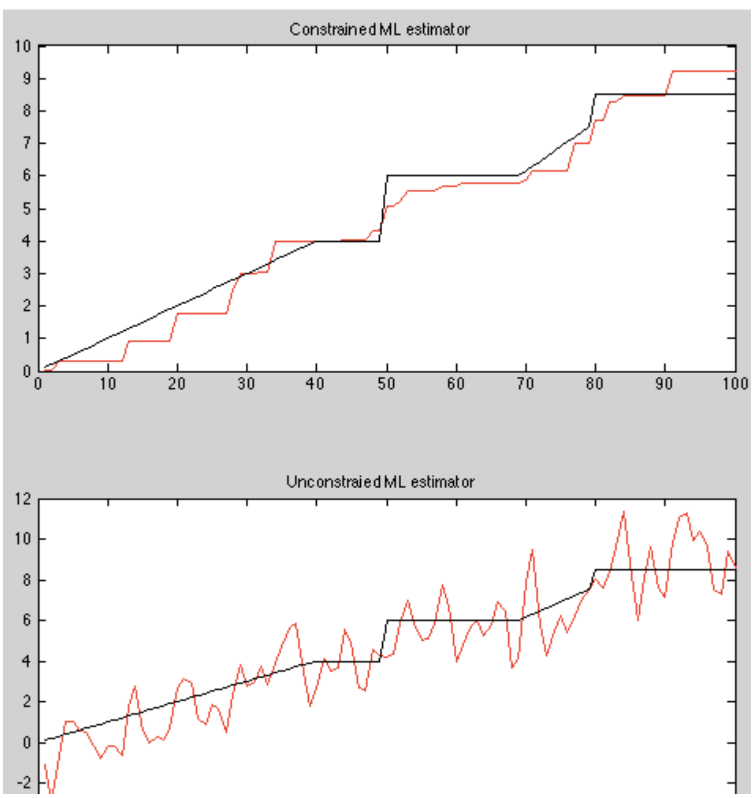
$$\Updownarrow$$

$$\begin{aligned} &\text{minimize} \quad s \\ &\text{s.t} \quad \|y(t) - (h * x)(t)\|_2 \le s \\ &\qquad x_i \le x_j \quad \forall \{(i,j) \mid i \le j\}. \end{aligned}$$

(b) Note: The problem data was missing the Gaussian noise so I added some in.
The ML estimator is found as described above. The ML, free (unconstrained ML estimator) is found by minimizing
$\|y(t) - (h * x)(t)\|_2$ without any other constraints on x.

See code for P A6.6

```
% EE 364A Homework 5 Problem A6.6 %
close all; clear all;

ml_estim_incr_signal_data;


y = yhat + randn(103,1);
cvx_begin
    variable xml(N)
    variable t
    minimize t
    subject to
        norm(y' - conv(xml,h)) <= t;
        xml >= 0;       % nonnegativity
        for s = 1:N-1,
            xml(s) <= xml(s+1);    % nondecreasing monotically
        end
cvx_end
subplot(211)
plot(1:100,xml,'r',1:100,xtrue,'k');
title('Constrained ML estimator');


% Free ML estimation %
cvx_begin
    variable xmlf(N)
    variable t
    minimize t
    subject to
        norm(y'-conv(xmlf,h)) <= t;
cvx_end
subplot(212);
plot(1:100,xmlf,'r',1:100,xtrue,'k');
title('Unconstraied ML estimator');
```



Problem A6.101

For a matrix P that represents the joint pdf of R_1 and R_2,
We have the following conditions:

Marginal pdf of R_1: $[-\mathbf{1}^T-]\,P = p_1^T$, where p_1 is the marginal pdf of R_1
$\sim N(\mu_1, \sigma_1^2)$

Marginal pdf of R_2: $P\begin{bmatrix}1\\ \mathbf{1}\\ 1\end{bmatrix} = p_2$ where p_2 is the marginal pdf of R_2.
$\sim N(\mu_2, \sigma_2^2)$

Correlation of R_1, R_2: $E[XY] - \mu_1\mu_2 = \rho\sigma_1\sigma_2$
ρ

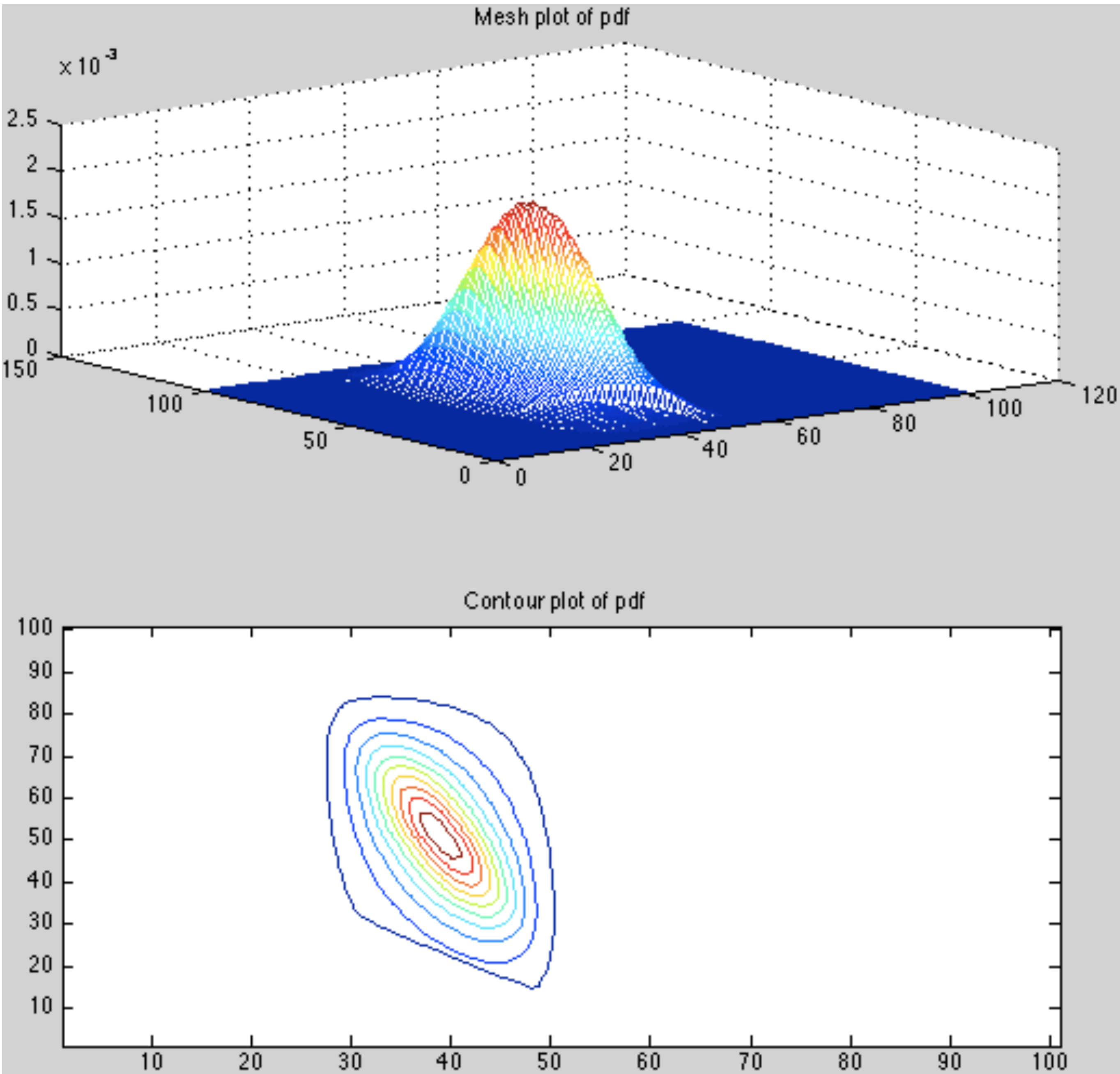
We are also trying to maximize $p_1^{loss} = P\{R_1 + R_2 \leq 0\}$.
Therefore we can formulate the problem

$$\begin{aligned}
&\text{maximize} \quad p^{loss}_{max} \\
&\text{s.t.} \quad P\{R_1 + R_2 \leq 0\} >= p^{loss}_{max} \\
&\qquad \mathbf{1}^T P = p_1^T \\
&\qquad P\mathbf{1} = p_2 \\
&\qquad E[XY] = \rho\sigma_1\sigma_2 + \mu_1\mu_2 .
\end{aligned}$$

See code. For $\mu_1 = 8$, $\mu_2 = 20$, $\sigma_1 = 6$, $\sigma_2 = 17.5$, $\rho = -.25$,

$p^{loss}_{max} = .0655$

```matlab
% EE 364A Homework 5 Problem A6.10 %
close all; clear all;

r = -30:70;
n = length(r);
mu1 = 8; mu2 = 20; sigma1 = 6; sigma2 = 17.5;
rho = -.25;
% Setting up denominator %
d1 = 0;     d2 = 0;
for j = 1:n,
    d1 = d1 + exp(-(r(j)-mu1)^2/(2*sigma1^2));
    d2 = d2 + exp(-(r(j)-mu2)^2/(2*sigma2^2));
end
% Setting up marginal distributions %
p1 = [];    p2 = [];
for i = 1:n,
    prob1 = exp(-(r(i)-mu1)^2/(2*sigma1^2))/d1;
    prob2 = exp(-(r(i)-mu2)^2/(2*sigma2^2))/d2;

    p1 = [p1; prob1];
    p2 = [p2; prob2];
end
X = r'*ones(1,n);
Y = ones(n,1)*r;
Z = zeros(n,n);
Z(X+Y <= 0) = 1;

XY = X.*Y;
cvx_begin
    variable P(n,n) nonnegative
    variable t
    maximize t
    subject to
        sum(P.*Z) >= t;     % Maximize objective
        P*ones(n,1) == p2;      % R2 marginal
        ones(1,n)*P == p1';     % R1 marginal
        XY(:)'*P(:) == rho*sigma1*sigma2 + mu1*mu2;
        % Correlation condition.
cvx_end

Ploss = sum(sum(P.*Z))/sum(sum(P));

subplot(211); mesh(P); title('Mesh plot of pdf');
subplot(212); contour(P); title('Contour plot of pdf');
```



Mesh plot of pdf
x 10^-3
2.5
2
1.5
1
0.5
0
150
100
50
0
0
20
40
60
80
100
120
Contour plot of pdf
100
90
80
70
60
50
40
30
20
10
10
20
30
40
50
60
70
80
90
100

Problem A13.9

(a) x is constraind to $0 \le x_i \le q_i$ and the profit made by the house in case j is : $prof_j = x^T p - x^T S e_j = x^T(p - Se_j)$

where $S = \begin{bmatrix} - 1_{S_1} - \\ - 1_{S_2} - \\ \vdots \\ - 1_{S_n} - \end{bmatrix}$ where I_{S_i} is the vector with element $k = 1$ for $k \in S_i$.

Therefore, the problem becomes:

$$\begin{array}{ll} \text{maximize} & \min_j prof_j \\ \text{s.t} & 0 \le x \le q \end{array} \Rightarrow \begin{array}{ll} \max. & \min_j \left(x^T(p - Se_j)\right) \\ \text{s.t} & 0 \le x \le q \end{array}$$

$$\Rightarrow \begin{array}{ll} \max. & t \\ \text{s.t} & x^T(p - Se_j) \ge t \quad \forall j = 1, ..., m \quad \leftarrow = (p - Se_j)^T x \quad \forall j = 1, ..., m \\ & Ix \ge 0 \\ & Ix \le q \quad \rightarrow \begin{bmatrix} -I \\ I \end{bmatrix} x \le \begin{bmatrix} 0 \\ q \end{bmatrix} \end{array}$$

(b) Expected profit $E[prof] = x^T(p - S\pi)$.

This is minimized given fixed x by:

$$\begin{array}{ll} \text{minimize} & x^T(p - S\pi) \\ \text{s.t} & 1^T \pi = 1 \\ & \pi \ge 0 \end{array}$$

This is the equivalent to the dual probleem of the problem from part (a). Since (a) is an LP and meets KKT conditions, the optimal objective and optimal dual objective will be equal.

```
% EE 364A Homework 5 Problem A13.9 %
close all; clear all;

p = [.5 .6 .6 .6 .2]';
q = [10 5 5 20 10]';
S = [1 1 0 0 0; 0 0 0 1 0; 1 0 0 1 1; 0 1 0 0 1; 0 0 1 0 0];

cvx_begin
    variable x(5) %integer
    variable t
    maximize t
    subject to
        x >= 0;
        x <= q;
        for j = 1:5,
            x'*(p- S(:,j)) >= t;
        end
cvx_end

% caculating worst case house profit for x = q %
wchp = 1e10; % Initializing house profit to a high value
for j = 1:5,
    profit_j = q'*(p-S(:,j));
    wchp = min(wchp, profit_j);
end

% Calculating imputed probabilities %
cvx_begin
    variable ppi(5)
    maximize (x'*(p-S*ppi));
    subject to
        eye(5)*ppi >=0;
        ones(1,5)*ppi == 1;
cvx_end
```

x_opt = [5 ; 5 ; 5 ; 5 ; 10]
max worst case profit = 3.5
Imputed probability pi for x_opt:

Pi = [.2007 ; .1993 ; .2004 ; .1999 ; .1997]