testing with MLE for 1-margules
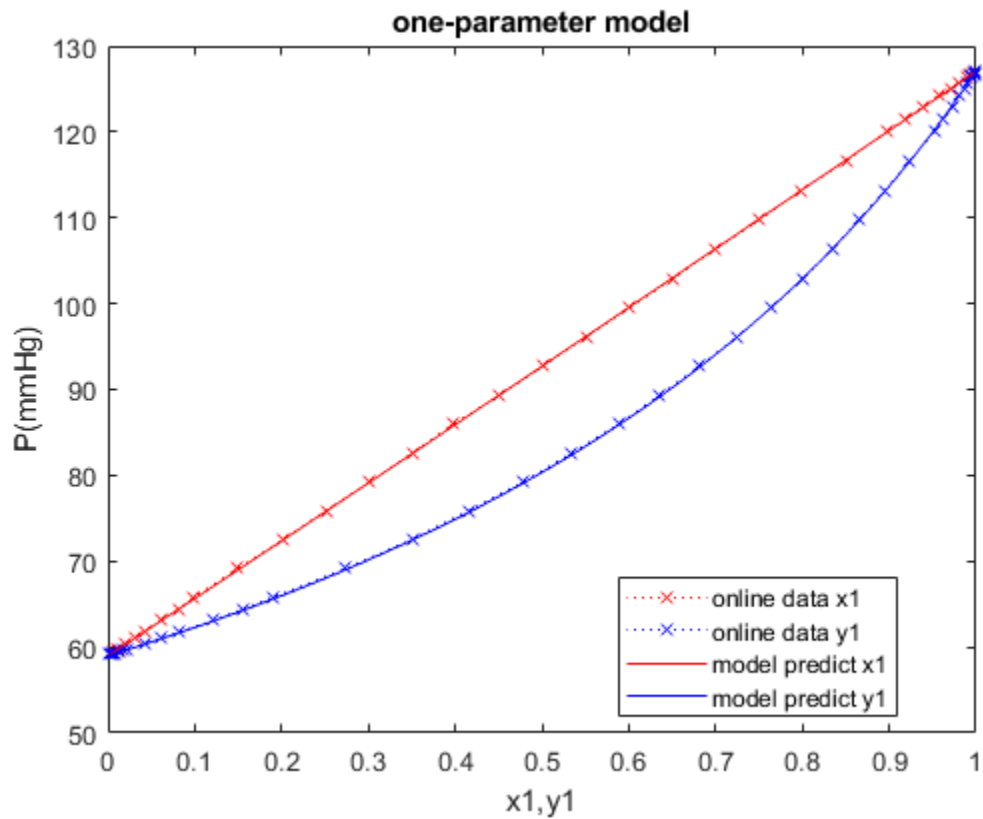
```matlab
global x2 x1
% we have margules 1-parameter model:
%Ge/RT = A12x1x2
%ln(g1) = A12(x2)^2 => g1 = exp(A12(x2)^2)
%ln(g2) = A12(x1)^2 => g2 = exp(A12(x1)^2)
%in the full form:
%P = x1*exp(A12(x2)^2)*Psat1 + x2*exp(A12(x1)^2)*Psat2
%we let beta B = A12
%ln(g1) = B*x2^2. => g1 = e^(B*x2^2)
% L(B) = exp(B*x2(1)^2)*exp(B*x2(2)^2)...*exp(B*x2(i)^2) =
% exp(B*sum(x2.^2)
%log L(B) = +sum(B*x2(i)^2)
%data and Psat can be change for different models
dat1 = load("C:\Users\buing\Documents\MATLAB\CBE
 562\meth(1)_eth(2).dat");
dat_rand = dat1;
dat1_rand_2 = dat1(4:40,2) + (- 0.002 +
 (0.004).*rand(length(dat1(4:40,2)),1));
dat1_rand_3 = dat1(2:38,3) + (- 0.002 +
 (0.004).*rand(length(dat1(2:38)),1));
dat_rand(4:40,2) = dat1_rand_2;
dat_rand(2:38,3) = dat1_rand_3;
dat = dat_rand; %randomized
Psat1 = 127.04;
Psat2 = 58.98;
x1 = dat(:,2);
y1 = dat(:,3);
x2 = 1.- dat(:,2);
opt = optimoptions('fmincon','Display','off','Algorithm','sqp');
P_1 = dat(:,1).*750;
%need to set equality of P = x1*exp(A12(x2)^2)*Psat1 +
 x2*exp(A12(x1)^2)*Psat2
%g1 = exp(A12x2^2)
% exp(A12x2^2)x1Psat1/y1 = P
% => exp(A12x2^2) = y1P/x1Psat1
%A12x2^2 = log(y1P/x1Psat1) = Beq
%
Aeq = x2.^2;
Beq = log(y1.*P_1./(x1.*Psat1));
theta= fmincon(@func,2,[],[],Aeq,Beq,[],[],[],opt);
A12 = (theta);
%theta_est = A12
%back calculate g1 and g2
x1_pre = linspace(0.001,0.999,100);
x2_pre = 1.- x1_pre;
g1_pre = exp(A12*(x2_pre.^2));
g2_pre = exp(A12*(x1_pre.^2));
P1_pre = Psat1.*x1_pre.*g1_pre + Psat2.*x2_pre.*g2_pre;
y1_pre = g1_pre.*x1_pre.*Psat1./P1_pre;
%output
```

```matlab
plot(x1,P_1,':xr')
hold on
plot(y1,P_1,':xb')
plot(x1_pre,P1_pre,'-r')
plot(y1_pre,P1_pre,'-b')
title('one-parameter model')
legend('online data x1','online data y1','model predict x1', 'model
 predict y1','location','best')
ylabel('P(mmHg)')
xlabel('x1,y1')

function LogL1 = func(theta)
global x2 x1
LogL1 = [(theta*sum(x2.^2))];
end
```