Contents

```
clc
clear all;
```

```
mydir = '/Users/vignesh/Desktop/RPI/Fall 2021/MDO/Project_3/GPML_OPT_code';
addpath(mydir(1:end-1))
addpath([mydir,'cov'])
addpath([mydir,'doc'])
addpath([mydir,'inf'])
addpath([mydir,'lik'])
addpath([mydir,'mean'])
addpath([mydir,'prior'])
addpath([mydir,'util'])
```

sample the function

```
% r_2
r_2_in = 0.1;
r_2_out = 1.5;

% \alpha_1
alpha1_in = 0;
alpha1_out= 0.3;

% \omega
omega_in = 2*pi*3/60;
omega_out = 2*pi*8/60;

% Range vector
% [ r_2(in) r_2(out)]
% [ alp(in) alp(out)]
% [ omg(in) omg(out)]
Range = [r_2_in r_2_out;alpha1_in alpha1_out;omega_in omega_out];


% sampling process
bins = 1000;
dim = length(Range);
x = samplingProject3(Range,bins,dim);

% true objective generation from sampled data points
tau_int = 1000;
tau = 0:0.01:tau_int;
y = zeros(length(x),1);
Y0 = [pi/3;0];

for i=1:length(x)
```

```
    y(i,1) = ODEsim(Y0,tau,x(i,:));
end
```

train gaussian process surrogate

```
% set the squared exponential covariance function
covfunc = {@covMaterniso,1};  % @covSEiso
hyp.cov = [log(0.3); log(1)]; % first component is log(l) and second is log(sigma)

% set the likelihood function to Gaussian
likfunc = @likGauss;
sn = 0.25; %1e-16; % this is the noise level
hyp.lik = log(sn);

% maximize the likelihood function to find the hyperparameters
hyp = minimize(hyp, @gp, -300, @infExact, [], covfunc,...
    likfunc, x, y);
```

Comparison of surrogate model with Chaotic ODE

```
% Generate a 1-D slice of the function along \omega
z1 = 0.8;  % r_2 - constant
z2 = 0.058; % \alpha_1 - constant
Om = omega_in:0.005:omega_out; % range of omega

z = [ones(length(Om),1)*z1 ones(length(Om),1)*z2 Om'];
% surrogate approximates of the true objective (1-D slice)
[m s2] = gp(hyp, @infExact, [], covfunc, likfunc, x, y, z);

m2 = zeros(length(Om),1);
for i=1:length(Om)
    % True objective values along the 1-D slice
    m2(i) = ODEsim(Y0,tau,z(i,:));
end

% computing surrogate error
surrErr = compareSurrogate(m,m2,Om);
```

plot

```
figure
plot(Om,m); % surrogate model
hold on;
grid on;
plot(Om,m2); % Chaotic ODE
xlabel('$\omega$','Interpreter','latex');
ylabel('$\sigma(\frac{d\phi}{dt})$','Interpreter','latex');
legend('Surrogate','Chaotic ODE');
title('Checking accuracy of surrogate'...
    ,'Interpreter','latex');

figure
plot(Om,m2);
grid on
xlabel('$\omega$','Interpreter','latex');
ylabel('$\sigma(\frac{d\phi}{dt})$','Interpreter','latex');
title('1-D slice of objective function');
```

save variables

```
fname = input('Enter file name:','s');
save(fname,'x','y','Range','tau','hyp');
```