
```

%GP example with different values of standard deviation
%Author- Susan Sapkota
%assignment 7.2
meanfunc = []; % empty: don't use a mean
function
covfunc = {@covSum, {@covLIN, @covConst}}; % Linear covariance
function
likfunc = @likGauss; % Gaussian likelihood
hyp = struct('mean', [], 'cov', 0, 'lik', log(0.1));
sigma = 0.01;
N=100;
x=rand(N,1);
[Y,y,f]=data(100,3,sigma);
hyp2 = minimize(hyp, @gp, -100, @infGaussLik, meanfunc, covfunc,
    likfunc, Y, y);

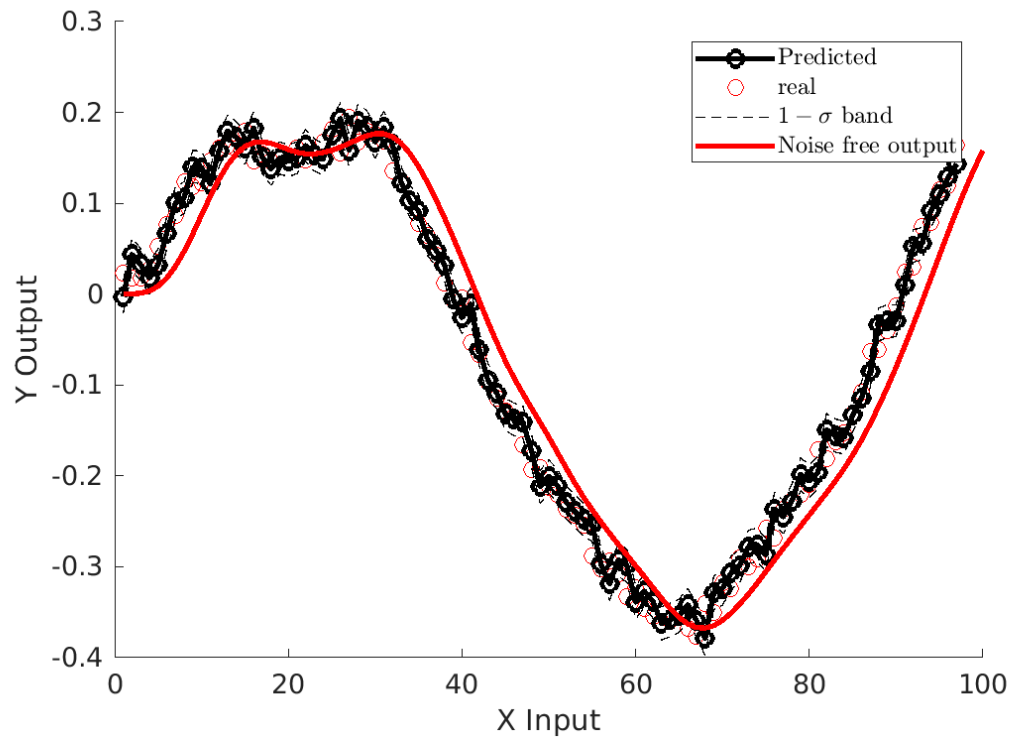
[a,b,c]=data(100,3,sigma);
[mu, s2] = gp(hyp2, @infGaussLik, meanfunc, covfunc, likfunc, Y, y,
    a);

hold on
p2 = plot(b, 'ro');
hold on
p3 = plot([1:length(mu)], mu-sqrt(s2), 'k--');
hold on
plot([1:length(mu)], mu+sqrt(s2), 'k--');
hold on
p1=plot(mu, 'ko-', 'Linewidth', 2);
hold on
xlabel('X Input')
ylabel('Y Output')
p4 = plot(c, 'r', 'Linewidth', 2);
leg1 = legend([p1,p2, p3, p4], {'Predicted', 'real', '$1-\sigma$
    band', 'Noise free output'});
set(leg1, 'Interpreter', 'latex');
hold on
function [Y,y,f]=data(N,m,sigma)
    % m: predictor input length (dimension of the input)
    % N: number of generated samples
    % sigma: noise standard deviation
    m=m+1;
    x=randn(N,1);
    [b,a]=butter(4,0.05);
    f=filter(b,a,x);
    temp=f+sigma*randn(size(x));
    temp=buffer(temp,m,m-1, 'nodelay');
    y=temp(end,:);
    Y=[temp(1:end-1,:)];
end

Function evaluation      0; Value -1.223978e+02 Function evaluation
      9; Value -2.400233e+02 Function evaluation     12; Value

```

```
-2.400837e+02 Function evaluation    20; Value -2.443864e+02
Function evaluation    22; Value -2.447056e+02 Function evaluation
    24; Value -2.447666e+02 Function evaluation    27; Value
-2.447671e+02 Function evaluation    29; Value -2.447671e+02
Function evaluation    32; Value -2.447671e+02 Function evaluation
    36; Value -2.447671e+02
```



Published with MATLAB® R2020b