
```

% Problem 3 code and result:

clear all
clc
Emu = 29*10^6; % lb/in^2
Esig = 2*10^6; % lb/in^2

Pmu = 50000; % lb
Psig = 10000; % lb

Wmu = 83.33; % lb/in
Wsig = 8.33; % lb/in

L = 360; % in
I = 1330; % in^4

stDev3 = (fullfact([3, 3, 3])-2)*3; % define 3 level factorial

% Define stDev for given distribution using 'fracDes'
stDev3(:,1) = stDev3(:,1)*Psig;
stDev3(:,2) = stDev3(:,2)*Esig;
stDev3(:,3) = stDev3(:,3)*Wsig;

% Define vector of means to define samples
L1=length(stDev3);
sampleMean(:,1) = Pmu*ones(L1,1);
sampleMean(:,2) = Emu*ones(L1,1);
sampleMean(:,3) = Wmu*ones(L1,1);

dataSamp = sampleMean + stDev3;

% next need to define the limit state function. We do not want beam
% displacement to be greater than 2 inches. Therefore following limit
% state function shall be used.
% -----
% 
$$g(P, E, W) = 2 - [(PL^3)/(48EI) + (5/385)*((W*L^4)/(E*I))] = 0$$

% -----
% If  $g(P, E, W) < 0$  then we have failure

% define response based on the data samples 'dataSamp'
LSF = @(P, E, W) 2-((P*L^3)./(48*E*I) + (5/385)*((W*L^4)./(E*I)));
sysResp = LSF(dataSamp(:,1), dataSamp(:,2), dataSamp(:,3));

% perform quadratic regression
quadReg = fitlm(dataSamp, sysResp, 'purequadratic');
% retrieve fitted coefficients
a = table2array(quadReg.Coefficients(:,1));

% define surrogate model
SurModel = @(P,E,W) a(1) ...
+ a(2).*P + a(3).*E + a(4).*W ...
+ a(5).*P.^2 + a(6).*E.^2 + a(7).*W.^2;

```

```
% Implement MCS sampling with million samples
N = 1000000;
Nvar = 3;

% Define random vector U(N, Nvar)
U=rand(N,Nvar);

% Generate samples
XP = Pmu + Psig*icdf('norm', U(:,1), 0, 1);
XE = Emu + Esig*icdf('norm', U(:,2), 0, 1);
XW = Wmu + Wsig*icdf('norm', U(:,3), 0, 1);

% Calculate surrogate response using generated samples XP, XE, XW
SurResp = SurModel(XP, XE, XW);

% Calculate failure probability
Pf = sum(SurResp<0)/N

Pf =

    2.1938e-01
```

Published with MATLAB® R2014b