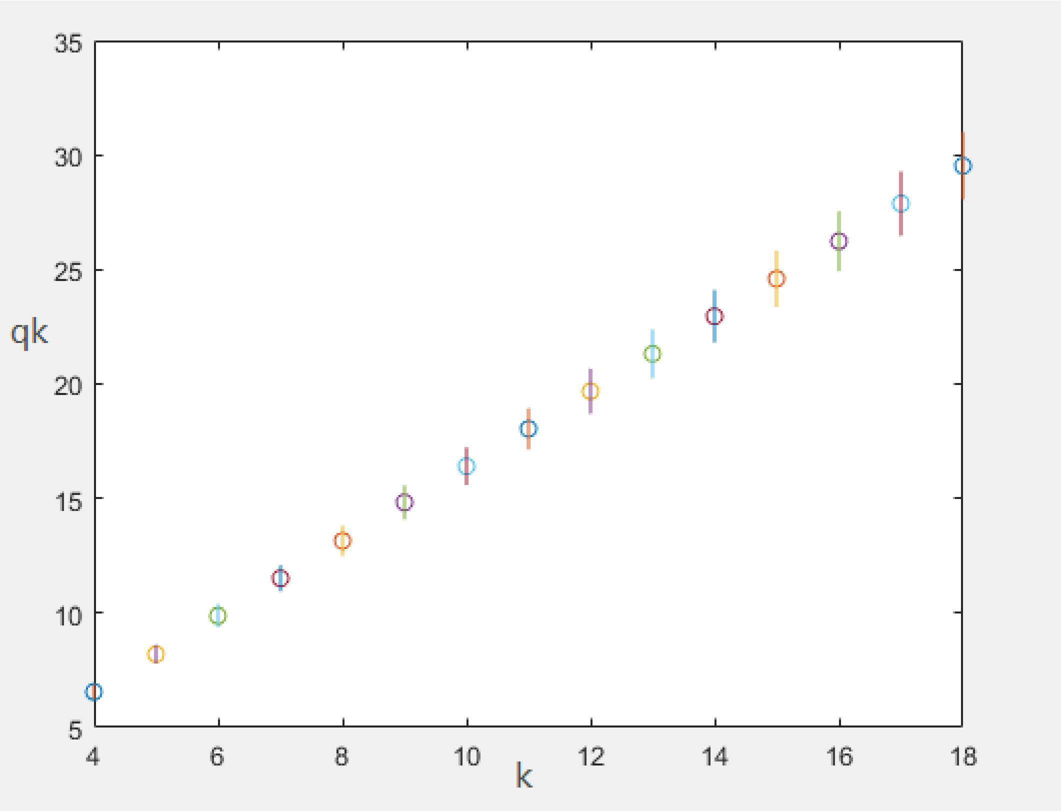
第三次作业

李阳 5150272910019 黄弘毅 515072910038 王潇卫 515072910032

1

(1)

Firstly we plot the data, with uncertainties explicitly showing on the graph.



And we apply linear regression method to this problem, using Round-off error susceptible version.

Code:

```
X=4:1:18;
Y=
[6.558,8.206,9.880,11.50,13.14,14.82,16.40,18.04,19.68,21.32,22.96,24.60,26.24,27.88,29.52];
U=0.05*Y;
for i=1:15
    plot(X(i),Y(i),'-o',[X(i),X(i)],[Y(i)-U(i),Y(i)+U(i)]);
    hold on;
end
S=sum(1./U./U);
Sx=sum(X./U./U);
Sy=sum(Y./U./U);
```

```
t=(X-Sx/S)./U;
Stt=sum(t.*t);
a1=sum(t.*Y./U)/Stt;
a0=(Sy-Sx*a1)/S;
u2a0=(1+Sx*Sx/S/Stt)/S;
u2a1=1/Stt;
Cov=-Sx/S/Stt;
r=Cov/sqrt(u2a0)/sqrt(u2a1);
Y1=a0+a1*X;
Chi2=sum((Y-Y1).*(Y-Y1)./U./U);
[a0,u2a0,a1,u2a1,Chi2,Cov,r]
```

Result:

$a_0 = 0.01175, \sigma_{a0} = 0.3838, a_1 = 1.640, \sigma_{a1} = 0.04749$
$X^2 = 0.01088, Cov(a_0, a_1) = -0.01631, r = -0.8949$

We get $q_k = 0.01175(\pm 0.3838) + 1.640(\pm 0.047)k$ The intercept has rather big uncertainty in the experiment $0.01175(\pm 0.3838)$,while the slope is relatively accurate$1.640(\pm 0.047)$. Here X^2 is about 0.01088, which is small. And coefficient of correlation is -0.8949, showing a fairly strong anticorrelation of error between a_0 and a_1.

2

（1）

$Y = y, X = \frac{1}{x}$ then, $Y = AX + B$

(2)

$Y = \frac{1}{y}, X = x, A = \frac{1}{D}, B = CA$ then, $Y = AX + B$

(3)

$Y = \frac{1}{y}, X = x$ then,$Y = AX + B$

(4)

$\frac{1}{y} = \frac{A}{x} + B$ $Y = \frac{1}{y}, X = \frac{1}{x}$ then,$Y = AX + B$

(5)

$Y = y, X = \ln x$ then,$Y = AX + B$

(6)

$\ln y = \ln C + A \ln x$ $Y = \ln y, X = \ln x, B = \ln C$ then,$Y = AX + B$

(7)

$y^{-0.5}=Ax+B\,Y=y^{-0.5},X=x$ then,$Y=AX+B$

(8)

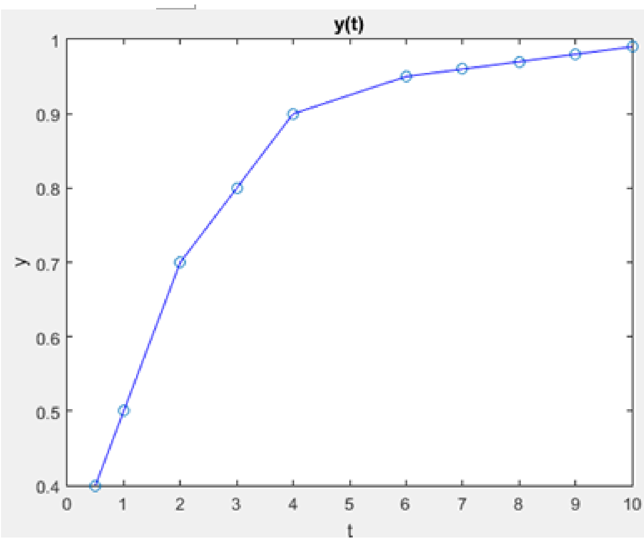
$\ln y=\ln C+\ln x-Dx\ln\frac{y}{x}=\ln C-Dx\,Y=\ln\frac{y}{x},X=x,A=-D,B=\ln C$ then,$Y=AX+B$

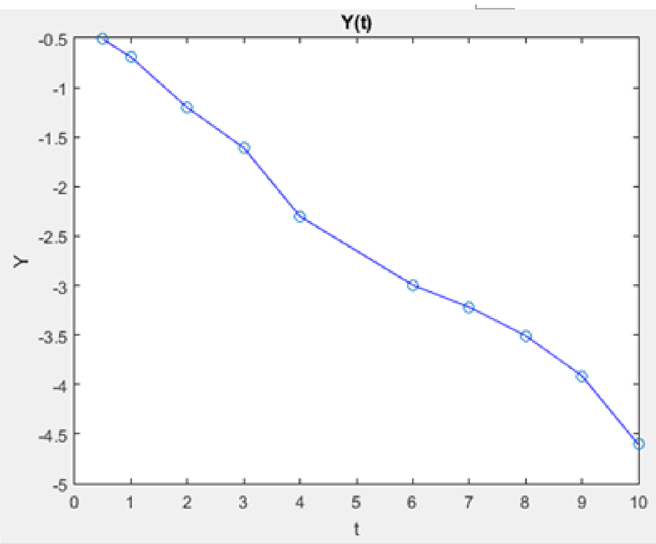
3

$y=1+Ae^{-\alpha t},A<0,\alpha>0\,\ln(1-y)=\ln(-A)-\alpha t\,Y=\ln(1-y),B=\ln(-A)$ then,$Y=B-\alpha t$

t_i | y_i | Y_i

- | :-: | -:
  0.5 | 0.4 | -0.5108
  1 | 0.5 | -0.6931
  2 | 0.7 | -1.2040
  3 | 0.8 | -1.6094
  4 | 0.9 | -2.3026
  6 | 0.95| -2.9957
  7 | 0.96| -3.2189
  8 | 0.97| -3.5066
  9 | 0.98| -3.9120
  10| 0.99| -4.6052

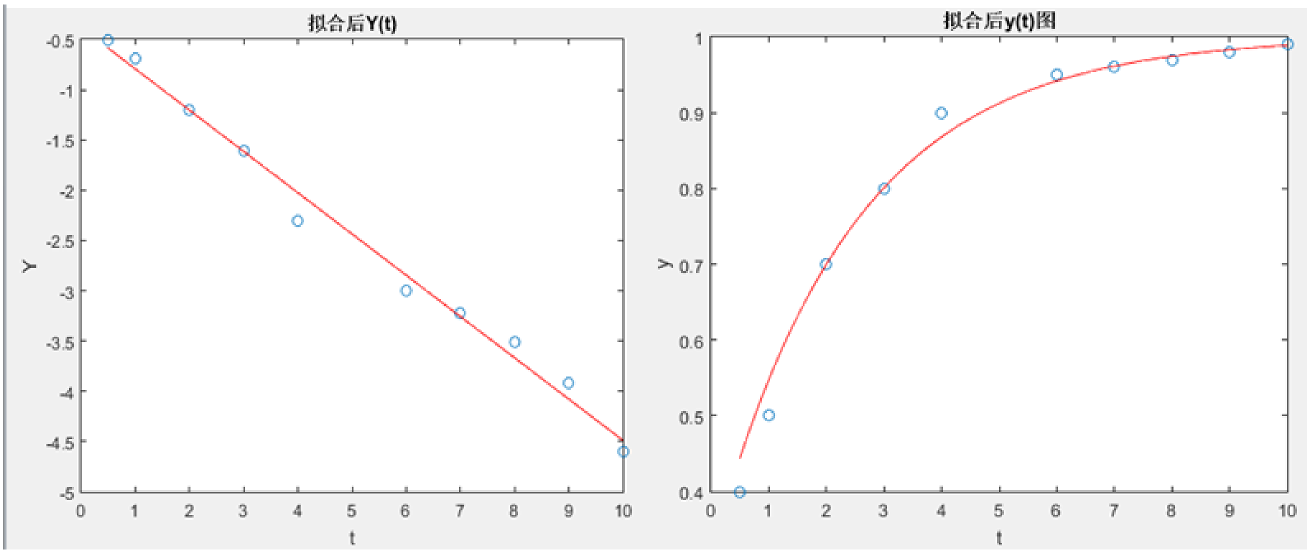




$S_t=\sum_i^{10}t_i=50.5\;S_Y=\sum_i^{10}Y_i=-24.5583\;S_tY=\sum_i^{10}t_iY_i=-167.2141\;S_tt=\sum_i^{10}t_it_i=360.25$

So, $\alpha=-\frac{10S_{tY}-S_tS_Y}{10S_{tt}-S_t^2}=0.410497\;B=\frac{S_{tt}S_Y-S_tS_{tY}}{10S_{tt}-S_t^2}=-0.382825\;A=-e^B=-0.681932$

$y=1-0.681932e^{-0.410497t}$



code:

```
%计算物理 第三次作业 第3题
%王潇卫 515072910032
clear all
clc

t = [0.5,1,2,3,4,6,7,8,9,10];
y = [0.4,0.5,0.7,0.8,0.9,0.95,0.96,0.97,0.98,0.99];
Y = log(1-y);
%关系式变形为 Y = B - at

%% 作图
figure(1) % y、t图
plot(t,y,'o',t,y,'b')
xlabel('t')
ylabel('y')
title('y(t)')
hold on

figure(2) % Y 、t图
plot(t,Y,'o',t,Y,'b')
xlabel('t')
ylabel('Y')
title('Y(t)')
hold on

%% 拟合
S_t = sum(t);
S_Y = sum(Y);
tY = t.*Y;
S_tY = sum(tY);
S_tt = sum(t.^2);

alpha = (S_t*S_Y-10*S_tY)/(10*S_tt-(S_t)^2);
```

```
B = (S_tt*S_Y-S_t*S_tY)/(10*S_tt-(S_t)^2);
A = -exp(B);

fprintf(1,'Fitting the curve gets Y = %f - %f t \n',B,alpha);
fprintf(1,'Equals to y = 1 %f e^(-%f t) \n',A,alpha);

%% 作图
t1 = 0.5:0.1:10;
y1 = 1 - 0.681932*exp(-0.410497*t1);
Y1 = -0.382825 - 0.410497*t1;

figure(3) % 拟合后y、t图
plot(t,y,'o',t1,y1,'r')
xlabel('t')
ylabel('y')
title('拟合后y(t)图')
hold on

figure(4) % 拟合后Y 、t图
plot(t,Y,'o',t1,Y1,'r')
xlabel('t')
ylabel('Y')
title('拟合后Y(t)')
hold on
```

4

We use the method of linearization of nonlinear data to fit the dataset. (i)For the power fitting function, k=a T^b, Definite Y=ln(k),X=ln(T), then we get Y=$a_1 X + a_0$, and we have the relation: b=a_1,a=$exp(a_0)$; (ii)For the exponential fitting function, k=ae^{bT}, Definite Y=ln(k),X=T, then we get Y=$a_1 X + a_0$, and we have the relation: b=a_1,a=$exp(a_0)$; (iii)For the saturation fitting function, k=$\frac{T}{aT+b}$, Definite Y=$\frac{1}{k}$,$X = \frac{1}{T}$ then we get Y=$a_1 X + a_0$, and we have the relation:b=a_1,a=a_0;

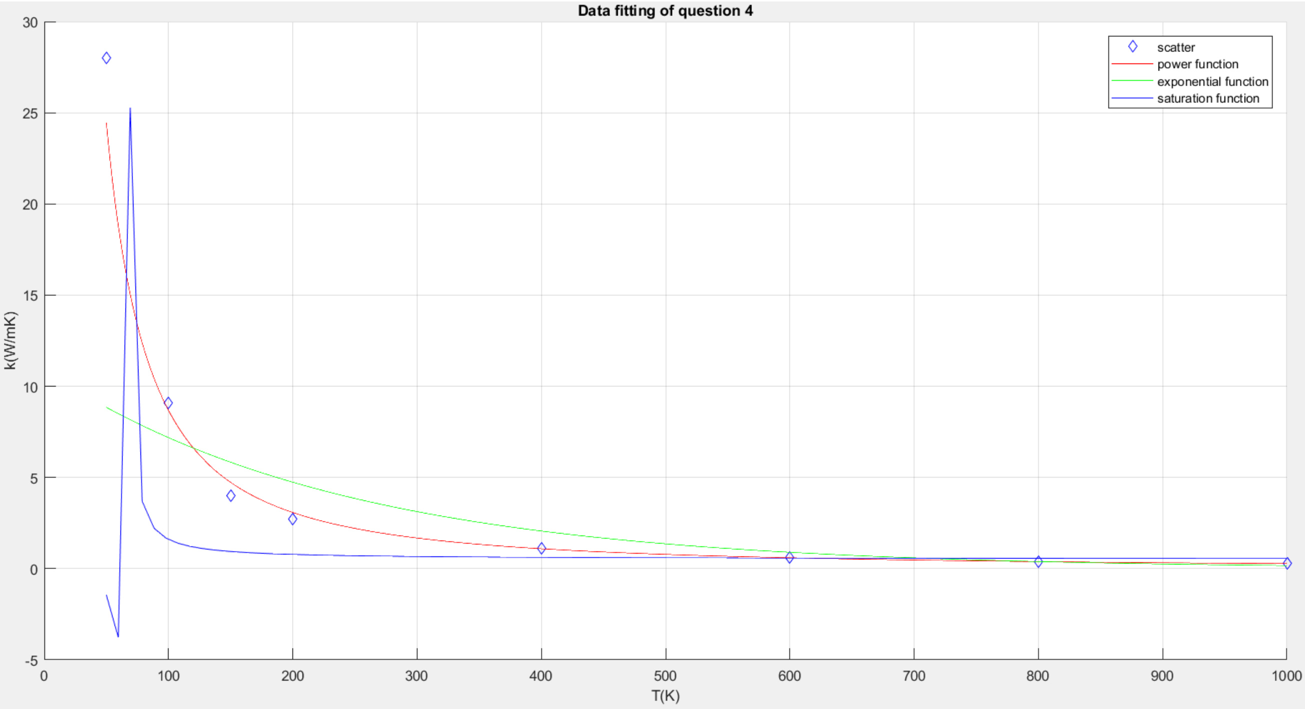
Code:Q4.m

```
%this script is used to solve Question 4.
clear all; clc;
%% the dataset:
n=8;
T=[50,100,150,200,400,600,800,1000];
k=[28,9.1,4.0,2.7,1.1,0.6,0.4,0.3];
%% Use the power function to fit dataset:
y1=log(k);x1=log(T);Sx1=0;Sy1=0;Sxy1=0;Sxx1=0;
for i=1:8
    Sx1=Sx1+x1(i);Sy1=Sy1+y1(i);
    Sxy1=Sxy1+x1(i)*y1(i);Sxx1=Sxx1+x1(i)*x1(i);
end
% To determine the coeffecients:
a1=(n*Sxy1-Sx1*Sy1)/(n*Sxx1-Sx1*Sx1);
```

```
a0=(Sxx1*Sy1-Sx1*Sxy1)/(n*Sxx1-Sx1*Sx1);
% Start to plot:
figure(1);
scatter(T,k,'db');hold on;
b=a1;a=exp(a0);
x=linspace(50,1000,100000);y=a.*(x.^b);
plot(x,y,'r-');% the red solidline to represent.
hold on;
%% Use the exponential function to fit dataset:
y2=log(k);x2=T;Sx2=0;Sy2=0;Sxy2=0;Sxx2=0;
for i=1:8
    Sx2=Sx2+x2(i);Sy2=Sy2+y2(i);
    Sxy2=Sxy2+x2(i)*y2(i);Sxx2=Sxx2+x2(i)*x2(i);
end
a1=(n*Sxy2-Sx2*Sy2)/(n*Sxx2-Sx2*Sx2);
a0=(Sxx2*Sy2-Sx2*Sxy2)/(n*Sxx2-Sx2*Sx2);
b=a1;a=exp(a0);
x=linspace(50,1000,100000);y=a.*(exp(b.*x));
plot(x,y,'g-');% the green solidline to represent.
hold on;
%% Use the saturation function to fit dataset:
y3=1./k;x3=1./T;Sx3=0;Sy3=0;Sxy3=0;Sxx3=0;
for i=1:8
    Sx3=Sx3+x3(i);Sy3=Sy3+y3(i);
    Sxy3=Sxy3+x3(i)*y3(i);Sxx3=Sxx3+x3(i)*x3(i);
end
% To determine the coeffecients:
a1=(n*Sxy3-Sx3*Sy3)/(n*Sxx3-Sx3*Sx3);
a0=(Sxx3*Sy3-Sx3*Sxy3)/(n*Sxx3-Sx3*Sx3);
b=a1;a=a0;
x=linspace(50,1000,100000);y=a.*(x.^b);
plot(x,y,'b-')% the blue solidline to represent.
legend('scatter','power function', 'exponential function','saturation function');
grid on;
xlabel('T(K)');
ylabel('k(W/mK)');
title('Data fitting of question 4');
```

Result:

The difference between different fitting is shown as the following picture:



(i)For power function, we get $a_0 = 9.0409, a_1 = -1.4940$; We check it by using matlab built-in cftool, and we get the same result. R-square=0.9959; (ii)For exponential function, we get $a_0 = 2.3879, a_1 = -0.0042$; We check it by using matlab built-in cftool, and we get the same result. R-square=0.8472; (iii)For exponential function, we get $a_0 = 1.9353, a_1 = -131.1678$; We check it by using matlab built-in cftool, and we get the same result. R-square=0.4715; So we can draw a conclusion that power function has a best fit, and the fitting function relation is $k = 8441.6x^{-1.4940}$;

5

Based on our experience, the sales volume y increases with x1's increase and decrease with x2's increase. We need to find a two-variable function. First, we set up the relation of y=$a_0 + a_1x_1 + a_2x_2$.

Now, we use the Least-Square Method to determine parameters a_0, a_1, a_2; $r_i = y_i - a_0 - a_1x_1 - a_2x_2$, $S = \frac{1}{N}\sum_{i=1}^{N} {r_i}^2$; Now $S = S(a_0, a_1, a_2)$, the equations need to be solved are: $\frac{\partial S}{\partial a_i} = 0$, where i=0,1,2; Calculate it we get:

$$\begin{cases} S_y - a_0S - a_1S_{x_1} - a_2S_{x_2} = 0 \\ S_{x_1y} - a_0S_{x_1} - a_1S_{x_1x_1} - a_2S_{x_1x_2} = 0 \\ S_{x_2y} - a_0S_{x_2} - a_1S_{x_1x_2} - a_2S_{x_2x_2} = 0 \end{cases}$$

So we define A=$\begin{Bmatrix} S & S_{x_1} & S_{x_2} \\ S_{x_1} & S_{x_1x_1} & S_{x_1x_2} \\ S_{x_2} & S_{x_1x_2} & S_{x_2x_2} \end{Bmatrix}$ b=$\begin{Bmatrix} S_y \\ S_{x_1y} \end{Bmatrix}$ then a=A\b;

Code Q5.m

```
% this script is used to solve question 5.
clear all;clc;

%% show the dataset and produce the summary of x1,x2,x1*x1,x1*x2,x2*x2,y,x1*y,x2*y;
```

```
x1=[110,130,180,125,150,165,120,145,175,155];
x2=[105,125,85,145,200,160,240,260,290,270];
y=[102,99,115,75,55,95,36,78,82,90];
s1=0;s2=0;s11=0;s12=0;s22=0;sy=0;sy1=0;sy2=0;s=10;
for i=1:10
    s1=s1+x1(i);s2=s2+x2(i);
    s11=s11+x1(i)*x1(i);s12=s12+x1(i)*x2(i);s22=s22+x2(i)*x2(i);
    sy=sy+y(i);sy1=sy1+x1(i)*y(i);sy2=sy2+x2(i)*y(i);
end
%% to solve the linear equations:
A=[s s1 s2;s1 s11 s12;s2 s12 s22];
b=[sy;sy1;sy2];
a=A\b;%a=[a0,a1,a2]
%% calculate the model f and compare with y
f=a(1)+a(2).*x1+a(3).*x2;
rr=(f-y)./y;
rr=rr';
```

Result

Fitting function: $y = 54.4432 + 0.4564x_1 - 0.2029x_2$. Using the matlab built-in function "regress",

```
X=[ones(10,1) x1' x2'];
regress(y', X);
```

We can get the same result.

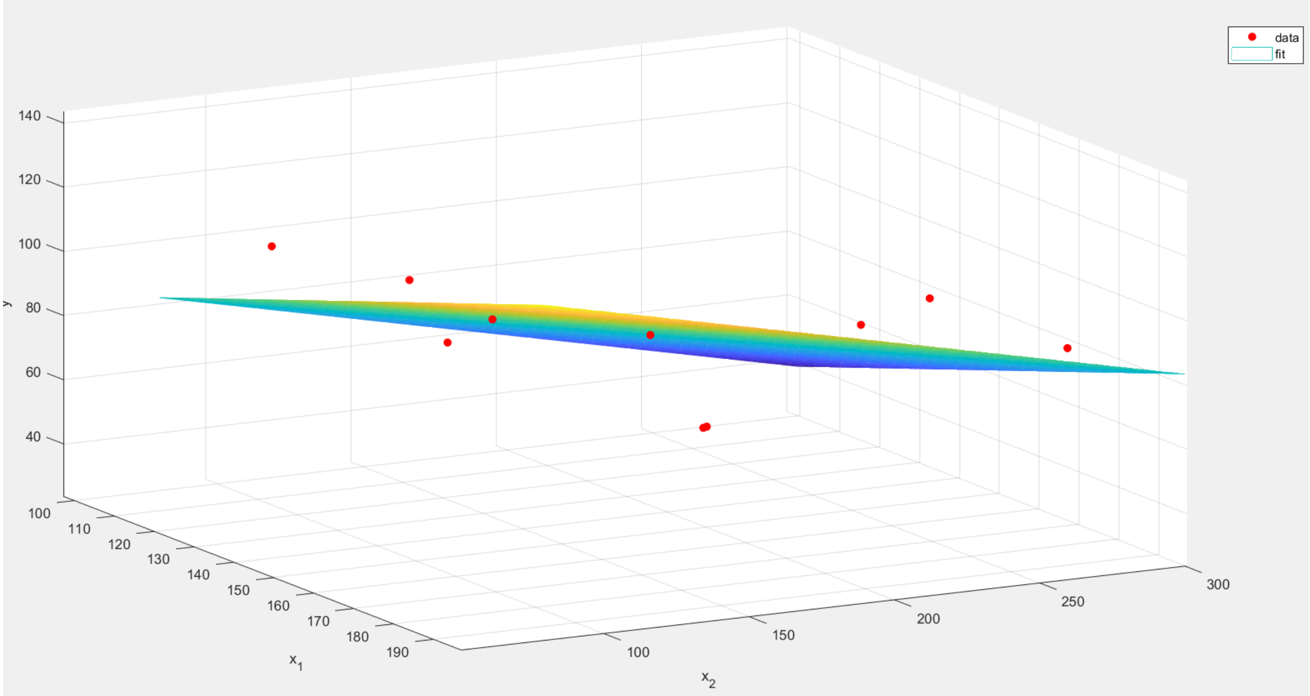
```
>> regress(y', X)

ans =

   54.4432
    0.4564
   -0.2029
```

Now we calculate the relative error of $y_i and f_i$.

\begin{array}{c|lcr} n & \text{y_i} & \text{$f(x_{1i}, x_{2i})$} & \text{relative error} \ \hline 1 & 102 & 83.3393 & 0.18 \ 2 & 99 & 88.4087 & 0.11 \ 3 & 115 & 119.3425 & 0.04\ 4 & 75 & 82.0590 & 0.09\ 5 & 55 & 82.3189 & 0.50\ 6 & 95 & 97.2801 & 0.02\ 7 & 36 & 60.5123 & 0.68\ 8 & 78 & 67.8635 & -0.13\ 9 & 82 & 75.4676 & -0.08\ 10 & 90 & 70.3982 & -0.22\ \end{array} From the table, we can see that the linear relation $y = 54.4432 + 0.4564x_1 - 0.2029x_2$ fits not so well, because the relative error of n=5 and n=7 are 0.50 and 0.68, which is so large. But we are pleased to see that other data points fits well, and $a_1 > 0, a_2 < 0$ are consistent with our experience. And the data and fitting plot is shown as the following picture:

code：

```
scatter3(x1,x2,y,'filled', 'r');
hold on;
x1fit =100:2:200;
x2fit =80:2:300;
[X1FIT,X2FIT] = meshgrid(x1fit,x2fit);
YFIT=a(1)+a(2)*X1FIT+a(3)*X2FIT;
mesh(X1FIT,X2FIT,YFIT)
view(50,40)
xlabel('x_1');
ylabel('x_2');
zlabel('y');
legend('data', 'fit');
```



We can see frow the picture that Data is distributed on both sides of the plane. Acctually, we can fit it better by adding quadratic polynomials. $y = a_0 + a_1x_1 + a_2x_2 + a_3x_1x_2 + a_4x_1^2 + a_5x_2^2$ Now we just use matlab built-in regress:

```
X = [ones(10,1) x1' x2' (x1.*x2)' (x1.^2)' (x2.^2)'];
[b, bint, r, rint, stats] = regress(y', X);
```

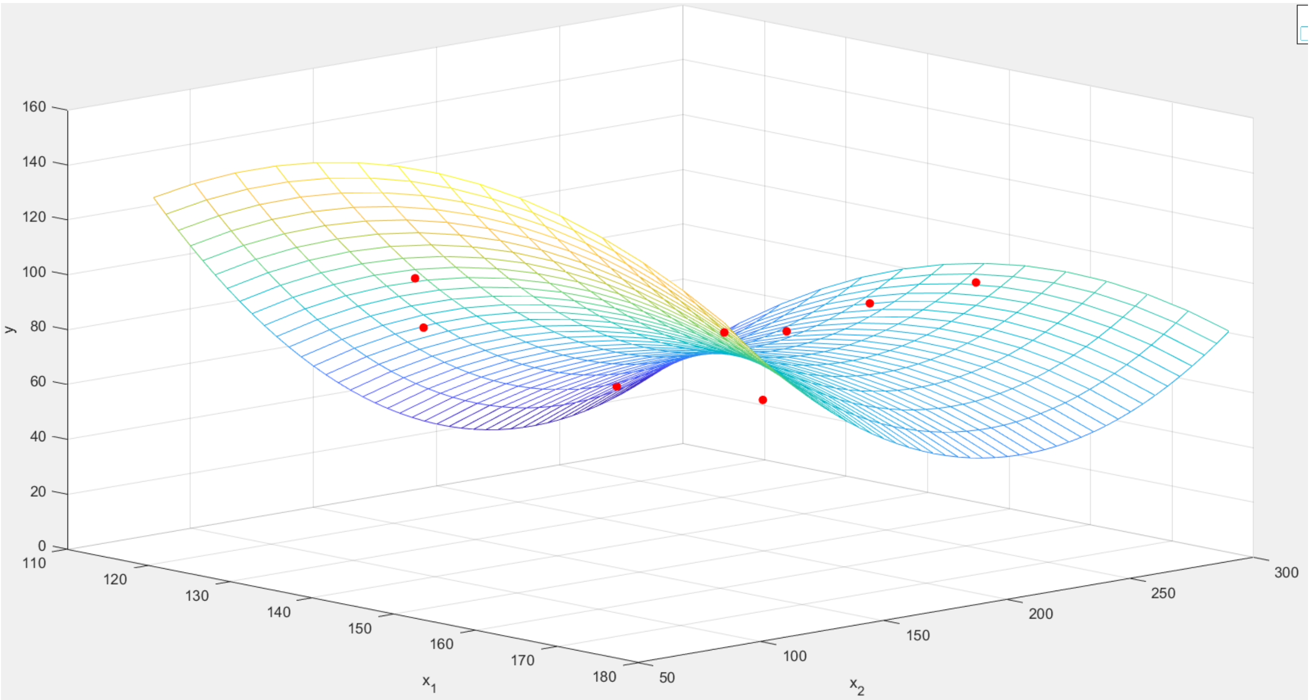
we get the coeffecients:

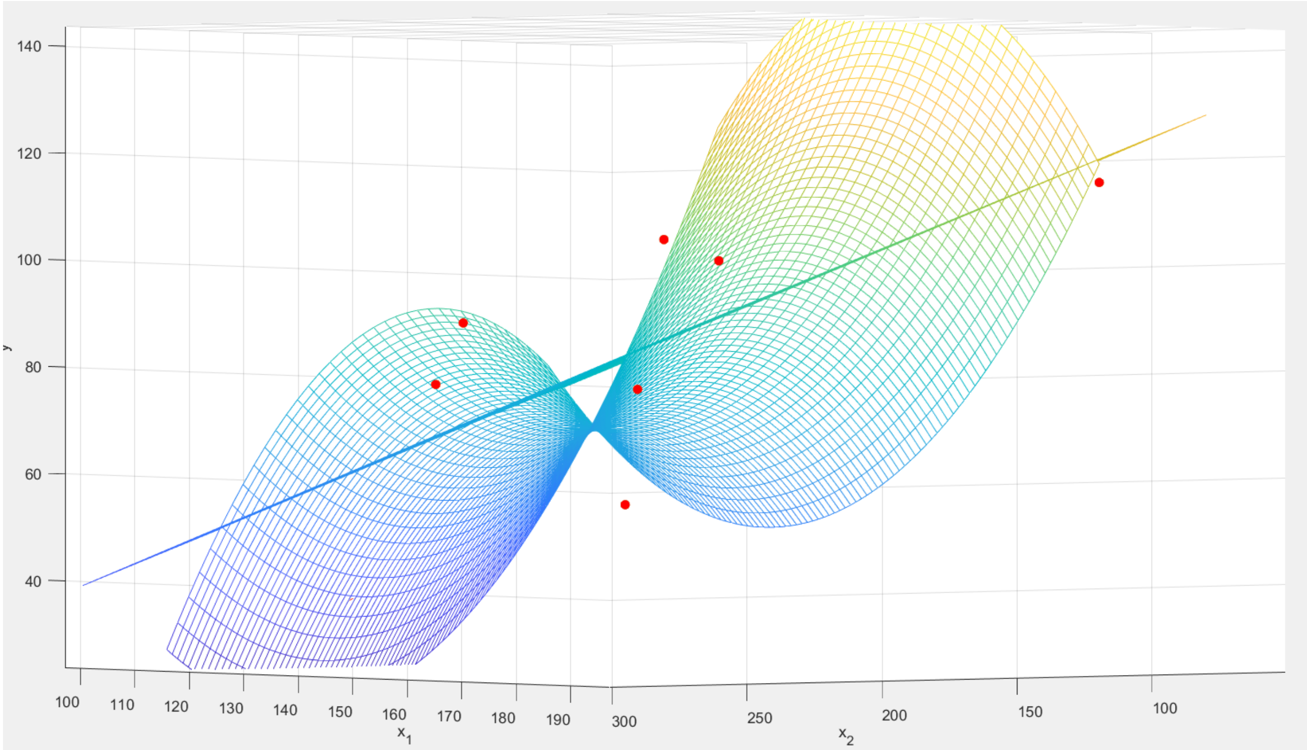
b

6x1 double

	1
1	-191.3181
2	7.0712
3	-2.6281
4	0.0042
5	-0.0258
6	0.0045

and its R-square is 0.857822, which is larger than the before one. the new fitting function: $y = -191.3181 + 7.0712x_1 - 2.6281x_2 + 0.0042x_1x_2 - 0.0258x_1^2 + 0.0045x_2^2$ the new fitting plot and the Comparison plot are as followings:





we can see that , after we adding quadratic polynomials, we get a better fit, which is good enough.