Homework 9, ME3215 Spring 2022

Table of Contents


```
Curve fitting: Least-squares fit of a sinusoid; Linearizing a non-linear equation
```

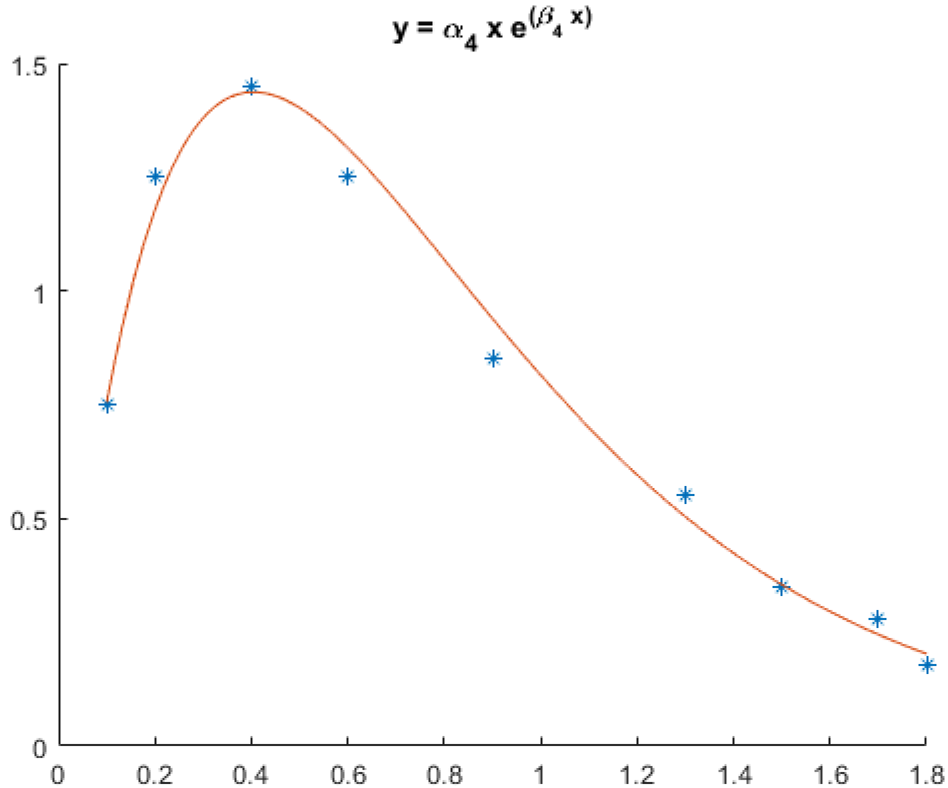
HW9_1: handwork--Linearizing a nonlinear function (textbook 14.8)

```
clear;clc;close all
%y = alpha4 * x * e ^(-beta4 * x)
%linearizes to--> ln(y)-ln(x) = ln(alpha1) - beta4 * x
%this is a line with slope beta4 & intercept alpha4

%original data
x=[0.1 0.2 0.4 0.6 0.9 1.3 1.5 1.7 1.8];
y=[0.75 1.25 1.45 1.25 0.85 0.55 0.35 0.28 0.18];

%transformed data
X = x;
Y = log(y)-log(x);

%the matlab way to solve
a = polyfit(X,Y,1);

%solving by hand...as assigned
sx=sum(X)
sy=sum(Y)
sxy=sum(X.*Y)
sx2=sum(X.^2)
xbar=mean(X)
ybar=mean(Y)
```

```
sx =

    8.5000


sy =

   -0.6095
```

```
sxy =

   -9.0399


sx2 =

   11.4500


xbar =

    0.9444


ybar =

   -0.0677

%the coefficients (linear fit)
a1=(9*sxy-sx*sy)/(9*sx2-sx^2)
a0=ybar-a1*xbar


a1 =

   -2.4733


a0 =

    2.2682

%transform the coefs back
beta4=a1
alpha4=exp(a0)


beta4 =

   -2.4733


alpha4 =

    9.6618

%plot
f=@(x)alpha4*x.*exp(beta4*x); %model
hold on
plot(x,y,'*');
fplot(f,[min(x) max(x)])
```

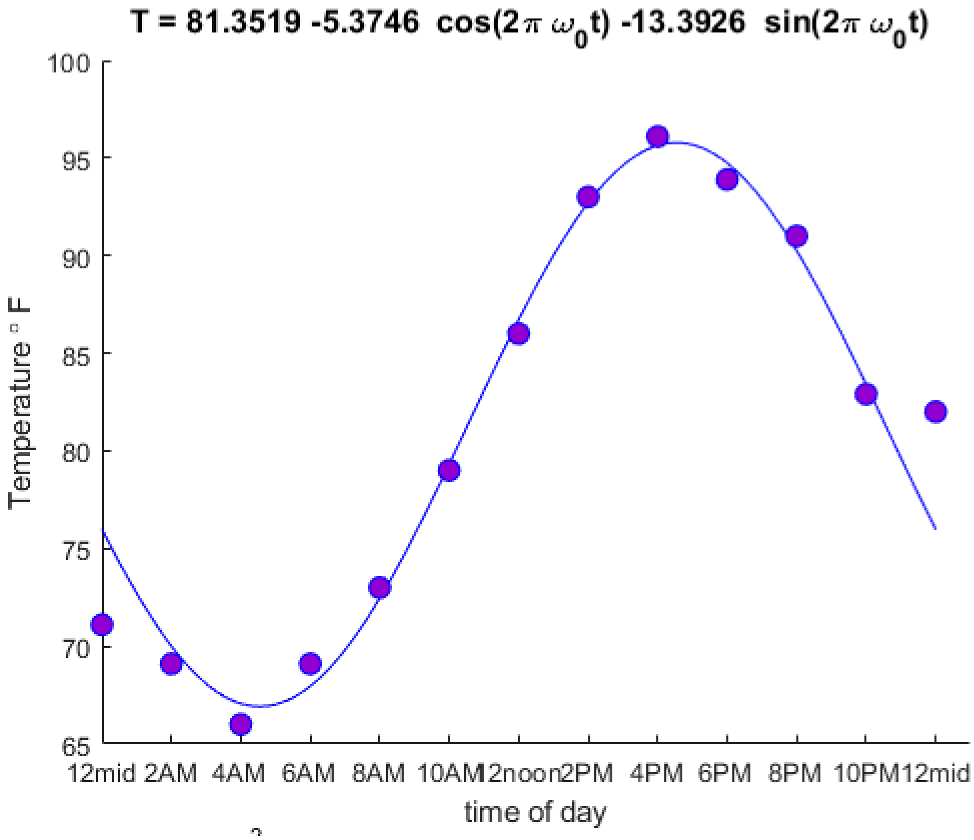
```
title('y = \alpha_4 x e^{(\beta_4 x)}')
```



HW9_2: Sinusoidal model for daily temperature

```
clear all;clc;close all

%repeated from last HW
t=[0:2:24]';
T=[71.1 69.1 66.0 69.1 73.0 79.0 86.0 93.0 96.1 93.9 91.0 82.9 82.0]';
Time_period=24;  % Time period
w0=2*pi/Time_period; % Angular frequency

%T = c1 + c2 cos(w0 t) + c3 sin(w0 t)
Z=[ones(size(t)) cos(w0*t) sin(w0*t)];
c=Z\T;
y1 = @(t)c(1) + c(2)*cos(w0*t) + c(3)*sin(w0*t);
r2=rsquaredF(y1,t,T);

figure
hold on
fplot(y1,[0 24],'b')
plot(t,T,'bo','MarkerFaceColor',[148/255 0 211/255],'MarkerSize',8)
xlabel('time of day')
ylabel('Temperature \circ F')
set(gca,'XTick',(0:2:24),'XTickLabel',
{'12mid','2AM','4AM','6AM','8AM','10AM','12noon','2PM','4PM','6PM','8PM','10PM'})
```

```
str=[sprintf('T = %5.4f %+5.4f ',c(1),c(2)) ' cos(2\pi \omega_0t) '
 sprintf('%+5.4f ',c(3)) ' sin(2\pi \omega_0t) '];
title(str,'FontSize',12)
text(4,60,'r^2=0.9086')
```



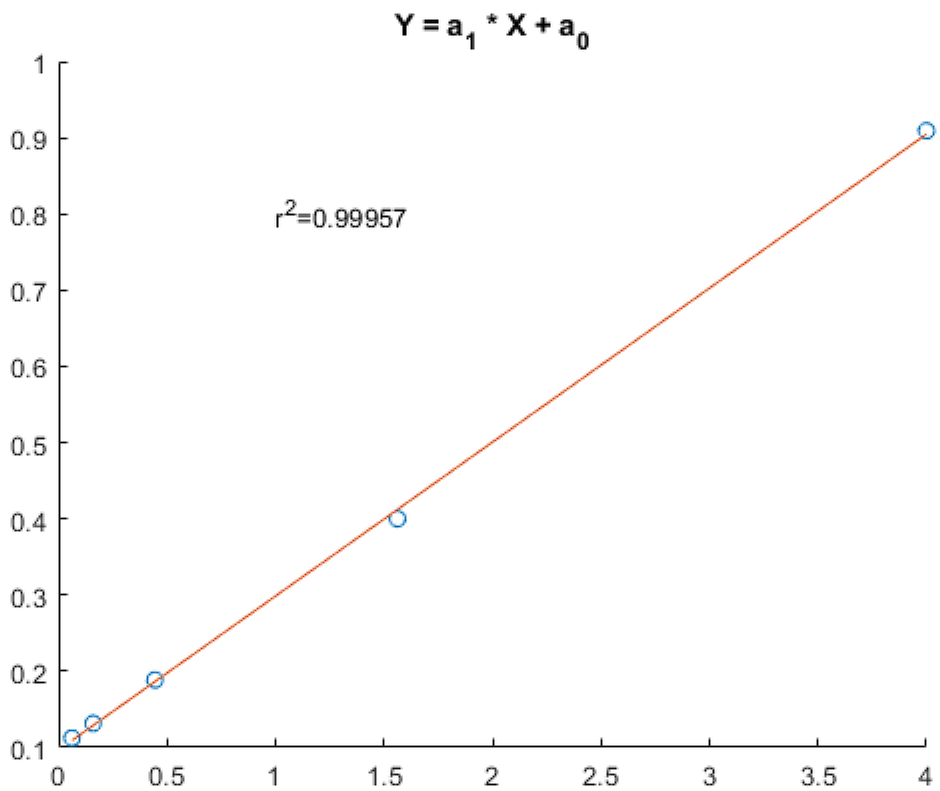

```
A0=c(1);A1=c(2);B1=c(3);
%compute mean
fprintf('The mean temperature on September 15: %4.2f F\n',A0)

%to get the rest of the info, let's put in a different form, eq 16.2
%f(t) = A0 + C1 cos(w0t + phi)
C1 = sqrt(A1^2 + B1^2); % amplitude
%phi should be positive, a leading phase angle, begins a new cycle
%BEFORE cos(w0t). See figure 16.3b
phi = atan(-B1/A1)+pi;  % leading phase angle

%temps
maxtemp = A0 + C1; % maximum temperature
mintemp = A0 - C1; % minimum temperature

%max occurs at cos(w0t + phi) = 1 => w0t + phi = 0, solve for t
maxtime = (-phi/w0) + 24; %positive times only
%min occurs cos(w0t + phi) = -1 => w0t + phi = pi, solve for t
mintime = (-phi + pi)/w0 ;

%print the results
```

```
fprintf('Minimum temp is %4.2f F @ %4.2f hours past midnight
\n',mintemp,mintime)
fprintf('Maximum temp is %4.2f F @ %4.2f hours past midnight
\n',maxtemp,maxtime)
```

```
The mean temperature on September 15: 81.35 F
Minimum temp is 66.92 F @ 4.54 hours past midnight
Maximum temp is 95.78 F @ 16.54 hours past midnight
```

HW9_3: Linearizing a nonlinear function (textbook 14.14)

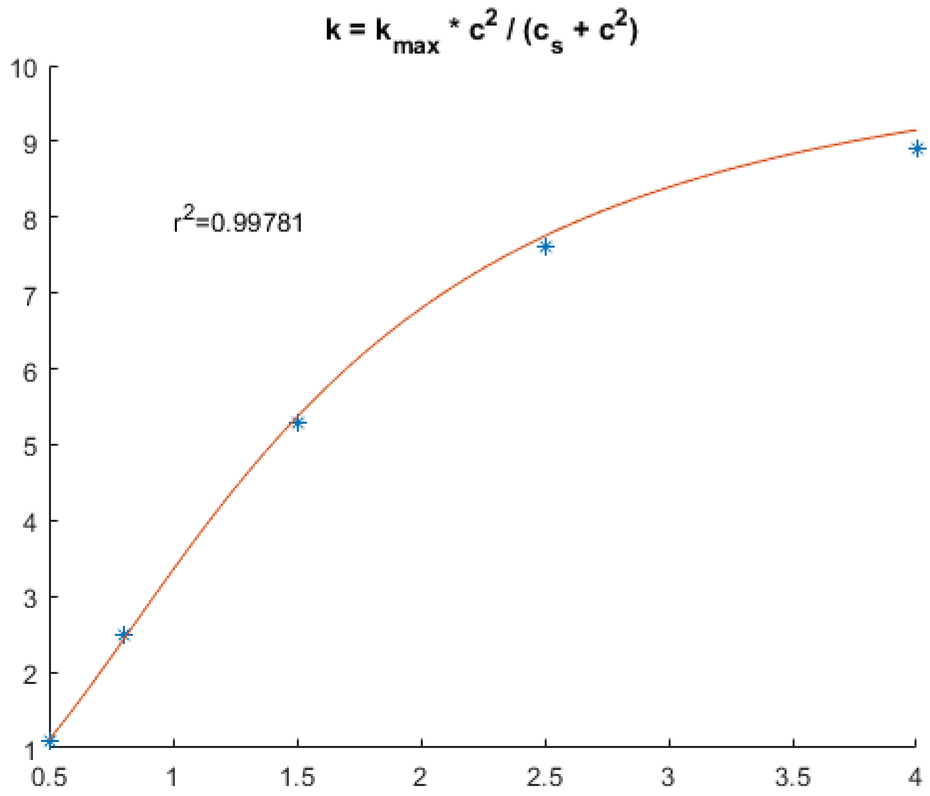
```
clear;clc;close all
%k = k_max * c^1/(c_s+c^2)
%linearizes to--> 1/k= (c_s/k_max)*(1/c^2) + 1/k_max
%this is a line with slope (c_s/k_max) & intercept (1/k_max)

%original data
c=[0.5  0.8  1.5  2.5  4];
k=[1.1  2.5  5.3  7.6  8.9];

%transformed data
X = 1./c.^2;
Y = 1./k;

%the coefficients (linear fit)
a = polyfit(X,Y,1);
a1=a(1);
a0=a(2);

%linearized model plot
flin=@(x)a1*x+a0; % linearized model
r_lin=rsquaredF(flin,X,Y);
hold on
plot(X,Y,'o');
fplot(flin,[min(X) max(X)])
text(1,0.8, ['r^2=' num2str(r_lin)])
title('Y = a_1 * X + a_0')
hold off
```




```
%transform the coefs back
k_max=1/a0;
c_s=a1/a0;
```

saturation growth model plot

```
f_sat=@(c)k_max * c.^2./(c_s+c.^2); % original model
r_sat=rsquaredF(f_sat,c,k);
figure
hold on
plot(c,k,'*');
fplot(f_sat,[min(c) max(c)])
text(1,8, ['r^2=' num2str(r_sat)])
title('k = k_{max} * c^2 / (c_s + c^2)')
```



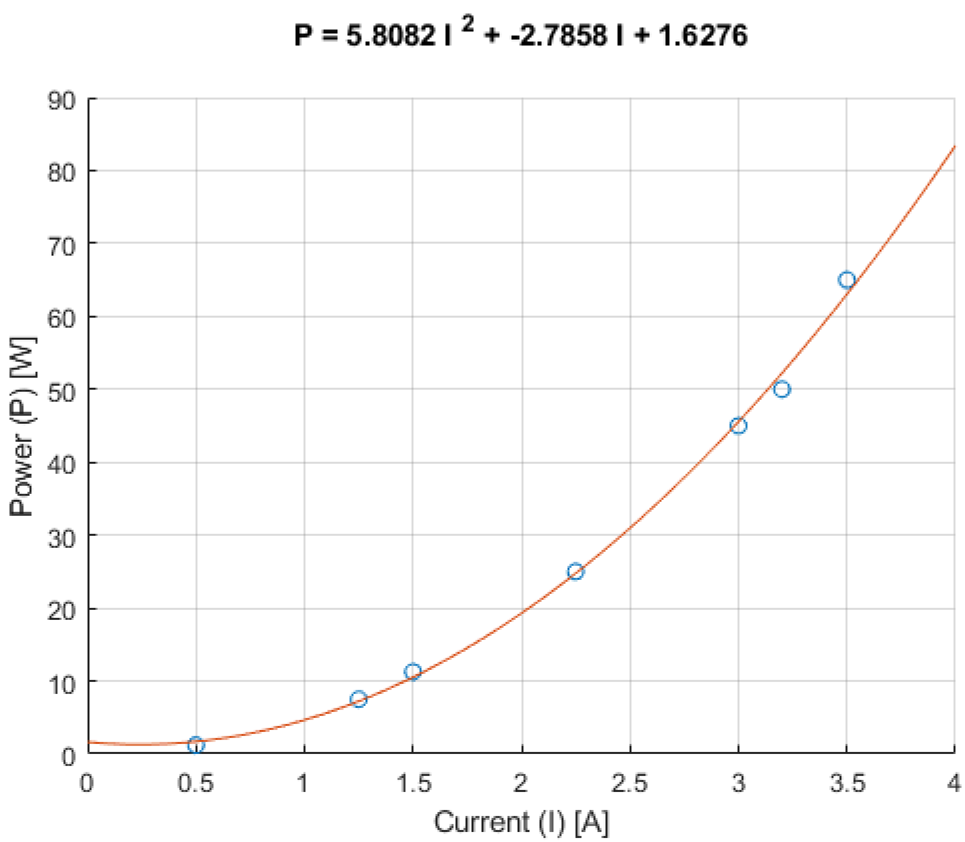
HW9_4 Joule effect

```
clear all;clc;close all
%data
current=[0.5 1.25 1.5 2.25 3 3.2 3.5];
power=[1.2 7.5 11.25 25 45 50 65];

%linear
a_lin=polyfit(current, power,1);
f_lin=@(x) a_lin(1)*x+a_lin(2);
r_lin=rsquaredF(f_lin,current,power);

%parabola
a_para=polyfit(current, power,2);
f_para=@(x) a_para(1)*x.^2+a_para(2)*x+a_para(3);
r_para=rsquaredF(f_para,current,power);

%power
%linearized data:
X=log(current); Y=log(power);
a_pow_lin=polyfit(X, Y,1);
a1=a_pow_lin(1); a0=a_pow_lin(2);
alpha0=exp(a0); alpha1=a1;
f_pow=@(x) alpha0*x.^alpha1;
r_pow=rsquaredF(f_pow,current,power);
```

```
fprintf('The r^2 for the linear model = %.5f\n', r_lin )
fprintf('The r^2 for the parabolic model = %.5f\n', r_para )
fprintf('The r^2 for the power model = %.5f\n', r_pow )

The r^2 for the linear model = 0.95210
The r^2 for the parabolic model = 0.99722
The r^2 for the power model = 0.99661
```

The parabolic model shows the best fit

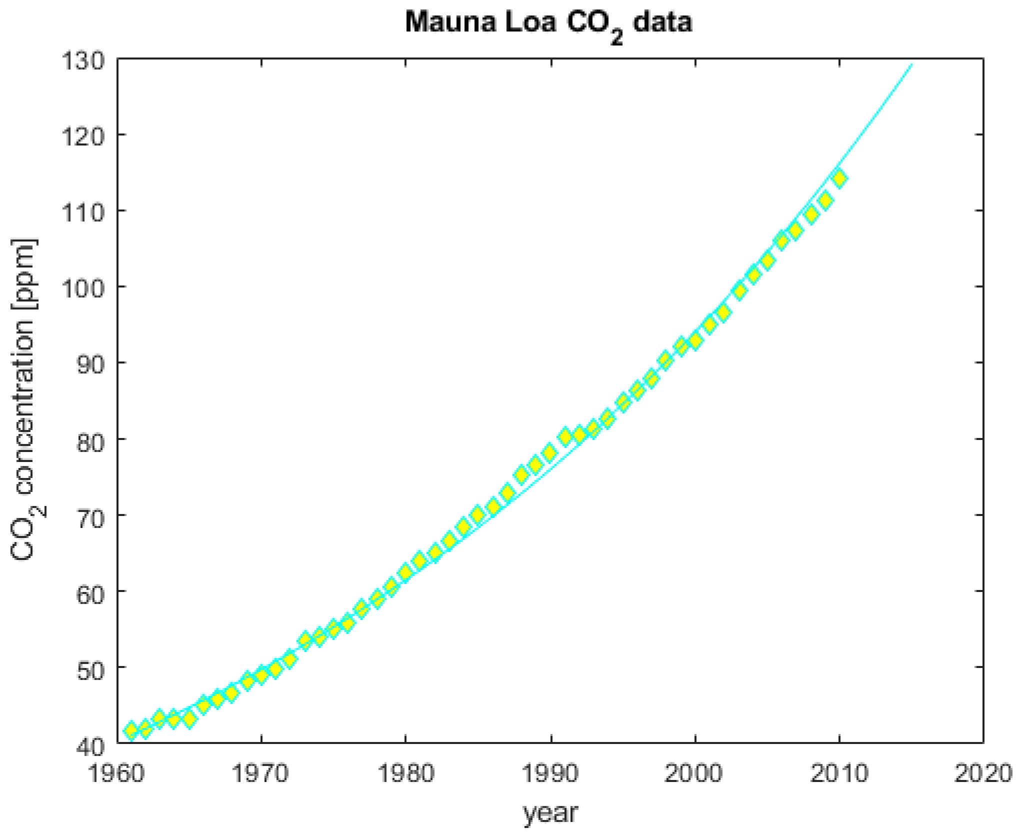
```
fprintf('The predicted power at 4A current = %.4f W\n',  f_para(4))

The predicted power at 4A current = 83.4160 W

%plot
hold on
plot(current,power,'o')
fplot(f_para,[0 4])
title('Joule''s First Law')
xlabel('Current (I) [A]')
ylabel('Power (P) [W]')
title(sprintf('P = %.4f I ^2 + %.4f I + %.4f\n',a_para))
grid on
```



HW9_5: exponential model for concentration of CO2

```
clear all;clc;close all
%load the data
load CO2.txt;
year=[1961:2010]';

%do the curve fit to transformed data
a=polyfit(year,log(CO2),1);
%transform coefs
al1=a(1); al0=exp(a(2));
%model:
f=@(y)al0*exp(al1*y);

plot(year,CO2,'cd','markerfacecolor','y');
hold
fplot(f,[1961 2015],'c')
xlabel('year')
ylabel('CO_2 concentration [ppm]')
title('Mauna Loa CO_2 data')
```

```
Current plot held
```




```
%actual May 15, 2015
```

```
Actual = 404.1 - 279;
Predicted = f(2015);
fprintf('The predicted value of CO2 on May 15, 2015 = %.2f ppm
\n',Predicted)
fprintf('The actual value of CO2 on May 15, 2015 = %.2f ppm\n',Actual)
```

```
The predicted value of CO2 on May 15, 2015 = 129.14 ppm
The actual value of CO2 on May 15, 2015 = 125.10 ppm
```

Published with MATLAB® R2020a