```
clsc; clear; close all;

N1=300;     % size of the array
X1 = zeros(N1,1);   % real data
Z1 = zeros(N1,1);   % measurements
sw = 28^2;      % sigma_w^2
se = 97^2;      % sigma_Eita^2
x1 = sw / se;   % xita
alpha1 = (-x1 + sqrt(x1^2 +4*x1))/2;    % coefficient for exponential smoothing
M1 = (2-alpha1)/alpha1;                 % window size for running mean


X1(1) = 10;
Z1(1) = X1(1) + normrnd(0,sqrt(se));

for i = 2:N1
    X1(i) = X1(i-1) + normrnd(0,sqrt(sw));
    Z1(i) = X1(i) + normrnd(0,sqrt(se));
end

% forward exponential
X1_exp= zeros(N1,1);    % forward exponential smoothed trajectory.
X1_exp(1) = Z1(1);
for i = 2:N1
X1_exp(i) = X1_exp(i-1) + alpha1 * (Z1(i) - X1_exp(i-1));
end

% backward exponential
X1_back= zeros(N1,1);   % backward exponential smoothed trajectory.
X1_back(N1) = X1_exp(N1);
for i = N1-1:-1:1
X1_back(i) = X1_exp(i+1) + alpha1 * (X1_exp(i) - X1_back(i+1));
end

% running mean
X1_mean= zeros(N1,1);   % running mean smoothed trajectory.
M1 = round((M1-1)/2)*2+1;
m1=(M1-1)/2;
X1_mean(1:m1)=sum(Z1(1:m1))/m1;
X1_mean(N1-m1+1:N1)=sum(Z1(N1-m1+1:N1))/m1;
for i =  (m1+1):N1-m1
X1_mean(i) = 1/M1 * sum(Z1(i-m1:i+m1));
end

figure(1)
subplot(2,1,1)
plot(1:N1,Z1,1:N1,X1,1:N1,X1_exp,1:N1, X1_back)
title('Figure (1.1) Measurements, Real data, F_exponential, and B_exponential')
xlabel('Steps')
ylabel('Data')
legend({'measurnments','data','forward','backward'})
```

```
subplot(2,1,2)
plot(1:N1,Z1,1:N1,X1,1:N1,X1_mean)
title('Figure (1.2) Real data, measurements,running mean')
xlabel('Steps')
ylabel('Data')
legend({'data','measurnments','mean'})

Id1_exp = sum((Z1-X1_back).^2)     % deviation indecator of backward smoothing
Iv1_array_exp = zeros(N1,1);
for i = 1:N1-2
    Iv1_array_exp(i)=(X1_back(i+2) - 2*X1_back(i+1) + X1_back(i))^2;
end
Iv1_exp = sum(Iv1_array_exp)       % variability indecator of backward smoothing

Id1_mean = sum((Z1-X1_mean).^2)    % deviation indecator of Running mean.
Iv1_array_mean = zeros(N1,1);
for i = 1:N1-2
    Iv1_array_mean(i)=(X1_mean(i+2) - 2*X1_mean(i+1) + X1_mean(i))^2;
end
Iv1_mean = sum(Iv1_array_mean)     % variability indecator of Running mean.

N = 300;
X = zeros(N,1);     % true trajectory points
V = zeros(N,1);     % true trajectory points
Z = zeros(N,1);     % measurnments
X(1) = 5;           % initial condition
V(1) = 0;           % initial condition
Z(1) = X(1) + normrnd(0,sqrt(500));
T = 0.1;
for i = 2:N
    a = normrnd(0,sqrt(10));
    X(i) = X(i-1) + V(i-1) * T +  (a*T^2) /2;
    V(i) = V(i-1) + a*T;
    Z(i) = X(i) + normrnd(0,sqrt(500));
end

% plot
figure(2)
plot(1:N,X,1:N,Z)
title('Figure (2) Real data vs measurements')
xlabel('Steps')
ylabel('Data')
legend({'real data','measurements'})

X_exp= zeros(N,1);
X_exp(1) = Z(1);
Iv_array_exp = zeros(N,1);

figure(3);

k=1; % dummy variable (counter)
```

```
for alpha = 0.01:0.01:0.23
    for i = 2:N
        X_exp(i) = X_exp(i-1) + alpha * (Z(i) - X_exp(i-1));
    end
    Id_exp(k) = sum((Z-X_exp).^2);
    for i = 1:N-2
        Iv_array_exp(i)=(X_exp(i+2) - 2*X_exp(i+1) + X_exp(i))^2;
    end
    Iv_exp(k) = sum(Iv_array_exp);
    alpha_array(k) = alpha;

    subplot(5,5,k);
    %plot(1:N,Z,1:N,X,1:N,X_exp)
    plot(1:N,X,1:N,X_exp)
    xlabel('Steps')
    ylabel('Data')
    title(sprintf('alpha = %.2f', alpha))

    k=k+1;
end
subplot(5,5,25)
plot(0,0,0,0,0,0)
axis off
%legend({'Measurements','Real Data','Exponential Data'})
legend({'Real Data','Exponential Data'})

X_mean= zeros(N,1);

figure(4);
%title('Figure (3) Real data, Running mean data, and  measurnments')
k=1;
for M = 7:2:35
    m=(M-1)/2;
    X_mean(1:m)=sum(Z(1:m))/m;
    X_mean(N-m+1:N)=sum(Z(N-m+1:N))/m;
    for i =  (m+1):N-m
    X_mean(i) = 1/M * sum(Z(i-m:i+m));
    end
Id_mean(k) = sum((Z-X_mean).^2);
Iv_array_mean = zeros(N,1);
for i = 1:N-2
    Iv_array_mean(i)=(X_mean(i+2) - 2*X_mean(i+1) + X_mean(i))^2;
end
Iv_mean(k) = sum(Iv_array_mean);
M_array(k) = M;

subplot(4,4,k);
%plot(1:N,Z,1:N,X,1:N,X_mean)
plot(1:N,X,1:N,X_mean)
xlabel('Steps')
ylabel('Data')
```

```
title(sprintf('M = %.2f', M))

k=k+1;
end
subplot(4,4,16)
plot(0,0,0,0,0,0)
axis off
%legend({'Measurements','Real Data','Running mean Data'})
legend({'Real Data','Smoothed Data'})

M_used = 11;            % chosen window size M
alpha_used = 0.08;      % chosen alpha
alpha_index=0;
for i=1:length(alpha_array)
if alpha_array(i) > alpha_used-0.01  && alpha_array(i) < alpha_used+0.01
    alpha_index=i;
end
end
Id_exp_value = Id_exp(alpha_index)  % Deviation indicator for exponential smoothing
Iv_exp_value = Iv_exp(alpha_index)  % Variability indecator of forward smoothing
M_index = find(M_array == M_used);
Id_mean_value = Id_mean(M_index)    % Deviation indecator of Running mean.
Iv_mean_value = Iv_mean(M_index)     % Variability indecator of Running mean.
% Plot
X_mean= zeros(N,1);
m=(M_used-1)/2;
X_mean(1:m)=sum(Z(1:m))/m;
X_mean(N-m+1:N)=sum(Z(N-m+1:N))/m;
for i =  (m+1):N-m
X_mean(i) = 1/M_used * sum(Z(i-m:i+m));
end

X_exp= zeros(N,1);
X_exp(1) = Z(1);
for i = 2:N
X_exp(i) = X_exp(i-1) + alpha_used * (Z(i) - X_exp(i-1));
end
figure(5)
plot(1:N,X,1:N,X_exp,1:N,X_mean,'g')
xlabel('Steps')
ylabel('Data')
title(sprintf('Figure(5) alpha = 0.08, M = 11'))
legend({'Real Data','exponential Data','RunningMean Data'})

Nc = 200;         % Size of this cyclic trajectory
A = zeros(Nc,1);
Xc = zeros(Nc,1);   % Data for the cyclic trajectory
Zc = zeros(Nc,1);   % measurnments

T = 32;           % Period of estimation
w = 2*pi/T;       % (Omega) angle frequency
```

```
A(1) = 1;         %  Initial condition
Xc(1) = A(1) +  sin(w + 3);
Zc(1) = Xc(1) + normrnd(0,sqrt(0.05));

for i = 2:Nc
    A(i) = A(i-1) +  normrnd(0,0.08);
    Xc(i) = A(i) +  sin(w*i + 3);
    Zc(i) = Xc(i) + normrnd(0,sqrt(0.05));
end

% plot
figure(6)
plot(1:Nc,Xc,1:Nc,Zc)
title('Figure (6) Real cyclic data vs measurnments')
xlabel('Steps')
ylabel('Data')
legend({'real data','measurements'})

Mc= 13;     % window size used to smooth the cyclic trajectory data
mc=(Mc-1)/2;
X_mean_c = zeros(Nc,1);
X_mean_c(1:mc)=sum(Zc(1:mc))/mc;
X_mean_c(Nc-mc+1:Nc)=sum(Zc(Nc-mc+1:Nc))/mc;
for i =  (mc+1):Nc-mc
X_mean_c(i) = 1/Mc * sum(Zc(i-mc:i+mc));
end

figure(7)
plot(1:Nc,Zc,1:Nc,Xc,1:Nc,X_mean_c)
xlabel('Steps')
ylabel('Data')
legend({'measurements','real data','smoothed data'})
title(sprintf('Figure(7) Running mean model with M = %.2f', Mc))

figure(8);

T_array=[15 21 40];

for k = 1:3
T = T_array(k);
    w = 2*pi/T;     % (Omega) angle frequency
    A1 = 1;         %  Initial condition
    Xc(1) = A(1) +  sin(w + 3);
    Zc(1) = Xc(1) + normrnd(0,sqrt(0.05));

    for i = 2:Nc
        A(i) = A(i-1) +  normrnd(0,0.08);
        Xc(i) = A(i) +  sin(w*i + 3);
        Zc(i) = Xc(i) + normrnd(0,sqrt(0.05));
    end
    Mc = 21;  % worked with group 4 data
```

```
    mc=(Mc-1)/2;
    X_mean_c(1:mc)=sum(Zc(1:mc))/mc;
    X_mean_c(Nc-mc+1:Nc)=sum(Zc(Nc-mc+1:Nc))/mc;
    for i =  (mc+1):Nc-mc
        X_mean_c(i) = 1/Mc * sum(Zc(i-mc:i+mc));
    end
    subplot(3,1,k)
    plot(1:Nc,Zc,1:Nc,Xc,1:Nc,X_mean_c)
    xlabel('Steps')
    ylabel('Data')
    title(sprintf('Figure (8) T = %.2f', T))
    legend({'Measurements','Real Data','Running mean Data'})
end
```