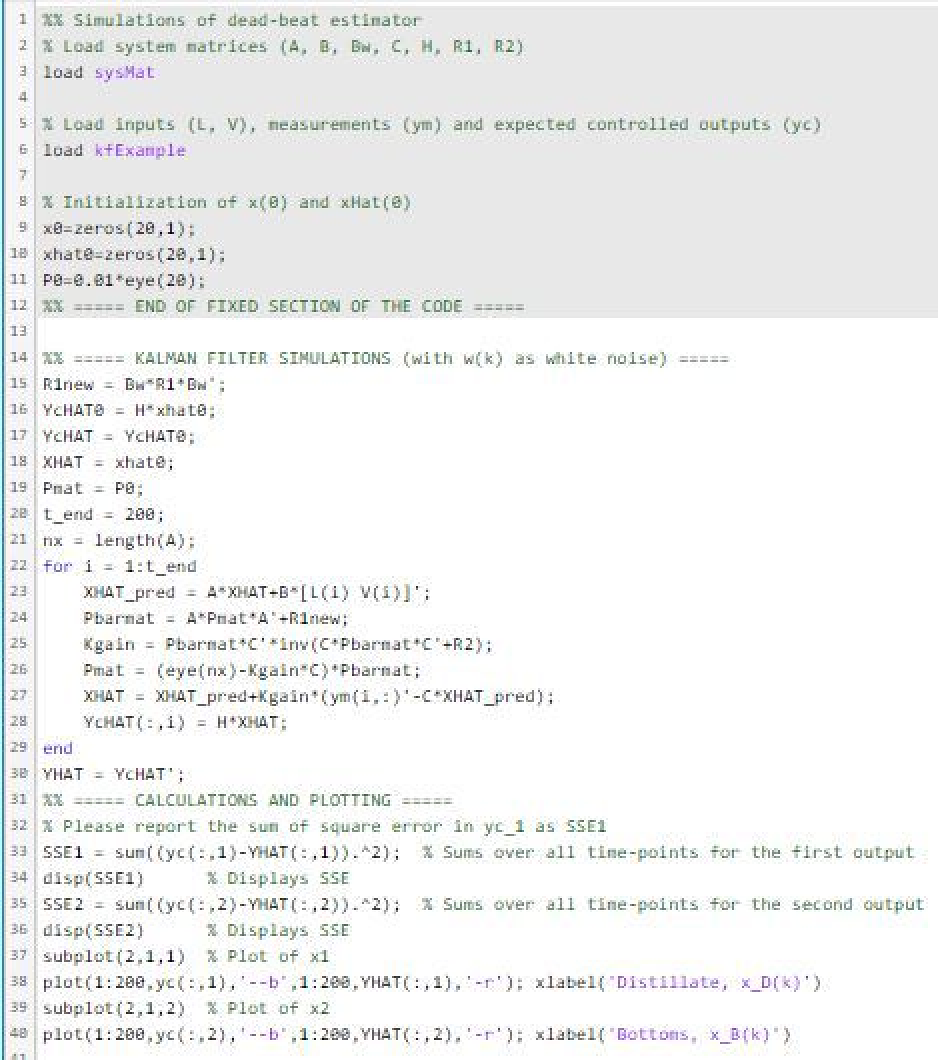
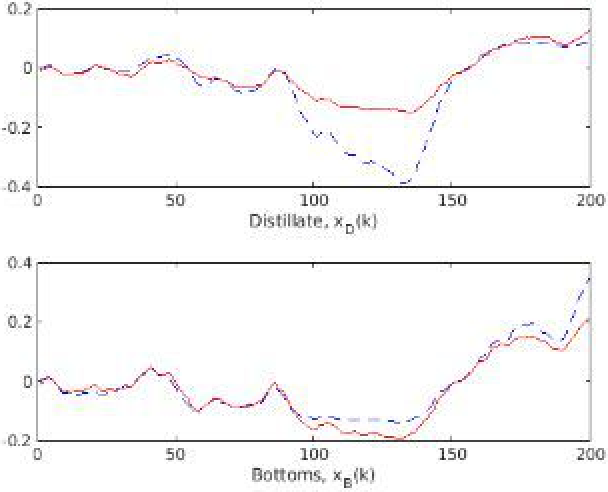
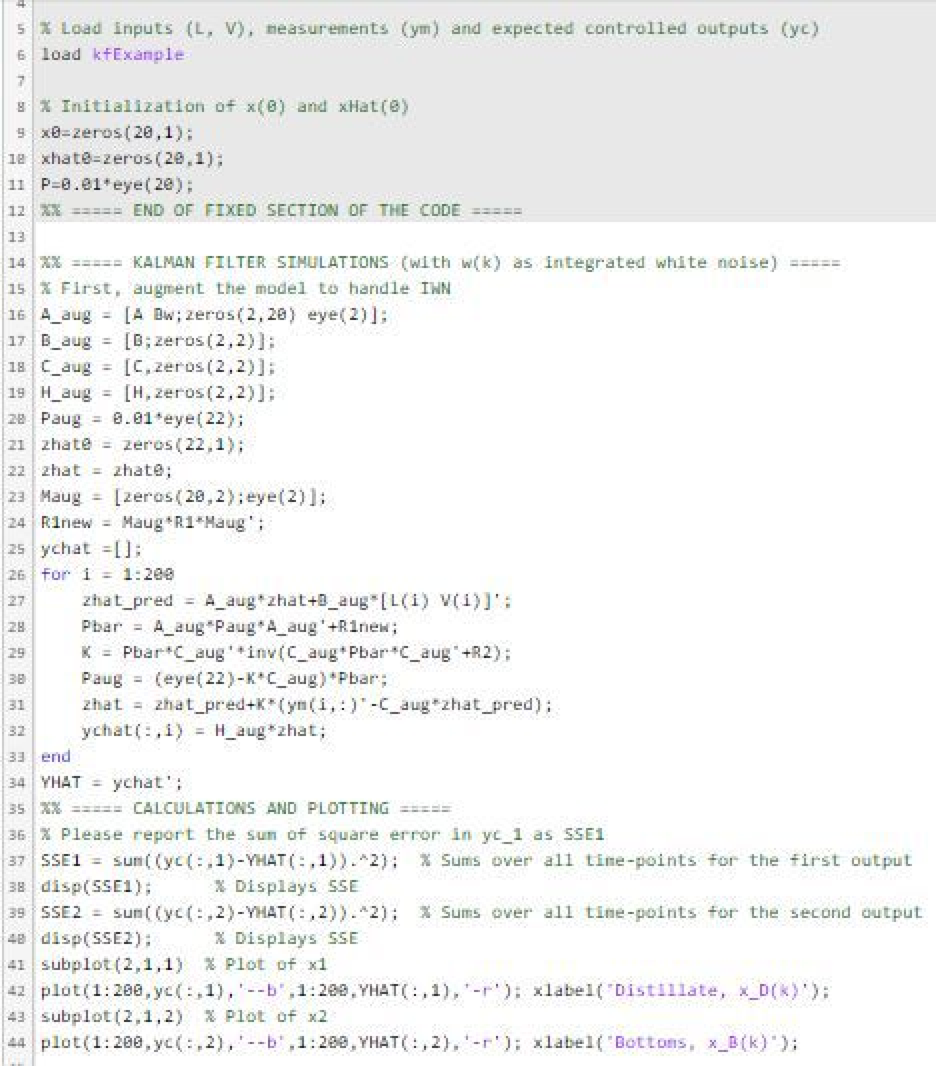


```
%% Simulations of dead-beat estimator
% Load system matrices (A, B, Bw, C, H, R1, R2)
load sysMat

% Load inputs (L, V), measurements (ym) and expected controlled outputs (yc)
load kfExample

% Initialization of x(0) and xHat(0)
x0=zeros(20,1);
xhat0=zeros(20,1);
P0=0.01*eye(20);
%% ===== END OF FIXED SECTION OF THE CODE =====

%% ===== KALMAN FILTER SIMULATIONS (with w(k) as white noise) =====
R1new = Bw*R1*Bw';
YcHAT0 = H*xhat0;
YcHAT = YcHAT0;
XHAT = xhat0;
Pmat = P0;
t_end = 200;
nx = length(A);
for i = 1:t_end
    XHAT_pred = A*XHAT+B*[L(i) V(i)]';
    Pbarmat = A*Pmat*A'+R1new;
    Kgain = Pbarmat*C'*inv(C*Pbarmat*C'+R2);
    Pmat = (eye(nx)-Kgain*C)*Pbarmat;
    XHAT = XHAT_pred+Kgain*(ym(i,:)'-C*XHAT_pred);
    YcHAT(:,i) = H*XHAT;
end
YHAT = YcHAT';
%% ===== CALCULATIONS AND PLOTTING =====
% Please report the sum of square error in yc_1 as SSE1
SSE1 = sum((yc(:,1)-YHAT(:,1)).^2);  % Sums over all time-points for the first output
disp(SSE1)       % Displays SSE
SSE2 = sum((yc(:,2)-YHAT(:,2)).^2);  % Sums over all time-points for the second output
disp(SSE2)       % Displays SSE
subplot(2,1,1)  % Plot of x1
plot(1:200,yc(:,1),'--b',1:200,YHAT(:,1),'-r'); xlabel('Distillate, x_D(k)')
subplot(2,1,2)  % Plot of x2
plot(1:200,yc(:,2),'--b',1:200,YHAT(:,2),'-r'); xlabel('Bottoms, x_B(k)')
```

```
1.4266
```

```
0.2191
```

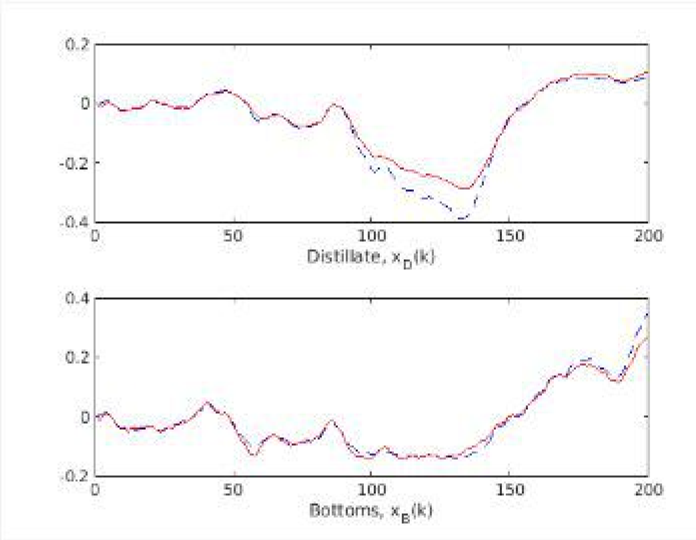

0.2
0
-0.2
-0.4
0
50
100
150
200
Distillate, $x_D(k)$
0.4
0.2
0
-0.2
0
50
100
150
200
Bottoms, $x_B(k)$

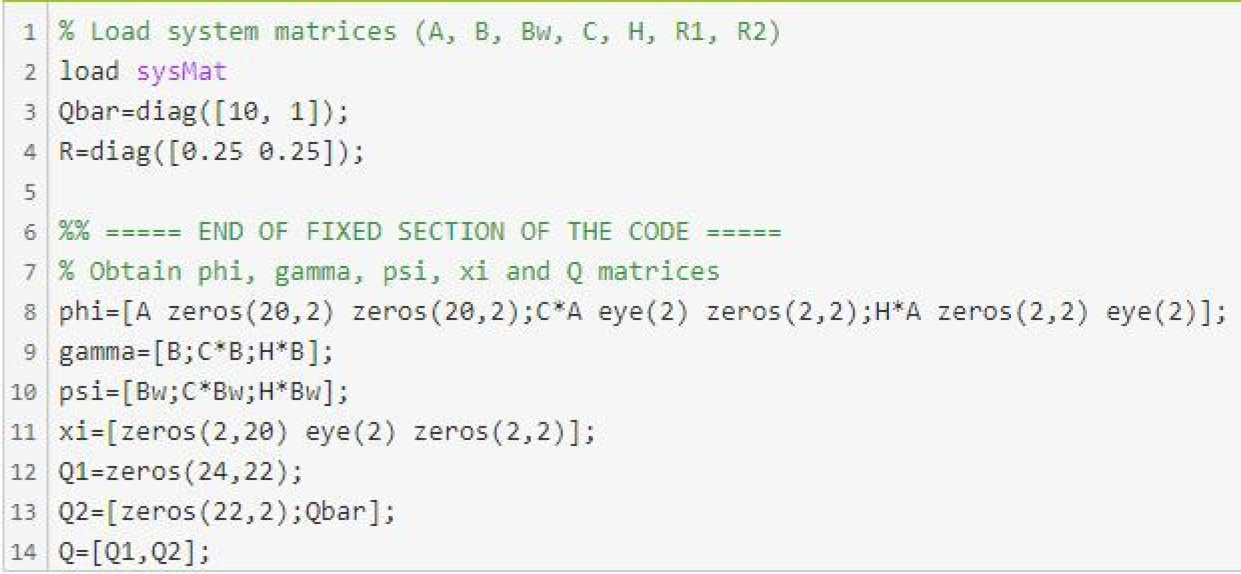


```
% Load inputs (L, V), measurements (ym) and expected controlled outputs (yc)
load kfExample

% Initialization of x(0) and xHat(0)
x0=zeros(20,1);
xhat0=zeros(20,1);
P=0.01*eye(20);
%% ===== END OF FIXED SECTION OF THE CODE =====

%% ===== KALMAN FILTER SIMULATIONS (with w(k) as integrated white noise) =====
% First, augment the model to handle IWN
A_aug = [A Bw;zeros(2,20) eye(2)];
B_aug = [B;zeros(2,2)];
C_aug = [C,zeros(2,2)];
H_aug = [H,zeros(2,2)];
Paug = 0.01*eye(22);
zhat0 = zeros(22,1);
zhat = zhat0;
Maug = [zeros(20,2);eye(2)];
R1new = Maug*R1*Maug';
ychat =[];
for i = 1:200
    zhat_pred = A_aug*zhat+B_aug*[L(i) V(i)]';
    Pbar = A_aug*Paug*A_aug'+R1new;
    K = Pbar*C_aug'*inv(C_aug*Pbar*C_aug'+R2);
    Paug = (eye(22)-K*C_aug)*Pbar;
    zhat = zhat_pred+K*(ym(i,:)'-C_aug*zhat_pred);
    ychat(:,i) = H_aug*zhat;
end
YHAT = ychat';
%% ===== CALCULATIONS AND PLOTTING =====
% Please report the sum of square error in yc_1 as SSE1
SSE1 = sum((yc(:,1)-YHAT(:,1)).^2);  % Sums over all time-points for the first output
disp(SSE1);      % Displays SSE
SSE2 = sum((yc(:,2)-YHAT(:,2)).^2);  % Sums over all time-points for the second output
disp(SSE2);      % Displays SSE
subplot(2,1,1)  % Plot of x1
plot(1:200,yc(:,1),'--b',1:200,YHAT(:,1),'-r'); xlabel('Distillate, x_D(k)');
subplot(2,1,2)  % Plot of x2
plot(1:200,yc(:,2),'--b',1:200,YHAT(:,2),'-r'); xlabel('Bottoms, x_B(k)');
```

```
0.2332
```

```
    0.0521
```


0.2
0
-0.2
-0.4
0
50
100
150
200
Distillate, $x_D(k)$
0.4
0.2
0
-0.2
0
50
100
150
200
Bottoms, $x_B(k)$



```
% Load system matrices (A, B, Bw, C, H, R1, R2)
load sysMat
Qbar=diag([10, 1]);
R=diag([0.25 0.25]);

%% ===== END OF FIXED SECTION OF THE CODE =====
% Obtain phi, gamma, psi, xi and Q matrices
phi=[A zeros(20,2) zeros(20,2);C*A eye(2) zeros(2,2);H*A zeros(2,2) eye(2)];
gamma=[B;C*B;H*B];
psi=[Bw;C*Bw;H*Bw];
xi=[zeros(2,20) eye(2) zeros(2,2)];
Q1=zeros(24,22);
Q2=[zeros(22,2);Qbar];
Q=[Q1,Q2];
```