Model Based Estimation Methods: Exercise 8

Jaeyong Jung (359804), Sangwook Lee (362209)

June 15, 2016

Things to discuss in this exercise

a) Use the discrepancy principle to obtain a good stopping index k^*. Plot the exact and estimated initial temperature ξ^* $\tilde{\xi}_k$ as well as the final temperature data

Stopping index k^* is 5.

Figure 1: Discrepancy Principle

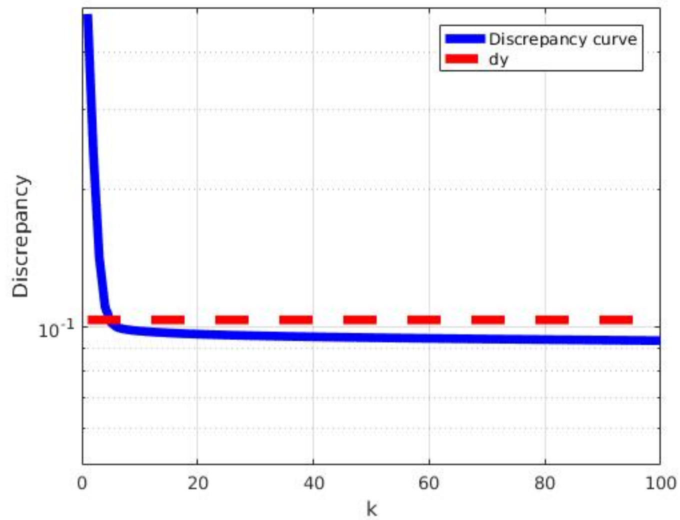
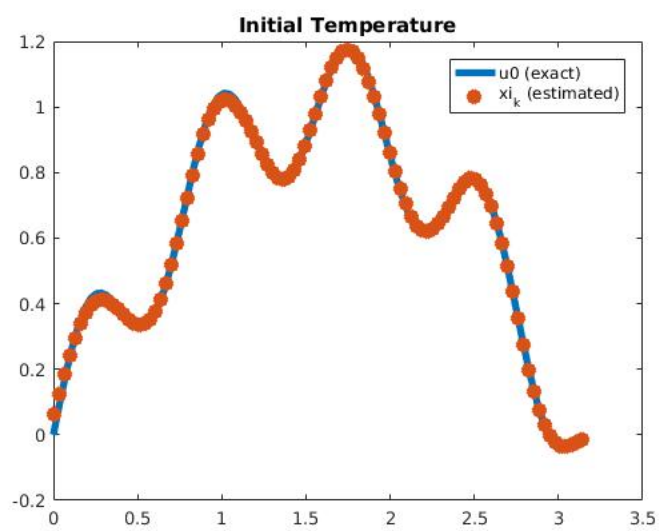


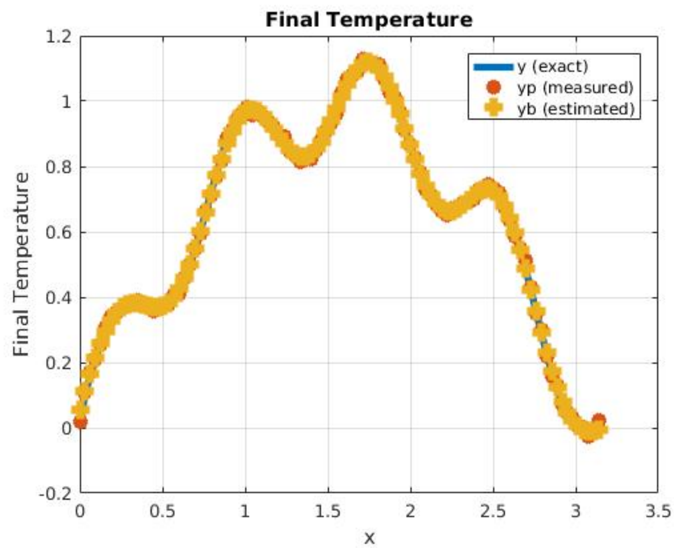
Figure 2: gg



I also made initial temperature graph. Estimated initial temperature is close to exact initial temperature. The norm of the error is 0.4246. (Norm of initial temperature is 7.1750) The error ratio is 5.9178%.

Final temperature graph is drawn below. 'yp' and 'yb' covered the exact final temperature. error of the estimated temperature is 0.0791. Considering the norm of exact temperature is 7.0771, the error is 1.1177%.

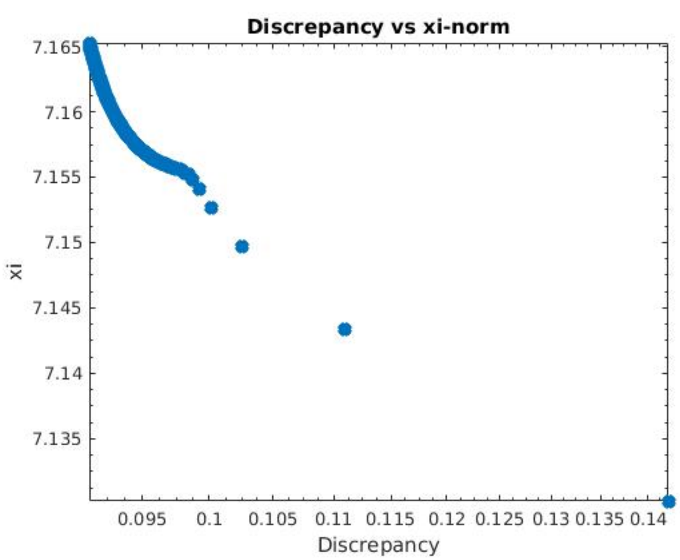
Figure 3: Final Temperature



b) Now we want to choose a stopping index k^* by means of the L-curve. To this end, plot the values $(||A\tilde{\xi}_k - \tilde{y}||_2, ||\tilde{\xi}_k||_2)$ for $k = 1, ..., 250$ in a log-log plot. What do you observe?

The L-curve described below is generated from $k = 3$. It is hard to see exact L-shaped curve. Thus, it is necessary to apply C-norm to get nice looking L-curve.

Figure 4: L-Curve

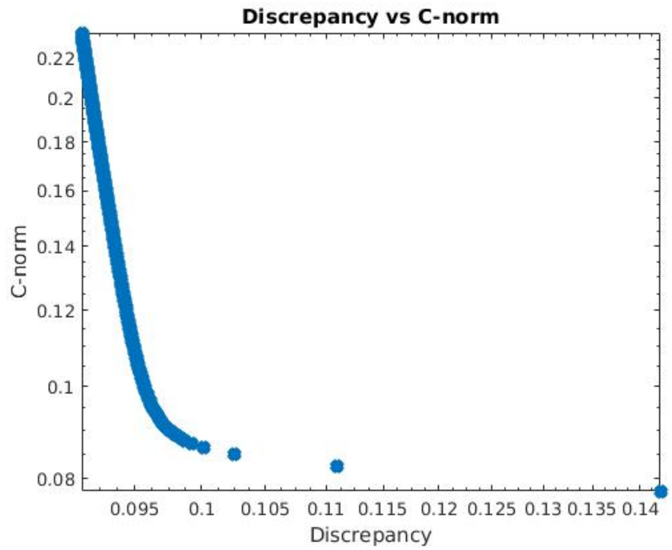


c) Plot another L-curve, replacing $||\tilde{\xi}_k||_2$ by $||C\tilde{\xi}_k||_2$. Note that the C-norm will

penalize oscillations in the estimate $\tilde{\xi}_k$. For the obtained index k^*, plot initial and final temperatures in respective plots as in a.), and discuss the results.

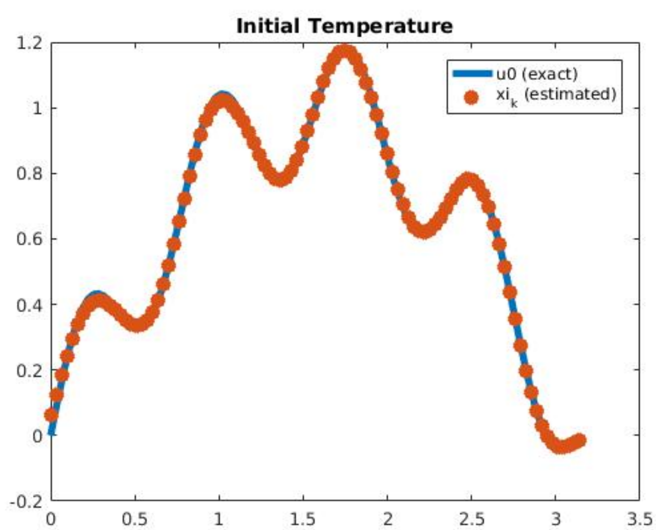
The L-Curve with C-Norm is also drawn from $k = 3$. I picked up k from a point which has biggest curvature. k^* is chosen to be 20. ($k^* = 20$)

Figure 5: L-Curve with C-Norm



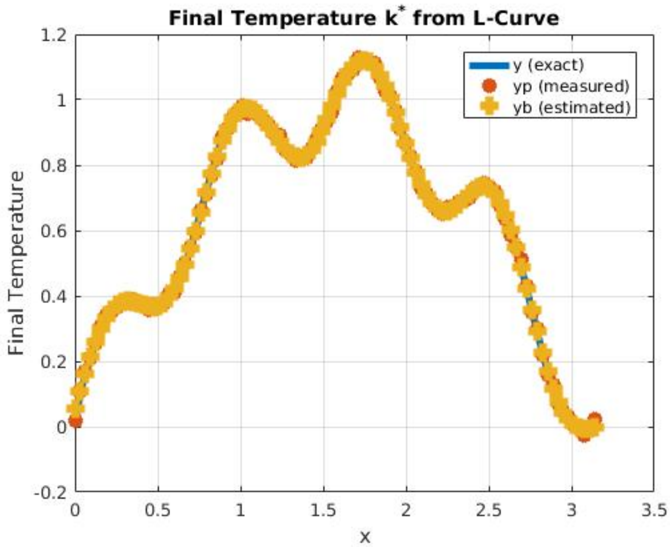
Initial temperature is made with $k^* = 20$. The error is 0.1146 and the error ratio is 1.5972. It is a little bit larger than the error ratio with discrepancy method.

Figure 6: L-Curve with C-Norm



The error of final temperature is 0.0724 and the error ratio is 1.0228%, much lower than previous error ratio 1.1177%.

Figure 7: L-Curve with C-Norm



MATLAB code:

```
close all;
clear;
L = pi;
n= 100;
T = 5;
x=linspace(0,100,n)/100*L;
u0=sin(x)+0.2*sin(8*x);
alpha = 10^-3;
C = zeros(n,n);
for i=1:100
C(i,i) = 2;
if i < 100
C(i,i+1) = -1;
C(i+1, i) = -1;
else
C(i,i-1) = -1;
end
end
h = L/(n-1);
C = C*alpha/h^2;
A = expm(-T*C);
ut=A*u0';
% applying homogeneous boundary condition
ut(1)=0;
ut(end)=0;
uT_meas = ut+10^-2*randn(n,1);
```

```
yp = uT_meas;

% building laplacian operator for a vector

%% (a)
        beta = 1;
        discrepancy = zeros(k,1);
        flag = 0;
        xi = zeros(n, 1);
        k = 250;
        for i=1:k
        xi = xi - beta* A'*(A*xi - yp);
        discrepancy(i) = norm(A*xi-yp);
        xi_v(i) = norm(xi);
        xi_cv(i) = norm(C*xi);
        if (discrepancy(i) <= norm(uT_meas - ut)) &&(flag ==0);
        xi_k = xi;
        kk = i;
        disp(kk);
        flag =1;
        end
        end

        dy = ones(k) * norm(uT_meas -ut);
        if flag ==1
        fig1=figure( );
        semilogy(discrepancy, 'linewidth',5,'color','b');
        hold on
        semilogy(dy,'--','linewidth',5,'color','r');
        hold off
        grid;
        legend('Discrepancy curve','dy');
        xlabel('k');
        axis([0 100 0.05 0.5]);
        ylabel('Discrepancy');


        y=ut;
        yb = A* xi_k;
        fig2=figure();
        sizeofline=4;
        sizeofline2=6;
        plot(x, y,'linewidth',sizeofline);
        hold on;
        plot(x, yp,'*','linewidth',sizeofline2);
```

```
        plot(x, yb,'s','linewidth',sizeofline2);
        xlabel('x');
        ylabel('Final Temperature');
        grid;
        title('Final Temperature');
        legend('y (exact)','yp (measured)','yb (estimated)');

        fig3 = figure();
        plot(x, u0,'linewidth',sizeofline);
        hold on;
        plot(x, xi_k,'*','linewidth',sizeofline2);
        hold off
        legend('u0 (exact)','xi_k (estimated)');
        title('Initial Temperature');
        end

%% (b)
        starting_index=3;
        figure;
        loglog(discrepancy(starting_index:end),xi_v(starting_index
           :end),'*', 'linewidth',5);
        title('Discrepancy vs xi-norm');
        axis tight;
        xlabel('Discrepancy');
        ylabel('xi');

%% (c)



        fig5=figure( );
        loglog(discrepancy(starting_index:end,1),xi_cv(
           starting_index:end),'*', 'linewidth',5);
        title('Discrepancy vs C-norm');
        xlabel('Discrepancy');
        ylabel('C-norm');
        axis tight;


% Initial temperature and Final temperature estimation from (c)

        % calculating xi when k* = 20
        xi = zeros(n, 1);
        k = 20;
        for i=1:k
        xi = xi - beta* A'*(A*xi - yp);
```

```
discrepancy(i) = norm(A*xi-yp);
xi_v(i) = norm(xi);
xi_cv(i) = norm(C*xi);
if (discrepancy(i) <= norm(uT_meas - ut)) &&(flag ==0);
xi_k = xi;
kk = i;
disp(kk);
flag =1;
end
end


y=ut;
yb = A* xi;
figure;
plot(x, y,'linewidth',sizeofline);
hold on;
plot(x, yp,'*','linewidth',sizeofline2);
plot(x, yb,'s','linewidth',sizeofline2);
xlabel('x');
ylabel('Final Temperature');
grid;
title('Final Temperature k^* from L-Curve');
legend('y (exact)','yp (measured)','yb (estimated)');


figure();
plot(x, u0,'linewidth',sizeofline);
hold on;
plot(x, xi_k,'*','linewidth',sizeofline2);
hold off
legend('u0 (exact)','xi_k (estimated)');
title('Initial Temperature');
```