```
clc
close all

%Implementation of the nuclear reactor %problem

prior=0.077*[1 1 1 1 1 1 1 1 1 1 1 1 1];  % %0 to 6 failures per year discretized into %equal intervals  so p = 1/13 = 0.077

 lambda=[0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6];  % 0 to 6 failures per year %discretized into equal intervals

 L1=(((3*lambda).^5)/factorial(5)) ;

 L2= exp(-3*lambda);

 L=L1.*L2; %Liklihood

 L_p= L.*prior; % Posterior

 posterior= L_p/sum(L_p); % Normalized

 xplot=[0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6];

figure, bar(xplot,posterior,'r');ylim([0 0.5]); ylabel('Probability');
xlabel('Failures per operating year');
hold on
bar(xplot,prior,'b');ylim([0 0.5]); ylabel('Probability');
xlabel('Failures per operating year');
```

```
clc
close all


 n = 20; %Sample size
 sigma = 20; % Gaussian distribution with known sigma
%x = normrnd(50,sigma,n,1); %Generate random number and
save it in a file
%
% clear all

 load data_gaussian_mu50_sig20.mat  %save it in a file so that
the random sequence doesnt change everytime you run it


%=====================================%=
==============================

mu = 30;    %hyperparameters
tau = 20;


theta = linspace(-40, 100, 500); dth=(100-(-40))/500;

y1 = normpdf(theta,mean(x),sigma/sqrt(n));  % Liklihood:
Derive the formula yourself, %sigma known

y2 = normpdf(theta,mu,tau);                % Prior


postMean = tau^2*mean(x)/(tau^2+sigma^2/n) +
sigma^2*mu/n/(tau^2+sigma^2/n);
% Using formula for known sigma

postSD = sqrt(tau^2*sigma^2/n/(tau^2+sigma^2/n));
```

```
% Using formula for known sigma

y3 = normpdf(theta, postMean,postSD);
% Posterior


y_post_Nr=y1.*y2;   % Likelihood x prior

sum=sum(y_post_Nr)*dth; % Denominstor
y_post=y_post_Nr/sum;
%Posterior using Bayes rule

 plot(theta,y1,'m:',theta,y2,'k--',theta,y3,'b','linewidth',2); hold on
 plot(theta,y_post,'o-r','linewidth',1); xlim([-30 80])

legend('Likelihood','Prior','Posterior-formula','Using bayes rule')
xlabel('Range of RV X'); ylabel('PDF of X')
```

```
clc
close all

   n = 10; %keep changing this parameter
 sigma = 20; % Gaussian distribution with known sigma

 % x = normrnd(50,sigma,n,1); %Generate random number and
save it in a file
% clear

load data_gaussian_mu50_sig20_10samples.mat  %save it in a
file so that the random sequence doesnt change everytime you
run it

%=======================================%=
================================

mu = 30;    %hyperparameters
tau = 20;
nt=500;



theta = linspace(-40, 100, nt); dth=(100-(-40))/nt;

fold=zeros(1,nt);

%---------------------------------------%---------------------------
-
%=====================================
%% Using Formula
```

```
%=====================================

y1 = normpdf(theta,mean(x),sigma/sqrt(n));
% Liklihood: Derive the formula yourself, sigma known

y2 = normpdf(theta,mu,tau);              % Prior


postMean = tau^2*mean(x)/(tau^2+sigma^2/n) +
sigma^2*mu/n/(tau^2+sigma^2/n);
% Using formula for known sigma

postSD = sqrt(tau^2*sigma^2/n/(tau^2+sigma^2/n));


y3 = normpdf(theta, postMean,postSD);
% Posterior using formula

%---------------------------------------------------------------------

   for i=1:9

       data=x(1:i); nn=length(data);

       y1 = normpdf(theta,mean(data),sigma/sqrt(nn));
 % Liklihood: Derive the formula yourself, sigma known

       y2 = normpdf(theta,mu,tau);              % Prior

       y_post_Nr=y1.*y2*dth;   %Posterior using Bayes rule
```

```
        summ=sum(y_post_Nr)*dth;
        y_post=y_post_Nr/summ;

%         fold=fold+y_post;


         %subplot(1,2,1)
         plot(theta,y2,'k--',theta,y3,'b','linewidth',3); hold on
         plot(theta,y_post,'--r','linewidth',2); xlim([-30 80]);
         xlabel('Range of RV X'); ylabel('PDF of X')
         legend('Prior','Posterior using formula','Posterior--
Bayesian update')
%         subplot(1,2,2)
%          plot(theta,y1,'k-',theta,y_post,'--r','linewidth',2);  xlim([-
30 80]);
%          xlabel('Range of RV X'); ylabel('PDF of X')
%          legend('Likelihood','Posterior--Bayesian update')
    drawnow

      end
```

```
clc

close all

global Mu0 Sigma0 Sigma D

Mu0=[12 28]; %Prior on mu: hyperparameter-1 (mean of mu)

Sigma0=[9 0;
        0 8];  %Prior on mu: hyperparameter-2 (cov-mat of mu)

Sigma=[16 0;
        0 9];  % Known: covariance matrix of C = {16 0; 0 9}

D=[10.3  34.2 ;
   12.2  31.1;
   8.5   35.7;
   14.2  30.8];  % Data on c & phi

a1=0; a2=30; delta1=0.2; theta1=a1:delta1:a2;

b1=20; b2=40; delta2=0.2; theta2=b1:delta2:b2;

for i=1:length(theta1)
    for j=1:length(theta2)
    %q(i,j)=qfun([theta1(i),theta2(j)]);
                    x=[theta1(i),theta2(j)];
                    Mu=x;
                    like=mvnpdf(D,Mu,Sigma);
                    prior=mvnpdf(Mu,Mu0,Sigma0);
                    q(i,j)=prod(like)*prior;
```

```
    end
end
f12=q/sum(sum(q))/delta1/delta2;

for i=1:length(theta1)
    f1(i)=sum(sum(q(i,:)));
end
f1=f1/sum(f1*delta1);

for k=1:length(theta2)
    f2(k)=sum(sum(q(:,k)));
end
f2=f2/sum(f2*delta2);

[mu1,var1]=MargStat(theta1,f1,delta1);
[mu2,var2]=MargStat(theta2,f2,delta2);

for i=1:length(theta1)
    for j=1:length(theta2)
    var12=(theta1(i)-mu1)*(theta2(j)-mu2)*f12(i,j);
    end
end
var12=sum(sum(var12))*delta1*delta2;
postmean=[mu1 mu2]
postcovariance=[var1 var12;
                var12 var2]
```