HW IV Solutions

EE 588: Optimization for the Information and Data Sciences

University of Southern California

6.2 ℓ_1,ℓ_2, *and* ℓ*-norm approximation by a constant vector.* What is the solution of the norm approximation problem with one scalar variable $x \in \mathbb{R}$,

$$\text{minimize}||x1 - b||,$$

for ℓ_1,ℓ_2, and $\ell - norms$?

Solution itemize

- ℓ_2-norm: the average $1^T b/m$.
- ℓ_1-norm: the (or a) median of the coefficients of b.
- ℓ_∞: the midrange point $(\text{max} b_i - \text{min} b_i)/2$

6.6 *Duals of some penalty function approximation problems.* Derive a Lagrange dual for the problem

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^m \phi(r_i) \\ \text{subject to} & r = Ax - b, \end{array}$$

for the following penalty functions $\phi : \mathbf{R} \to \mathbf{R}$. The variables are $x \in \mathbf{R}^n$, $r \in \mathbf{R}^m$.

(a) *Deadzone-linear penalty* (with deadzone width $a = 1$),

$$\phi(u) = \begin{cases} 0 & |u| \leq 1 \\ |u| - 1 & |u| > 1. \end{cases}$$

(b) *Huber penalty* (with $M = 1$),

$$\phi(u) = \begin{cases} u^2 & |u| \leq 1 \\ 2|u| - 1 & |u| > 1. \end{cases}$$

(c) *Log-barrier* (with limit $a = 1$),

$$\phi(u) = -\log(1 - x^2), \qquad \mathbf{dom}\, \phi = (-1, 1).$$

(d) *Relative deviation from one,*

$$\phi(u) = \max\{u, 1/u\} = \begin{cases} u & u \geq 1 \\ 1/u & u \leq 1, \end{cases}$$

with $\mathbf{dom}\, \phi = \mathbf{R}_{++}$.

Solution. We first derive a dual for general penalty function approximation. The Lagrangian is

$$L(x, r, \lambda) = \sum_{i=1}^{m} \phi(r_i) + \nu^T (Ax - b - r).$$

The minimum over x is bounded if and only if $A^T\nu = 0$, so we have

$$g(\nu) = \begin{cases} -b^T\nu + \sum_{i=1}^m \inf_{r_i}(\phi(r_i) - \nu_i r_i) & A^T\nu = 0 \\ -\infty & \text{otherwise.} \end{cases}$$

Using

$$\inf_{r_i}(\phi(r_i) - \nu_i r_i) = -\sup_{r_i}(\nu_i r_i - \phi(r_i)) = -\phi^*(\nu_i),$$

we can express the general dual as

$$\begin{array}{ll} \text{maximize} & -b^T\nu - \sum_{i=1}^m \phi^*(\nu_i) \\ \text{subject to} & A^T\nu = 0. \end{array}$$

Now we'll work out the conjugates of the given penalty functions.

(a) *Deadzone-linear penalty.* The conjugate of the deadzone-linear function is

$$\phi^*(z) = \begin{cases} |z| & |z| \le 1 \\ \infty & |z| > 1, \end{cases}$$

so the dual of the dead-zone linear penalty function approximation problem is

$$\begin{array}{ll} \text{maximize} & -b^T\nu - \|\nu\|_1 \\ \text{subject to} & A^T\nu = 0, \quad \|\nu\|_\infty \le 1. \end{array}$$

(b) *Huber penalty.*

$$\phi^*(z) = \begin{cases} z^2/4 & |z| \le 2 \\ \infty & \text{otherwise,} \end{cases}$$

so we get the dual problem

$$\begin{array}{ll} \text{maximize} & -(1/4)\|\nu\|_2^2 - b^T\nu \\ \text{subject to} & A^T\nu = 0 \\ & \|\nu\|_\infty \le 2. \end{array}$$

(c) *Log-barrier.* The conjugate of ϕ is

$$\begin{aligned} \phi^*(z) &= \sup_{|x|<1} \left(xz + \log(1 - x^2)\right) \\ &= -1 + \sqrt{1 + z^2} + \log(-1 + \sqrt{1 + z^2}) - 2\log|z| + \log 2. \end{aligned}$$

(d) *Relative deviation from one.* Here we have

$$\phi^*(z) = \sup_{x>0}(xz - \max\{x, 1/x\}) = \begin{cases} -2\sqrt{-z} & z \leq -1 \\ z-1 & -1 \leq z \leq 1 \\ -\infty & z > 1. \end{cases}$$

Plugging this in the dual problem gives

$$\begin{array}{ll} \text{maximize} & -b^T\nu + \sum_{i=1}^m s(\nu_i) \\ \text{subject to} & A^T\nu = 0, \quad \nu \preceq \mathbf{1}, \end{array}$$

where

$$s(\nu_i) = \begin{cases} 2\sqrt{-\nu_i} & \nu_i \leq -1 \\ 1-\nu_i & \nu_i \geq -1. \end{cases}$$

6.9 *Minimax rational function fitting.* Show that the following problem is quasiconvex:

$$\text{minimize} \quad \max_{i=1,\ldots,k} \left| \frac{p(t_i)}{q(t_i)} - y_i \right|$$

where

$$p(t) = a_0 + a_1 t + a_2 t^2 + \cdots + a_m t^m, \qquad q(t) = 1 + b_1 t + \cdots + b_n t^n,$$

and the domain of the objective function is defined as

$$D = \{(a,b) \in \mathbf{R}^{m+1} \times \mathbf{R}^n \mid q(t) > 0,\ \alpha \leq t \leq \beta\}.$$

In this problem we fit a rational function $p(t)/q(t)$ to given data, while constraining the denominator polynomial to be positive on the interval $[\alpha, \beta]$. The optimization variables are the numerator and denominator coefficients a_i, b_i. The interpolation points $t_i \in [\alpha, \beta]$, and desired function values y_i, $i = 1, \ldots, k$, are given.

Solution. Let's show the objective is quasiconvex. Its domain is convex. Since $q(t_i) > 0$ for $i = 1, \ldots, k$, we have

$$\max_{i=1,\ldots,k} |p(t_i)/q(t_i) - y_i| \leq \gamma$$

if and only if

$$-\gamma q(t_i) \leq p(t_i) - y_i q(t_i) \leq \gamma q(t_i), \quad i = 1, \ldots, k,$$

which is a pair of linear inequalities.

7.8 *Estimation using sign measurements.* We consider the measurement setup

$$y_i = \mathbf{sign}(a_i^T x + b_i + v_i), \quad i = 1, \ldots, m,$$

where $x \in \mathbf{R}^n$ is the vector to be estimated, and $y_i \in \{-1, 1\}$ are the measurements. The vectors $a_i \in \mathbf{R}^n$ and scalars $b_i \in \mathbf{R}$ are known, and v_i are IID noises with a log-concave probability density. (You can assume that $a_i^T x + b_i + v_i = 0$ does not occur.) Show that maximum likelihood estimation of x is a convex optimization problem.

Solution. We re-order the observations so that $y_i = 1$ for $i = 1, \ldots, k$ and $y_i = 0$ for $i = k+1, \ldots, m$. The probability of this event is

$$\begin{aligned}&\textstyle\prod_{i=1}^k \mathbf{prob}(a_i^T x + b_i + v_i > 0) \cdot \prod_{i=k+1}^m \mathbf{prob}(a_i^T x + b_i + v_i < 0)\\ &= \textstyle\prod_{i=1}^k F(-a_i^T x - b_i) \cdot \prod_{i=k+1}^m (1 - F(-a_i^T x - b_i)),\end{aligned}$$

where F is the cumulative distribution of the noise density. The integral of a log-concave function is log-concave, so F is log-concave, and so is $1 - F$. The log-likelihood function is

$$l(x) = \sum_{i=1}^k \log F(-a_i^T x - b_i) + \sum_{i=k+1}^m \log(1 - F(-a_i^T x - b_i)),$$

which is concave. Therefore, maximizing it is a convex problem.

- *Total variation image interpolation.* A grayscale image is represented as an $m \times n$ matrix of intensities U^{orig}. You are given the values U_{ij}^{orig}, for $(i, j) \in \mathcal{K}$, where $\mathcal{K} \subset \{1, \ldots, m\} \times \{1, \ldots, n\}$. Your job is to *interpolate* the image, by guessing the missing values. The reconstructed image will be represented by $U \in^{m \times n}$, where U satisfies the interpolation conditions $U_{ij} = U_{ij}^{\text{orig}}$ for $(i, j) \in \mathcal{K}$.

  The reconstruction is found by minimizing a roughness measure subject to the interpolation conditions. One common roughness measure is the ℓ_2 variation (squared),

  $$\sum_{i=2}^m \sum_{j=2}^n \left((U_{ij} - U_{i-1,j})^2 + (U_{ij} - U_{i,j-1})^2\right).$$

  Another method minimizes instead the *total variation*,

  $$\sum_{i=2}^m \sum_{j=2}^n \left(|U_{ij} - U_{i-1,j}| + |U_{ij} - U_{i,j-1}|\right).$$

  Evidently both methods lead to convex optimization problems.

  Carry out ℓ_2 and total variation interpolation on the problem instance with data given in `tv_img_interp.m`. This will define `m`, `n`, and matrices `Uorig` and `Known`. The matrix `Known` is $m \times n$, with (i, j) entry one if $(i, j) \in \mathcal{K}$, and zero otherwise. The mfile also has skeleton plotting code. (We give you the entire original image so you can compare your reconstruction to the original; obviously your solution cannot access U_{ij}^{orig} for $(i, j) \notin \mathcal{K}$.)

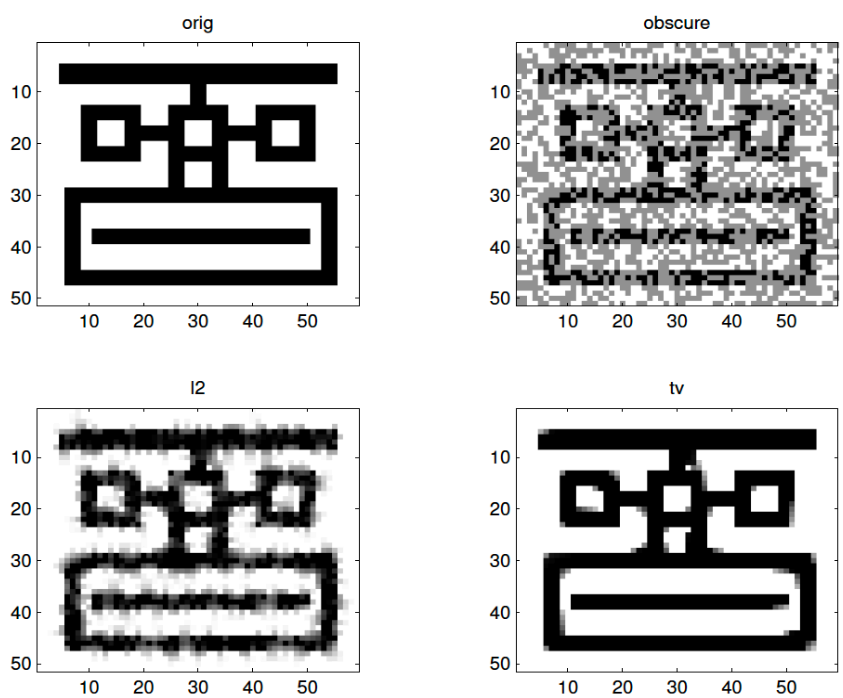
Solution. The code for the interpolation is very simple. For ℓ_2 interpolation, the code is the following.

```
cvx_begin
    variable Ul2(m, n);
    Ul2(Known) == Uorig(Known); % Fix known pixel values.
    Ux = Ul2(1:end,2:end) - Ul2(1:end,1:end-1); % x (horiz) differences
    Uy = Ul2(2:end,1:end) - Ul2(1:end-1,1:end); % y (vert) differences
    minimize(norm([Ux(:); Uy(:)], 2)); % l2 roughness measure
cvx_end
```

For total variation interpolation, we use the following code.

```
cvx_begin
    variable Utv(m, n);
    Utv(Known) == Uorig(Known); % Fix known pixel values.
    Ux = Utv(1:end,2:end) - Utv(1:end,1:end-1); % x (horiz) differences
    Uy = Utv(2:end,1:end) - Utv(1:end-1,1:end); % y (vert) differences
    minimize(norm([Ux(:); Uy(:)], 1)); % tv roughness measure
cvx_end
```

We get the following images



Piecewise-linear fitting. In many applications some function in the model is not given by a formula, but instead as tabulated data. The tabulated data could come from empirical measurements, historical data, numerically evaluating some complex expression or solving some problem, for a set of values of the argument. For use in a convex optimization model, we then have to fit these data with a convex function that is compatible with the solver or other system that we use. In this problem we explore a very simple problem of this general type.

Suppose we are given the data (x_i, y_i), $i = 1, \dots, m$, with $x_i,\ y_i \in \mathbf{R}$. We will assume that x_i are sorted, *i.e.*, $x_1 < x_2 < \cdots < x_m$. Let $a_0 < a_1 < a_2 < \cdots < a_K$ be a set of fixed knot points, with $a_0 \leq x_1$ and $a_K \geq x_m$. Explain how to find the convex piecewise linear function f, defined over $[a_0, a_K]$, with knot points a_i, that minimizes the least-squares fitting criterion

$$\sum_{i=1}^{m} (f(x_i) - y_i)^2.$$

You must explain what the variables are and how they parametrize f, and how you ensure convexity of f.

Hints. One method to solve this problem is based on the Lagrange basis, $f_0, \dots, f_K$, which are the piecewise linear functions that satisfy

$$f_j(a_i) = \delta_{ij}, \quad i, j = 0, \dots, K.$$

Another method is based on defining $f(x) = \alpha_i x + \beta_i$, for $x \in (a_{i-1}, a_i]$. You then have to add conditions on the parameters α_i and β_i to ensure that f is continuous and convex.

Apply your method to the data in the file `pwl_fit_data.m`, which contains data with $x_j \in [0, 1]$. Find the best affine fit (which corresponds to $a = (0, 1)$), and the best piecewise-linear convex function fit for 1, 2, and 3 internal knot points, evenly spaced in $[0, 1]$. (For example, for 3 internal knot points we have $a_0 = 0,\ a_1 = 0.25,\ a_2 = 0.50,\ a_3 = 0.75,\ a_4 = 1$.) Give the least-squares fitting cost for each one. Plot the data and the piecewise-linear fits found. Express each function in the form

$$f(x) = \max_{i=1\dots,K} (\alpha_i x + \beta_i).$$

(In this form the function is easily incorporated into an optimization problem.)

Solution. Following the hint, we will use the Lagrange basis functions $f_0, \dots, f_K$. These can be expressed as

$$f_0(x) = \left(\frac{a_1 - x}{a_1 - a_0} \right)_+,$$

$$f_i(x) = \left(\min \left(\frac{x - a_{i-1}}{a_i - a_{i-1}}, \frac{a_{i+1} - x}{a_i - a_{i+1}} \right) \right)_+, \quad i = 1, \dots, K-1,$$

and

$$f_K(x) = \left(\frac{x - a_{K-1}}{a_K - a_{K-1}} \right)_+.$$

The function f can be parametrized as

$$f(x) = \sum_{i=0}^{K} z_i f_i(x),$$

where $z_i = f(a_i)$, $i = 0, \ldots, K$. We will use $z = (z_0, \ldots, z_K)$ to parametrize f. The least-squares fitting criterion is then

$$J = \sum_{i=1}^{m} (f(x_i) - y_i)^2 = \|Fz - y\|_2^2,$$

where $F \in \mathbf{R}^{m \times (K+1)}$ is the matrix

$$F_{ij} = f_j(x_i), \quad i = 1, \ldots, m, \quad j = 0, \ldots, K.$$

(We index the columns of F from 0 to K here.)

We must add the constraint that f is convex. This is the same as the condition that the slopes of the segments are nondecreasing, *i.e.*,

$$\frac{z_{i+1} - z_i}{a_{i+1} - a_i} \geq \frac{z_i - z_{i-1}}{a_i - a_{i-1}}, \quad i = 1, \ldots, K-1.$$

This is a set of linear inequalities in z. Thus, the best PWL convex fit can be found by solving the QP

$$\begin{array}{ll} \text{minimize} & \|Fz - y\|_2^2 \\ \text{subject to} & \frac{z_{i+1}-z_i}{a_{i+1}-a_i} \geq \frac{z_i - z_{i-1}}{a_i - a_{i-1}}, \quad i = 1, \ldots, K-1. \end{array}$$

The following code solves this problem for the data in `pwl_fit_data`.

```
figure
plot(x,y,'k:','linewidth',2)
hold on

% Single line
p = [x ones(100,1)]\y;
alpha = p(1)
beta = p(2)
plot(x,alpha*x+beta,'b','linewidth',2)
mse = norm(alpha*x+beta-y)^2

for K = 2:4
    % Generate Lagrange basis
    a = (0:(1/K):1)';
    F = max((a(2)-x)/(a(2)-a(1)),0);
    for k = 2:K
        a_1 = a(k-1);
        a_2 = a(k);
        a_3 = a(k+1);
        f = max(0,min((x-a_1)/(a_2-a_1),(a_3-x)/(a_3-a_2)));
        F = [F f];
    end
    f = max(0,(x-a(K))/(a(K+1)-a(K)));
    F = [F f];
```

```
% Solve problem
cvx_begin
    variable z(K+1)
    minimize(norm(F*z-y))
    subject to
        (z(3:end)-z(2:end-1))./(a(3:end)-a(2:end-1)) >=...
            (z(2:end-1)-z(1:end-2))./(a(2:end-1)-a(1:end-2))
cvx_end

% Calculate alpha and beta
alpha = (z(2:end)-z(1:end-1))./(a(2:end)-a(1:end-1))
beta = z(2:end)-alpha(1:end).*a(2:end)

% Plot solution
y2 = F*z;
mse = norm(y2-y)^2
if K==2
    plot(x,y2,'r','linewidth',2)
elseif K==3
    plot(x,y2,'g','linewidth',2)
```

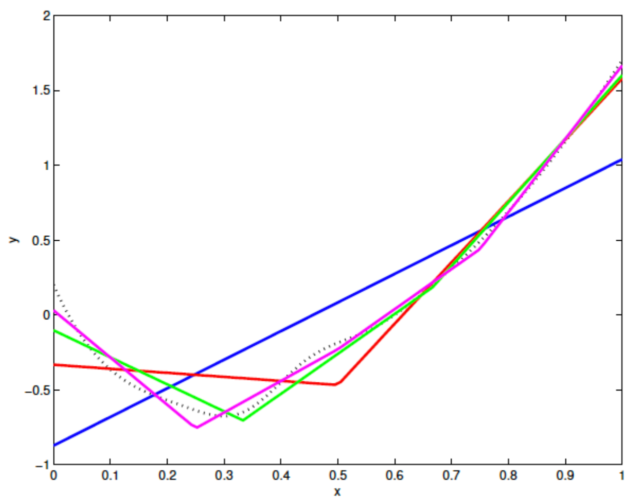


Figure 8: Piecewise-linear approximations for $K = 1, 2, 3, 4$

```
    else
        plot(x,y2,'m','linewidth',2)
    end

end
xlabel('x')
ylabel('y')
```

This generates figure 8. We can see that the approximation improves as K increases. The following table shows the result of this approximation.

K	$\alpha_1, \ldots, \alpha_K$	$\beta_1, \ldots, \beta_K$	J
1	1.91	-0.87	12.73
2	$-0.27, 4.09$	$-0.33, -2.51$	2.62
3	$-1.80, 2.67, 4.25$	$-0.10, -1.59, -2.65$	0.60
4	$-3.15, 2.11, 2.68, 4.90$	$0.03, -1.29, -1.57, -3.23$	0.22

There is another way to solve this problem. We are looking for a piecewise linear function. If we have at least one internal knot ($K \geq 2$), the function should satisfy the two following constraints:

- convexity: $\alpha_1 \leq \alpha_2 \leq \cdots \leq \alpha_K$
- continuity: $\alpha_i a_i + \beta_i = \alpha_{i+1} a_i + \beta_{i+1},\ i = 1, \ldots, K-1$.

Therefore, the opimization problem is

$$\begin{array}{ll} \text{minimize} & (\sum_{i=1}^m f(x_i) - y_i)^2 \\ \text{subject to} & \alpha_i \leq \alpha_{i+1}, \quad i = 1, \ldots, K-1 \\ & \alpha_i a_i + \beta_i = \alpha_{i+1} a_i + \beta_{i+1}, \quad i = 1, \ldots, K-1 \end{array}$$

Reformulating the problem by representing $f(x_i)$ in matrix form, we get

$$\begin{array}{ll} \text{minimize} & \| \mathbf{diag}(x) F\alpha + F\beta - y\|^2 \\ \text{subject to} & \alpha_i \leq \alpha_{i+1}, \quad i = 1, \ldots, K-1 \\ & \alpha_i a_i + \beta_i = \alpha_{i+1} a_i + \beta_{i+1}, \quad i = 1, \ldots, K-1 \end{array}$$

where the variables are $\alpha \in \mathbf{R}^K$ and $\beta \in \mathbf{R}^K$, and problem data are $x \in \mathbf{R}^m$, $y \in \mathbf{R}^m$ and

$$F_{ij} = \begin{cases} 1 & \text{if } a_{j-1} = x_i,\ j = 1 \\ 1 & \text{if } a_{j-1} < x_i \leq a_j \\ 0 & \text{otherwise .} \end{cases}$$

```
% another approach for PWL fitting problem
clear all;
pwl_fit_data;
m  = length(x);
xp = 0:0.001:1;  % for fine-grained pwl function plot
mp = length(xp);
yp = [];
```

```
for K = 1:4 % internal knot 1,2,3

    a = [0:1/K:1]'; % a_0,...,a_K
    % matrix for sum f(x_i)
    F = sparse(1:m,max(1,ceil(x*K)),1,m,K);

    % solve problem
    cvx_begin
    variables alpha(K) beta(K)
    minimize( norm(diag(x)*F*alpha+F*beta-y) )
    subject to
    if (K>=2)
        alpha(1:K-1).*a(2:K)+beta(1:K-1) == alpha(2:K).*a(2:K)+beta(2:K)
        a(1:K-1) <= a(2:K)
    end
    cvx_end

    fp = sparse(1:mp,max(1,ceil(xp*K)),1,mp,K);
    yp = [yp diag(xp)*fp*alpha+fp*beta];
end
plot(x,y,'b.',xp,yp);
```