ECE 602 - Introduction to Optimization

Solutions to Home Assignment 4

Exercise 1

What is the solution of the norm approximation problem with one scalar variable $x \in \mathbf{R}$

$$\text{minimize } \|\mathbf{1}x - b\|$$

for the ℓ_1-, ℓ_2-, and ℓ_∞-norms? Explain your answers.

Solution:

(a) ℓ_2-norm: the average $\mathbf{1}^T b/n$.

(b) ℓ_1-norm: the (or a) median of the coefficient of b.

(c) ℓ_∞-norm: the midrange point $(\max b_i + \min b_i)/2$.

For the l_1–norm:

$$\min \|\mathbf{1}x - b\|_1$$

$$\min \sum_{i=1}^{m} \|x^* - b\|$$

Since the problem is not constraint, the optimal condition is contingent on

$$\frac{d}{dx} \sum_{i=1}^{m} \|x^* - b\| = 0$$

The l_1 -norm is not differentiable around the center of coordinates b_i, and it is differentiable elsewhere yielding values that are either $\{-1, +1\}$. Therefore, to stay consistent, we can express as:

$$\begin{aligned}\sum_{i=1}^{m} sign(x^* - b_i) &= 0, \text{when} x_* \neq b_i; \\ 0, \text{when} x_* &= b_i;\end{aligned}$$

Let's assume that $b = \{1, 2, 9\}$, which is odd. The optimal answer would be$x* = 2$, since it corresonpds to $-1 + 0 + 1 = 0$. If$b = \{1, 2, 3, 9\}$, which is even. The optimal answer would be$x* = (2, 3)$, since it corresonpds to $-1 - 1 + 1 + 1 = 0$. Therefore, the optimal answer would be $x* = median(b)$.

For the l_2–norm:

$$\min \|\mathbf{1}x - b\|_2$$

is equivalent to minimize

$$\min \|\mathbf{1}x - b\|_2^2$$

Ther are no constraints, thus, the minimization is simpley by differentiating and equation to 0, ie,

$$\begin{aligned}\frac{d}{dx}((\mathbf{1}x - b)^T(\mathbf{1}x - b)) &= \frac{d}{dx}(x^T \mathbf{1}^T \mathbf{1}x - 2b(\mathbf{1}x) + b^T b) = 0 \\ 2\mathbf{1}^T \mathbf{1}x &= 2b^T \mathbf{1} \\ x &= \frac{\sum_{i=1}^{m} b_i}{m}\end{aligned}$$

For the l_∞–norm: This should be the -norm between the max and min elements. This is a natural choice when considering this -norm wishes to shrink the largest value. If we select the optimal to be the min value the largest error will be its distance to the largest value and vice versa. When moving closer to the max element but before the half way make the largest error must be the distance to the max. After the half way point the distance from the min is now larger than that of the max point. At the middle thet are equal rendering a min.

$$\begin{aligned}&\min \|\mathbf{1}x - b\|_\infty \\ &\max |x - b_{min}|, |x - b_{max}| \\ x^* - b_{min} &= -(x^* - b_{max}) \\ 2x^* &= (b_{min} + b_{max}) \\ x^* &= \frac{(b_{min} + b_{max})}{2}\end{aligned}$$

Exercise 2

An *interval matrix* in $\mathbf{R}^{m\times n}$ is a matrix whose entries are intervals:

$$\mathcal{A} = \{A \in \mathbf{R}^{m\times n} \mid |A_{ij} - \bar{A}_{ij}| \leq R_{ij},\ i = 1, \ldots, m,\ j = 1, \ldots, n\}$$

The matrix $\bar{A} \in \mathbf{R}^{m\times n}$ is called the *nominal value* or *centre value*, and $R \in \mathbf{R}^{m\times n}$, which is element-wise nonnegative, is called the *radius*.

The *robust least-squares problem*, with interval matrix, is

$$\min_x \sup_{A\in\mathcal{A}} \|Ax - b\|_2,$$

The problem data are $\mathcal{A}$ (i.e., $\bar{A}$ and R) and $b \in \mathbf{R}^m$. The objective, as a function of x, is called *the worst-case residual norm*. The robust least-squares problem is evidently a convex optimization problem.

Formulate the interval matrix robust least-squares problem as a standard optimization problem, e.g., a QP, SOCP, or SDP. You can introduce new variables if needed. Your reformulation should have a number of variables and constraints that grows linearly with m and n, and not exponentially.

Solution:

---

a. The problme looks like a QP when reformulated.

$$\min \sup_{A\in\boldsymbol{A}} \|Ax - b\|_2$$

$$s.t. \boldsymbol{A} = \{A \in \boldsymbol{R}^{mxn} || A_{ij} - \bar{A}_{ij}|, i \in \boldsymbol{N}^m, j \in \boldsymbol{N}^n\}$$

This can be rewritten as,

$$\min_x \sup_{A\in\boldsymbol{A}} (Ax - b)^T (Ax - b)$$

Also,

$$\sup_{A\in\boldsymbol{A}} (Ax - b)_i = \sum_{j=1}^{n} (\bar{A}_{ij} x_j + R_{ij}|x_j|) - b_j$$

$$\inf_{A\in \boldsymbol{A}} (Ax-b)_i \;\; = \;\; \sum_{j=1}^{n}(\bar{A}_{ij}x_j - R_{ij}|x_j|) - b_j$$

Thus, the QP can be written as

$$\begin{aligned} &\min_{x,t} t^T t \\ s.t.\bar{A}x + Ry - b &\leq t \\ \bar{A}x + Ry - b &\geq -t \\ -y \leq x &\leq y \end{aligned}$$

Now since we are taking $\|x\|$, we can bound it between values $-y \leq x \leq y$. As a result, the QP can be written as,

$$\begin{aligned} &\min_{x,t} t^T t \\ s.t.\bar{A}x + Ry - b &\leq t \\ \bar{A}x + Ry - b &\geq -t \\ -y \leq x &\leq y \end{aligned}$$

Exercise 3

Using MATLAB, generate the following test signal of length 256:

```
m=256;
t=linspace(0,1,m)';
y=exp(-128*((t-0.3).^2))-3*(abs(t-0.7).^0.4);
```

Also, generate the following matrix of size 256×512:

```
mpdict=wmpdictionary(m,'LstCpt',{'dct',{'wpsym4',2}});
A=full(mpdict);
```

The columns of A consist of two orthogonal bases commonly used in signal analysis, viz. a discrete cosine transform basis (the first 256 columns) and a wavelet basis (the last 256 columns). Our goal is to represent y in terms of the columns of A, i.e., to find x such that $Ax = y$. Needless to say, since the columns of A are linearly dependent, there is no unique way to achieve the above goal. To overcome this difficulty, we consider the following norm minimization problem

$$\min_x \ \|x\| \quad \text{subject to } Ax = y$$

which finds the "smallest" x among all possible solutions of $Ax = b$.

(a) Find solutions to the above problem for the case of ℓ_2- and ℓ_1-norm. In particular, in the case of ℓ_1-norm, cast the problem as an LP and then solve it using CVX.

(b) Using the above solutions (which we denote by x_2 and x_1, respectively), reconstruct their corresponding approximations of y. How close are they to the original signal y? (You may want to use $\|A*x_i - y\|_2^2 / \|y\|_2^2$, $i = 1, 2$, as a measure of relative error.)

(c) Modify x_1 and x_2 by keeping only 5% of their largest (in absolute value) entries, while setting the rest of their entries to zero. What is the accuracy of your reconstructions now in both cases? Plot these reconstructions overlapped over the original y and indicate the relative errors.

(d) Repeat the previous experiment, while keeping 3% and then 1% of the largest (in absolute value) entries of x_1 and x_2. What percentage of the entries of x_2 should you keep to (approximately) reach the same relative error as in the case of x_1 with 3% "compression rate".

Solution:

This question request we fit two orthogonal bases to a test signal y. Both A and y are provided. The problems form:

$$\min_x \|x\| \quad st.Ax = y$$

a. Considering the problem as the l_2 -norm, which has been squared for a tidy solution, so we have:

$$\begin{aligned} L(x,v) &= \|x\|_2^2 + \mu(Ax-y) \\ \nabla L(x,v) &= 2xA^T\mu \\ \text{KKT:}2x^* + A^T\mu^* &= 0\text{and}Ax^* = y \\ x = A^T(AA^T)^{-1}y\text{and}v^* &= -2(AA^T)^{-1}y \end{aligned}$$

Given rank $A = 256 < 512$, AA^T is invertable. Considering the l_1 -norm version of this problem, a dummy variable t is introduced and the problme can be written as an LP.

$$\min_{t,x} \mathbf{1}^T t$$
$$s.t. t \geq x \geq -t, Ax = b$$

b. Their relative error is very close to the original. They are reaching the machine epsilon so they are roughly equal. Then results found on th etest machine are as follows:

$$\begin{aligned} \frac{\|Ax_1 - y\|_2^2}{\|y\|_2^2} &= 1.1 * 10 * -15 \\ \frac{\|Ax_2 - y\|_2^2}{\|y\|_2^2} &= 8.6 * 10 * -18 \end{aligned}$$

c. The 5 % of max in absolute value threshold did not affect the l_1 -norm in a noticable amount but the l_2 -norm suffered since most of the "energy" is centered arouond the mean value.

d. The 3% and 1% of max in absolute value threshold show the strength of the l_2 -norm but the l_2 -norm suffers much more as seen in Figure 2 and 3. The 3% kept with the l_2 -norm gave relative error of $5.6 * 10^{-5}$ for the l_2 -norm to render roughly the same error it was found to require roughly a thershold of 18 %.

code:

```
m=256;
t=linespace(0,1,m)'; \\
y=/exp (-128*((t-0.3).^2))-3*(abs(t-0.7).^0.4);\\
mpdict=wmpdictionary(m, 'LstCpt',{'dct',{'wpsym4',2}});\\
A=full(mpdict);\\ \\
N=size (A,2)\\
one=ones(1,N);\\
cvx_begin\\
variable x(N)\\
minimize (norm(x,1))\\
subject to \\
A*x==y\\
cvx_end\\
x1=x;\\
clear x;\\
N=size(A,2);\\
one=ones(1,N);\\
cvx_begin\\
variable x(N)\\
minimize (norm(x,2)^2)\\
subject to \\
A*x==y\\
cvx_end\\
x2=x;\\

\%b
error1=norm(A*x1-y,2)^2/norm(y,2)^2;\\
error2=norm(A*x2-y,2)^2/norm(y,2)^2;\\
error1\\
error2\\

\%c
temp1=zeros(length(x),1);\\
for i=1:26\\
[val,pos]=max(abs(x1));\\
temp(pos)=x1(pos);\\
x1(pos)=0;\\
end\\
temp2=zeros(length(x),1);\\
for i=1:26\\
[val,pos]=max(abs(x2));\\
temp2(pos)=x2(pos);\\
x2(pos)=0;\\
```

```
end\\
error1_5=norm(A*temp1-y,2)^2/norm(y,2)^2;\\
error2_5=norm(A*temp2-y,2)^2/norm(y,2)^2;\\
error1_5\\
error2_5\\
figure(1)\\
hold all;\\
plot(A*temp1)\\
plot(A*temp2)\\
plot(y)\\
ylable('A*x')\\
legend(['l-1 err=', num2str(error1_5)],['l-2 err=', num2str(error2_5)],'y')\\

\%d
x1=x1+temp1;\\
end\\
temp1=zeros(length(x),1);\\
for i=1:ceil(512*.03)\\
[val,pos]=max(abs(x1));\\
temp1(pos)=x1(pos);\\
x1(pos)=0;\\
end\\
x2=x2+temp2;\\
temp2=zeros(length(x),1);\\
for i=1:ceil(512*.03)\\
[val,pos]=max(abs(x2));\\
temp2(pos)=x2(pos);\\
x2(pos)=0;\\
end\\
error1_3=norm(A*temp1-y,2)^2/norm(y,2)^2;\\
error2_3=norm(A*temp2-y,2)^2/norm(y,2)^2;\\
figure(2)\\
hold all;\\
plot(A*temp1)\\
plot(A*temp2)\\
plot(y)\\
ylable('A*x')\\
legend(['l-1 err=', num2str(error1_3)],['l-2 err=', num2str(error2_3)],'y')\\
x2=x2+temp2;\\
temp2=zeros(length(x),1);\\
for i=1:ceil(512*.18)\\
[val,pos]=max(abs(x2));\\
temp2(pos)=x2(pos);\\
x2(pos)=0;\\
```

```
end\\

x1=x1+temp1;\\
temp1=zeros(length(x),1);\\
for i=1:ceil(512*.01)\\
[val,pos]=max(abs(x1));\\
temp1(pos)=x1(pos);\\
x1(pos)=0;\\
end\\

x2=x2+temp2;\\
temp2=zeros(length(x),1);\\
for i=1:ceil(512*.01)\\
[val,pos]=max(abs(x2));\\
temp2(pos)=x2(pos);\\
x2(pos)=0;\\
end\\


error1_1=norm(A*temp1-y,2)^2/norm(y,2)^2;\\
error2_1=norm(A*temp2-y,2)^2/norm(y,2)^2;\\

figure(3)\\
hold all;\\
plot(A*temp1)\\
plot(A*temp2)\\
plot(y)\\
ylable('A*x')\\
legend(['l-1 err=', num2str(error1_1)],['l-2 err=', num2str(error2_1)],'y')\\
end\\
```

Figure 1: Norm Compression Comparision 5%

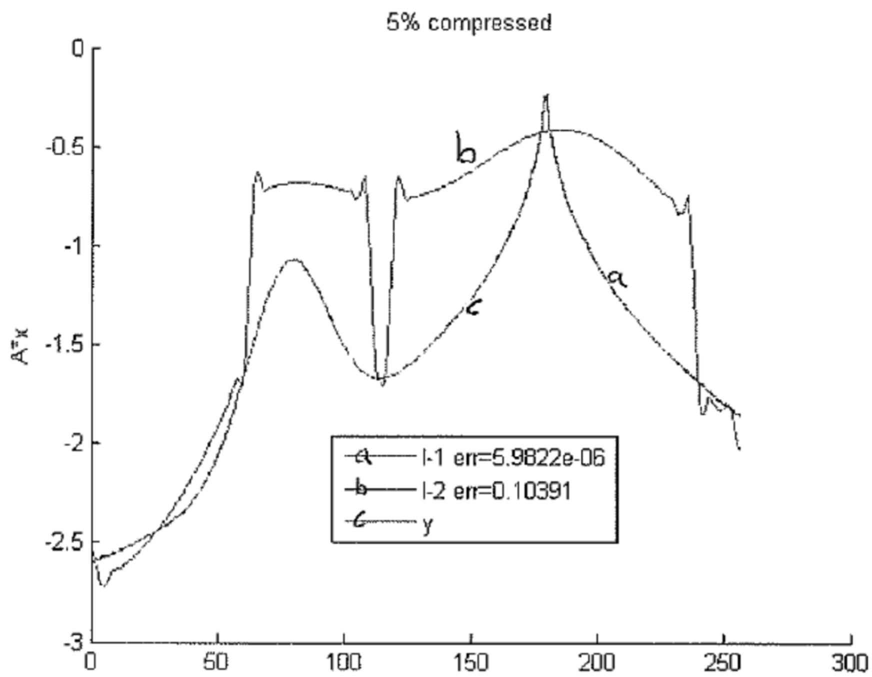

Figure 2: Norm Compression Comparision 3%

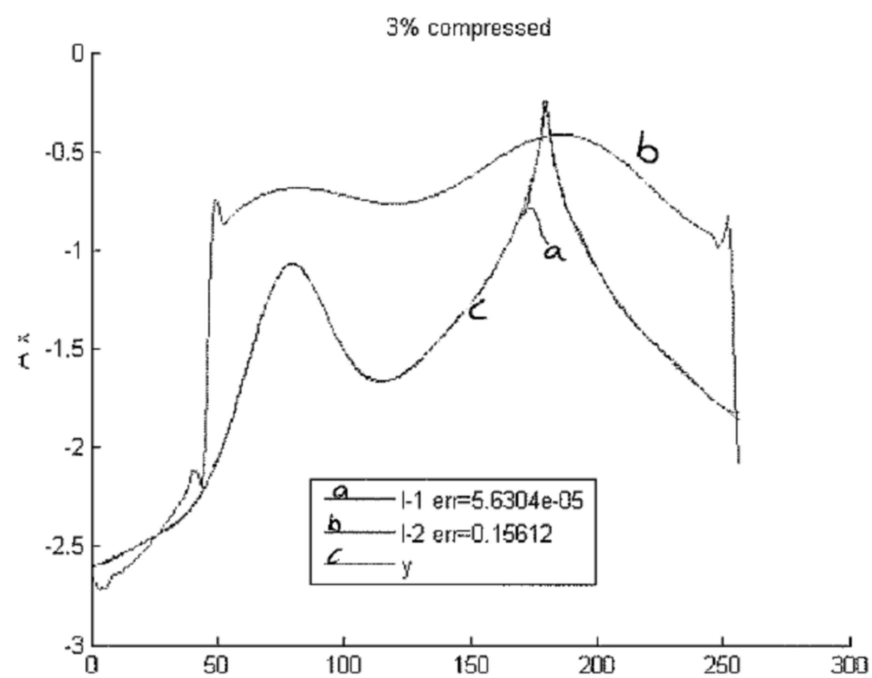

Figure 3: Norm Compression Comparision 1%