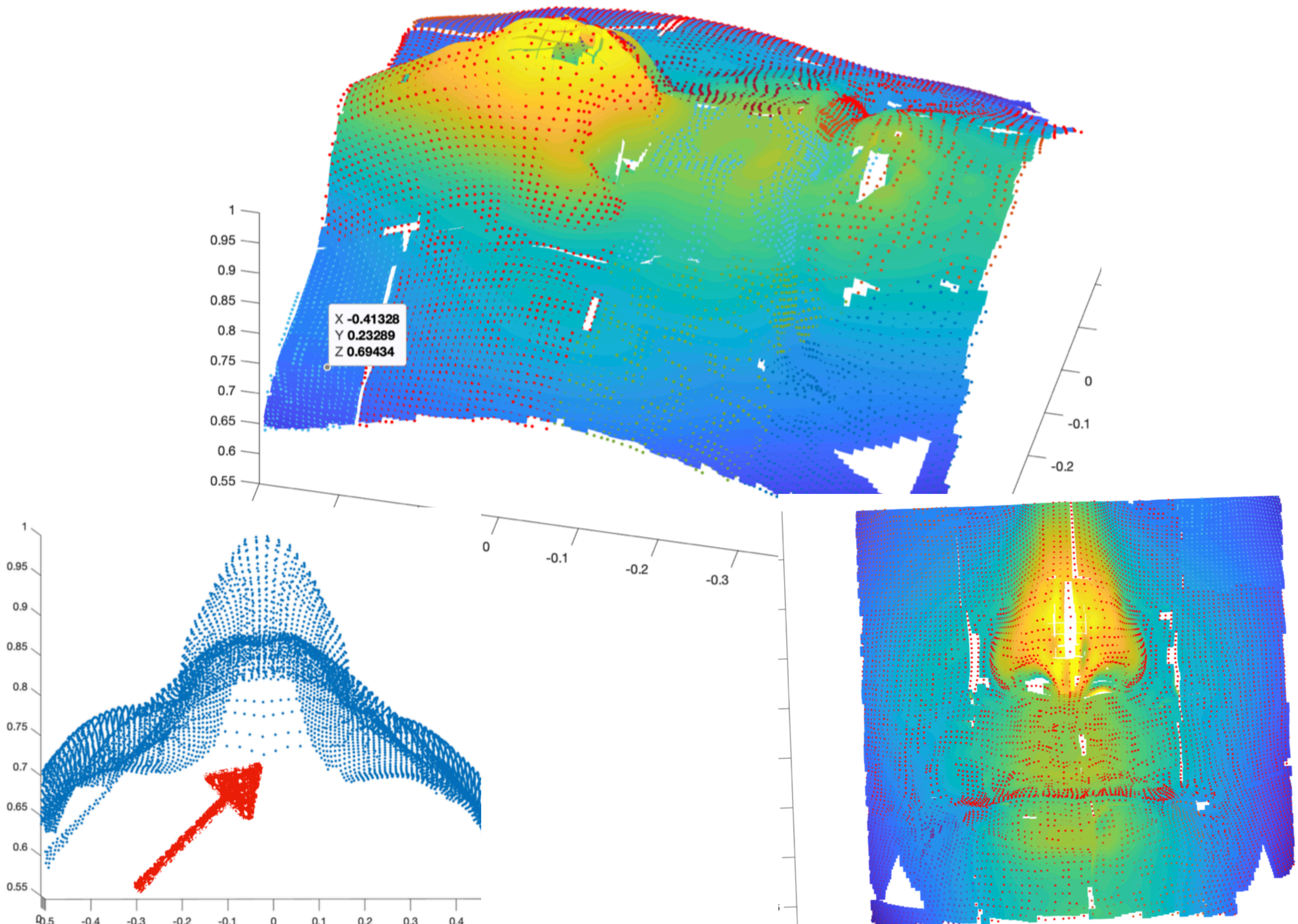
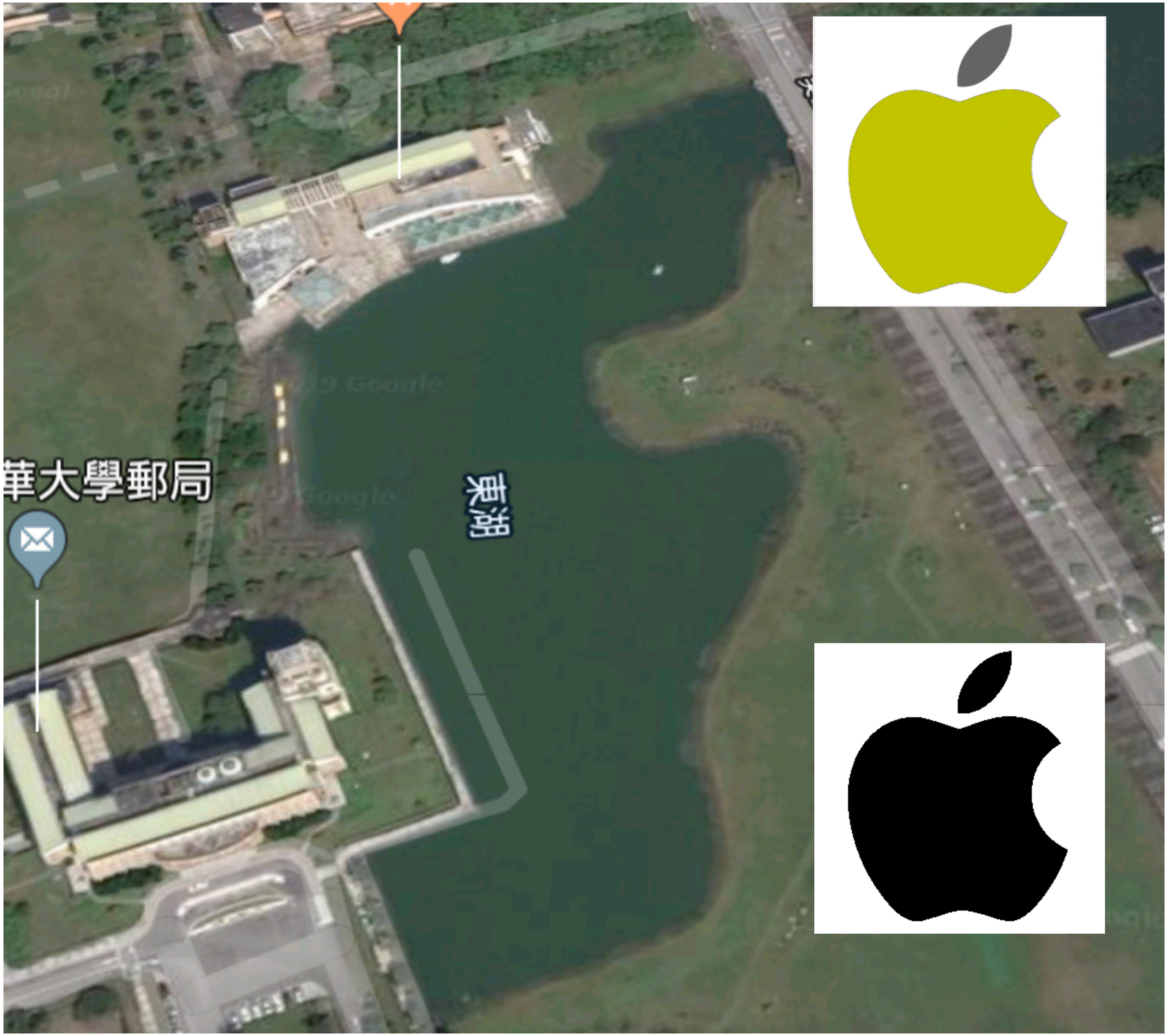
Data fitting



Data fitting

使用衛星圖或地圖，取包含東湖的正方形影像。假設我們已經透過測量方法測得影像正方形的實際邊或以地圖比例尺計算，已給定正方形衛星影像的實際總面積Ａ

1. 請同學使用Matlab計算東湖所佔的像素總數依照東湖總像素佔原正方形的像素比例，推算東湖面積 ____Ａ？
2. 依照比例尺換算為__平方公尺？


華大學郵局
東湖

?? Pixels

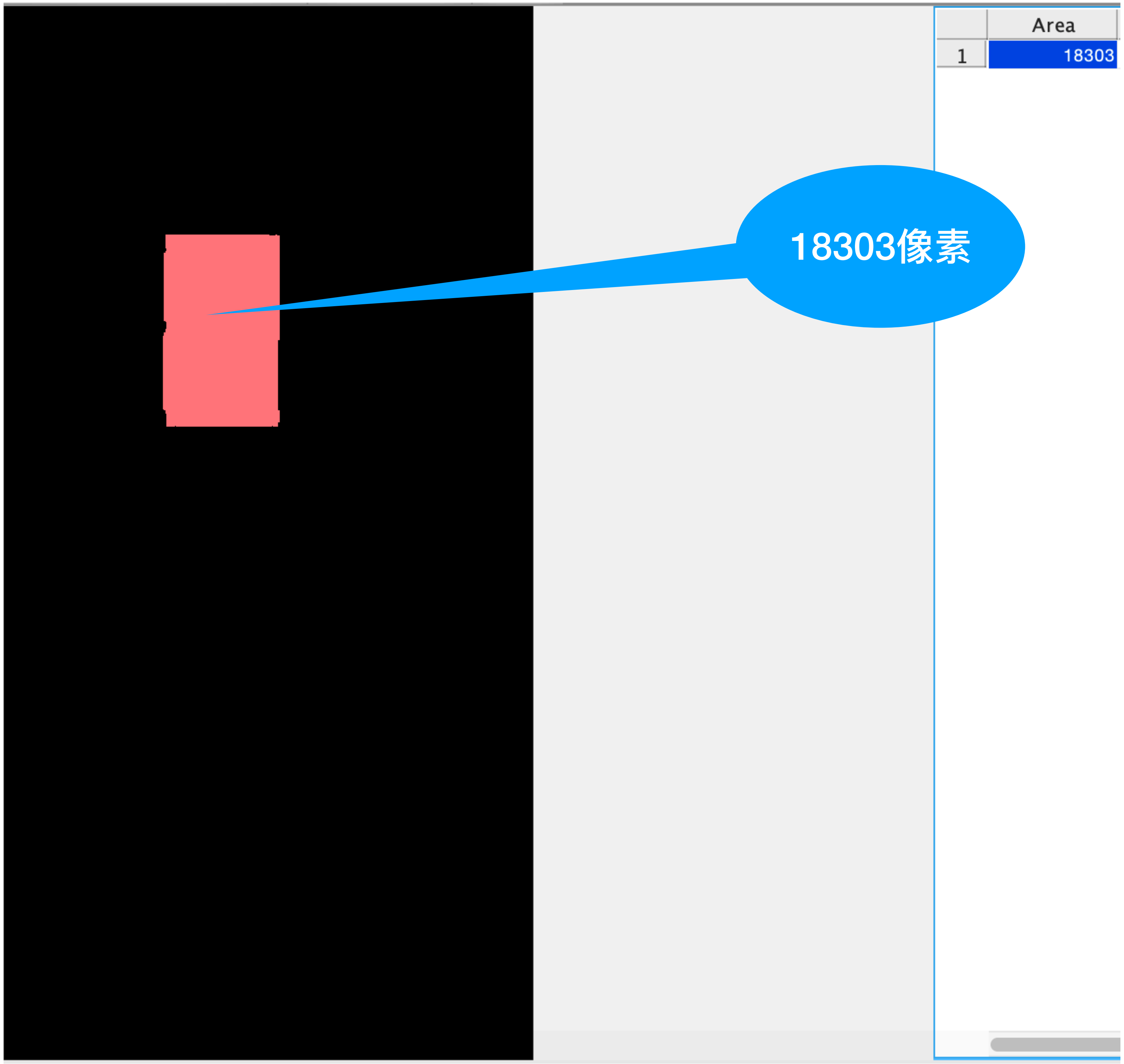
Step 1

73像素

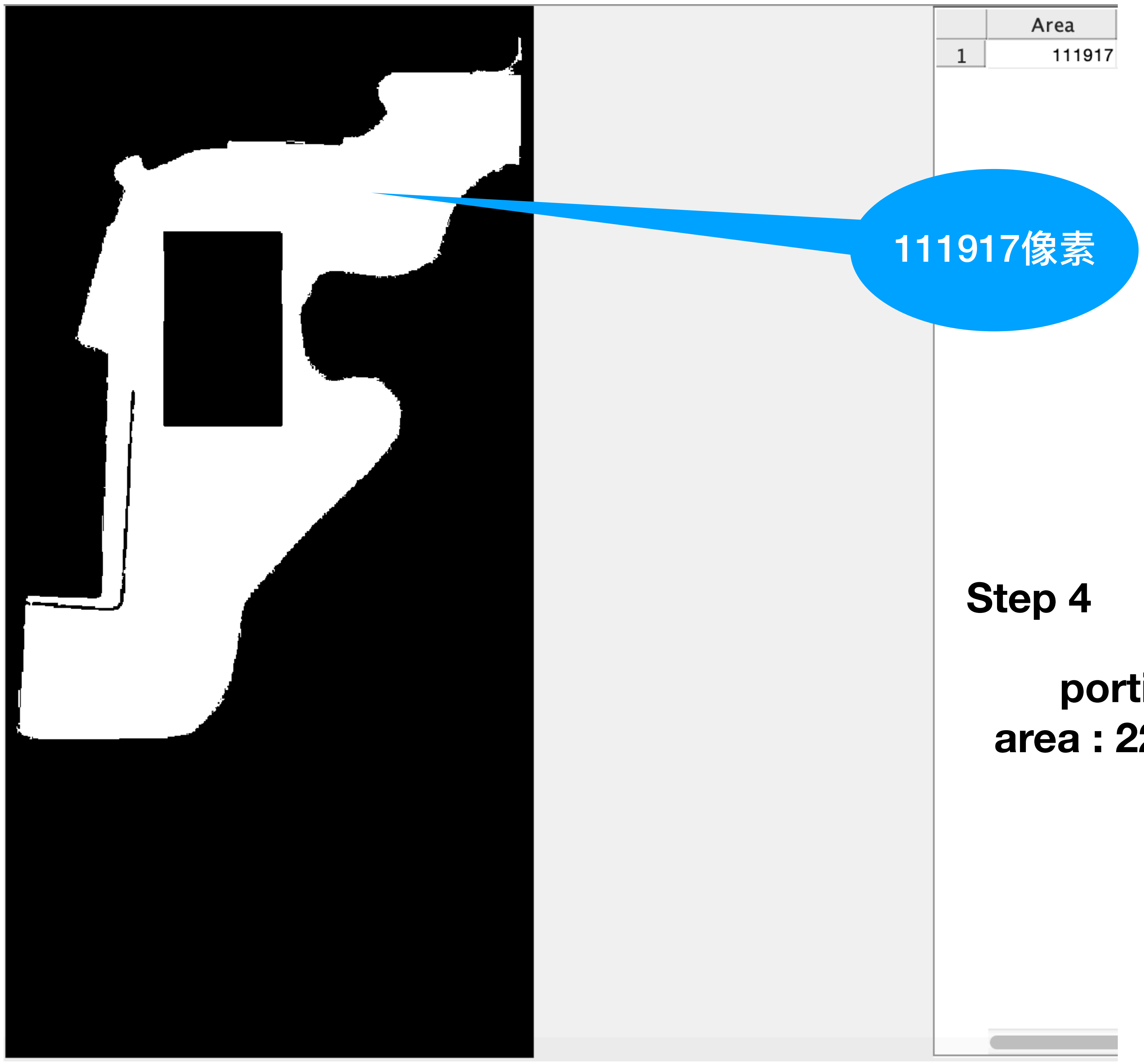
```
im size: 960 x 482
mark size: 74
 length =  1297.2973 feet
 width =   651.3514 feet
 area = 844996.3477 feet-square
 78502.7295 m^2
```

Step 2

Step 3.1

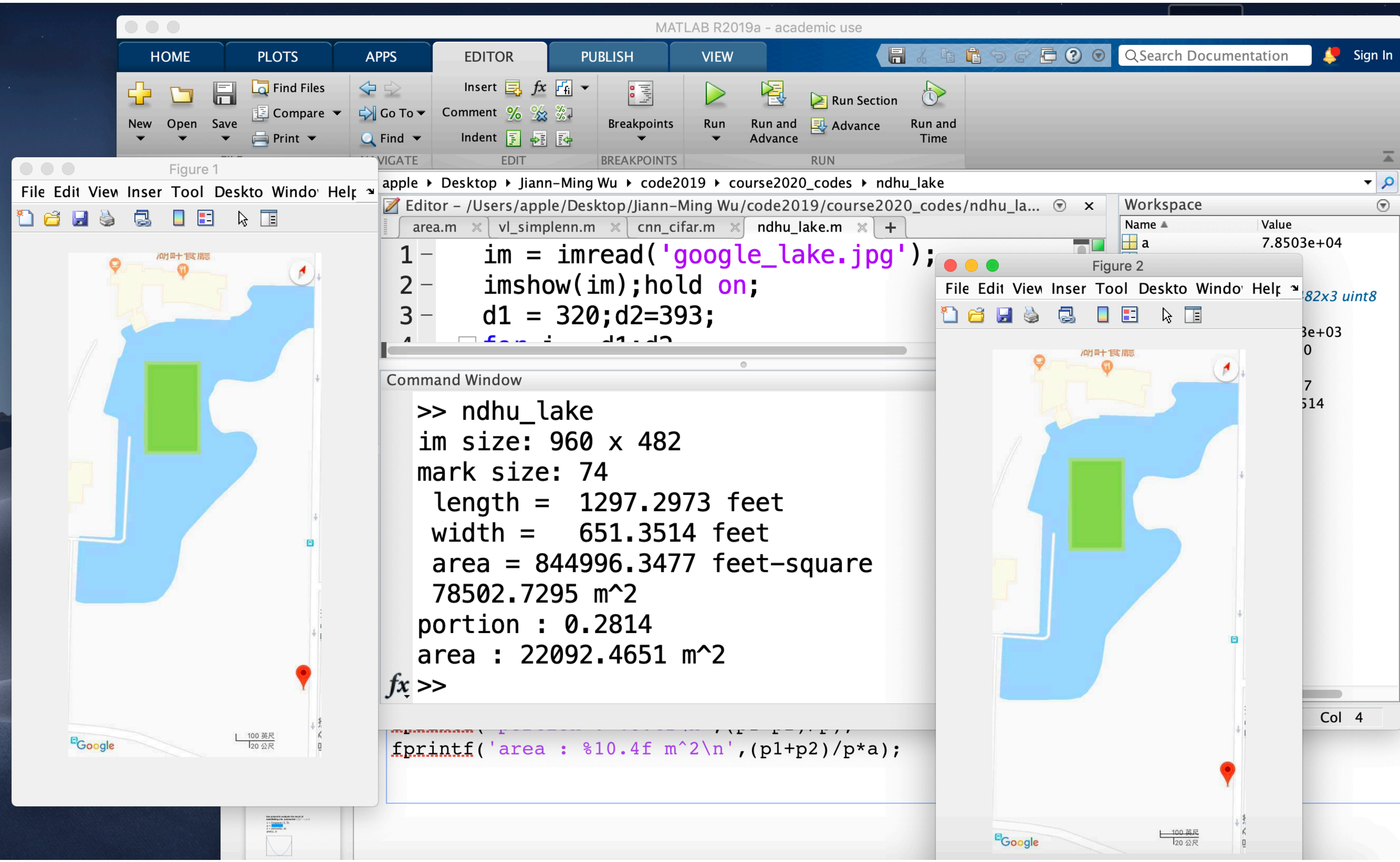


Step 3.2



Step 4

portion : 0.2814
area : 22092.4651 m^2

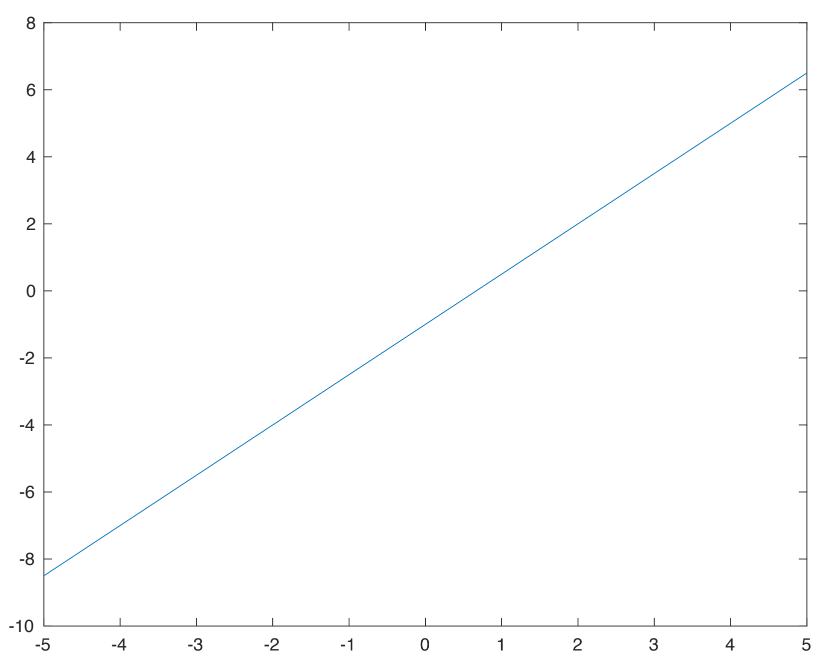


MATLAB R2019a - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
Sign In
New
Open
Save
Find Files
Compare
Print
Go To
Find
Insert
Comment
Indent
Breakpoints
Run
Run and Advance
Run Section
Advance
Run and Time
EDIT
BREAKPOINTS
RUN
apple ▸ Desktop ▸ Jiann-Ming Wu ▸ code2019 ▸ course2020_codes ▸ ndhu_lake
Editor – /Users/apple/Desktop/Jiann-Ming Wu/code2019/course2020_codes/ndhu_la...
area.m
vl_simplenn.m
cnn_cifar.m
ndhu_lake.m
1 – im = imread('google_lake.jpg');
2 – imshow(im);hold on;
3 – d1 = 320;d2=393;
fprintf('area : %10.4f m^2\n',(p1+p2)/p*a);
Command Window
>> ndhu_lake
im size: 960 x 482
mark size: 74
length = 1297.2973 feet
width = 651.3514 feet
area = 844996.3477 feet-square
78502.7295 m^2
portion : 0.2814
area : 22092.4651 m^2
>>
Workspace
Name
Value
a
7.8503e+04
Figure 1
Figure 2
File Edit View Inser Tool Deskto Windo Help
Google
Col 4

```matlab
im = imread('google_lake.jpg');
imshow(im);hold on;
d1 = 320;d2=393;
for j = d1:d2
im(923,j,1:3) = [0 0 0];
end
figure
imshow(im);
fprintf('im size: %d x %d\n',size(im,1),size(im,2));
fprintf('mark size: %d\n',d2-d1+1);
l = size(im,1)/(d2-d1+1)*100;
w = size(im,2)/(d2-d1+1)*100;
fprintf(' length = %10.4f feet\n width = %10.4f feet\n area = %10.4f feet-square\n',l,w,l*w);
a = w*l*0.09290304;
fprintf(' %10.4f m^2\n',a);
p1 = 18303;
p2 = 111917;
p = size(im,1)*size(im,2);
fprintf('portion : %6.4f\n',(p1+p2)/p);
fprintf('area : %10.4f m^2\n',(p1+p2)/p*a);
```

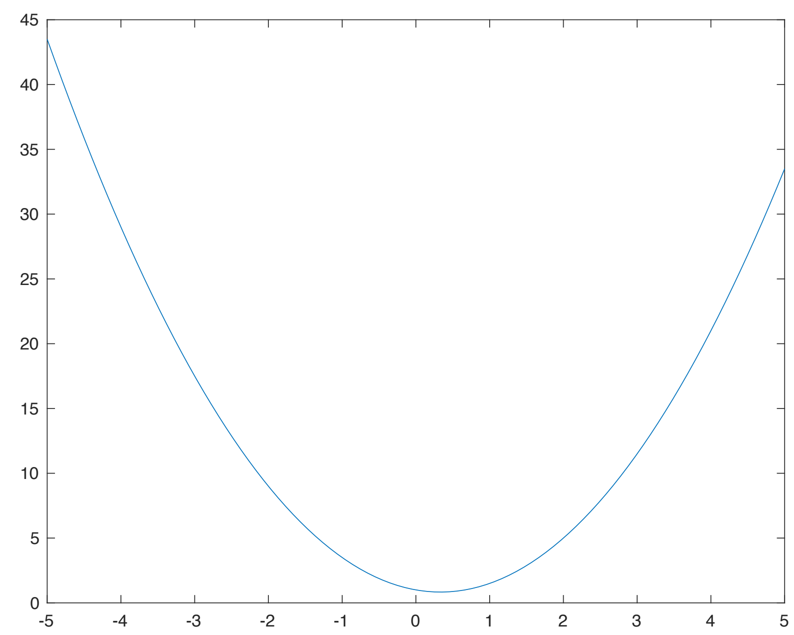
Use polyval to evaluate the result of substituting z to polynomial 1.5x+1

```
z = linspace(-5, 5);
p = [1.5 1]
y =         (p, z);
plot(z, y)
```



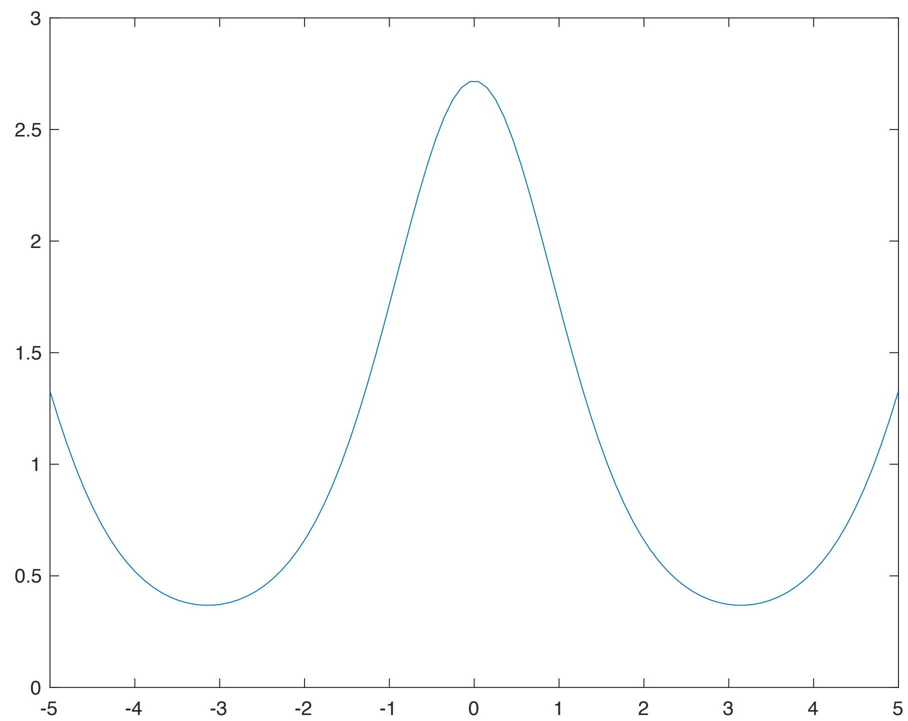
Use polyval to evaluate the result of substituting z to polynomial $1.5x^2 - x + 1$

```
z = linspace(-5, 5);
p = 
y = polyval(p, z);
plot(z, y)
```



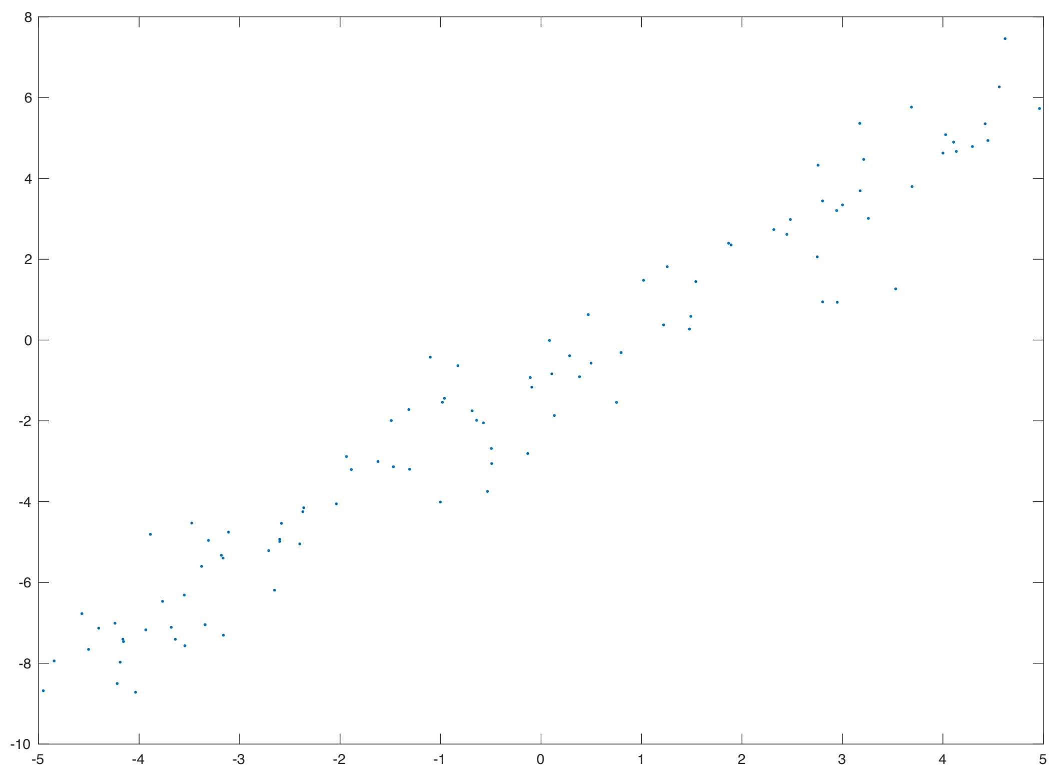
% Draw function exp(cos(x))

```
f = inline('exp(cos(x))');
Z = linspace(-5,5);
plot(          ); hold on
```



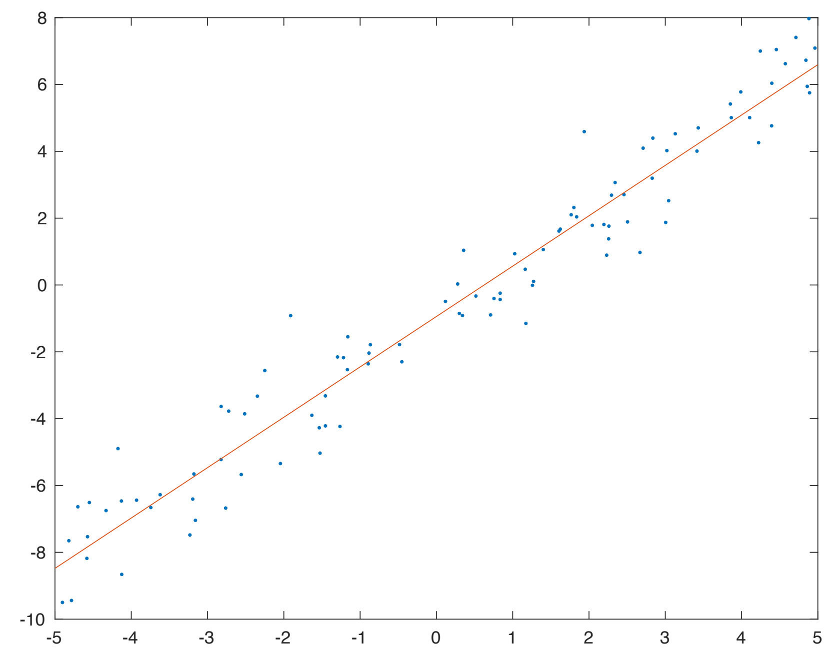
Generate a sample with noises from a line $y = 1.5x - 1$

```
n = 100;
x = rand(1, n) * 10 - 5;
noise = randn(1,n) ;
y = [redacted];
plot(x, y, '.')
```



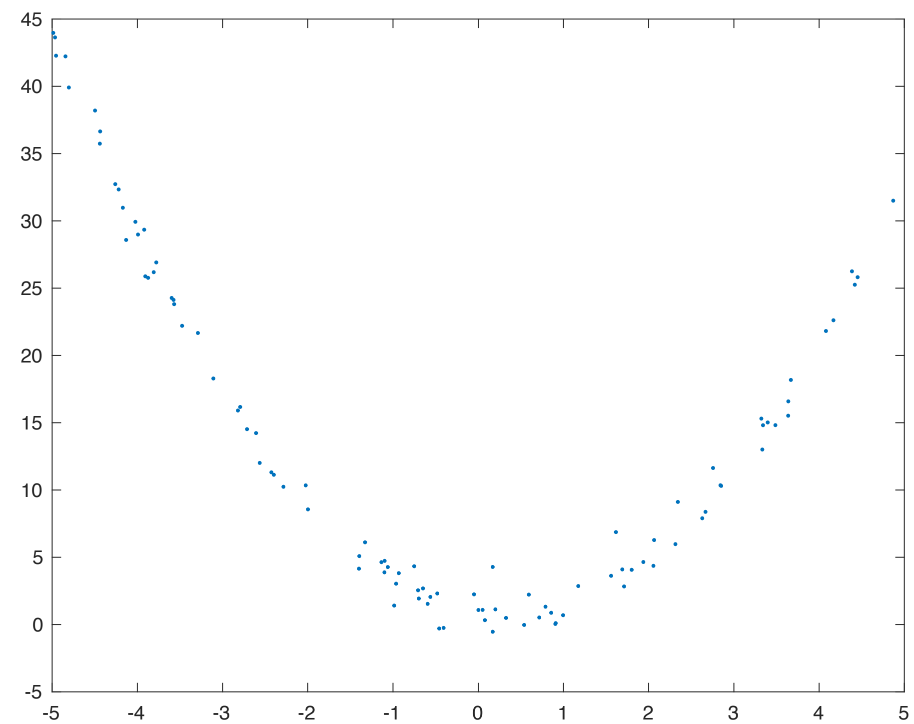
Apply polyfit to estimate polynomial coefficients subject to given data for line fitting

```
n = 100;
x = rand(1, n) * 10 - 5;
noise = randn(1,n) ;
y = polyval([1.5 1], x) + noise;
plot(x, y, '.'); hold on;

p = polyfit(          );
x_new = linspace(-5,5);
y_new = polyval(             );

plot(x_new, y_new)
```



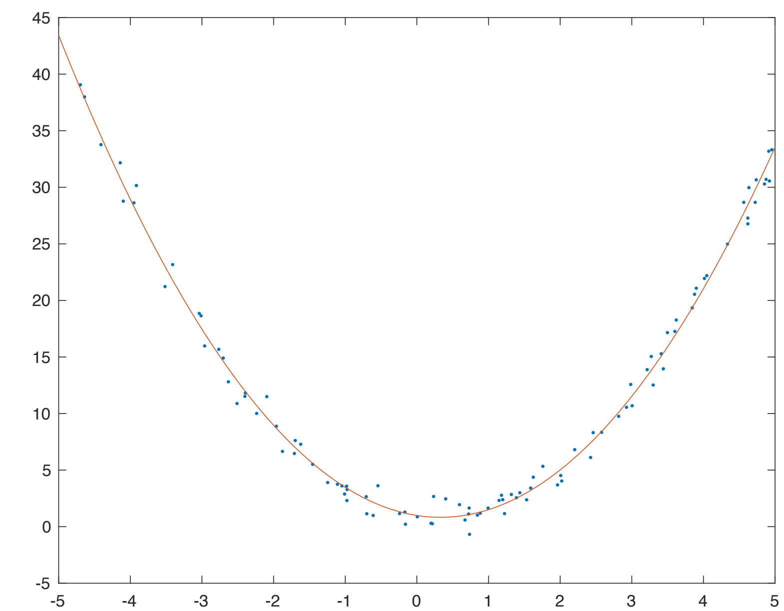
Generate a sample with noises from a curve $1.5x^2 - x + 1$

```
n = 100;
x = rand(1, n) * 10 - 5;
noise = randn(1,n) ;
y = ______ + noise;
plot(x, y, '.')
```



Apply polyfit to estimate polynomial coefficients subject to given data for curve fitting

```
n = 100;
x = rand(1, n) * 10 - 5;
noise = randn(1,n) ;
y = [redacted] + noise;
plot(x, y, '.'); hold on;

p = [redacted]
x_new = linspace(-5,5);
y_new = polyval([redacted]);

plot(x_new, y_new)
```

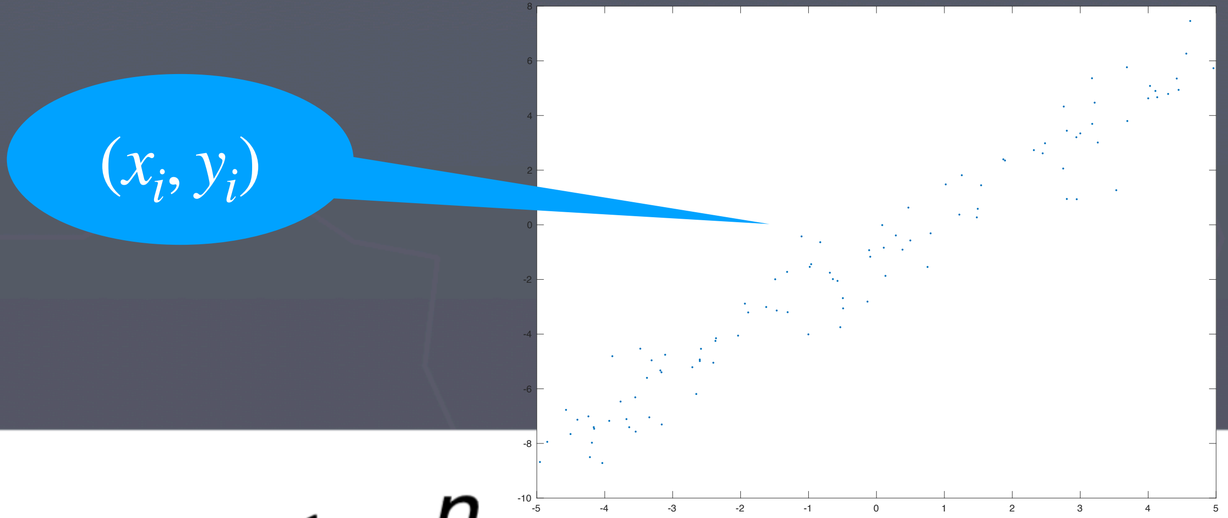


p =

1.4830 -1.0005 1.2400

Objective function I

- Line fitting



$$E_1(\theta) \equiv E_{\text{Line-fitting}}(a,b) = \frac{1}{n}\sum_{i=1}^{n}(ax_i + b - y_i)^2$$

$$y = ax + b$$

$$\mathrm{E}_1(\theta) \equiv \mathrm{E}_{\text{Line-fitting}}(\mathrm{a}, \mathrm{b}) = \frac{1}{n}\sum_{i=1}^{n}(ax_i + b - y_i)^2$$

$$\frac{dE_1}{d\theta} = 0$$

$$\frac{dE_1}{da} = \frac{2}{n}\sum_{i=1}^{n}(ax_i + b - y_i)x_i = 0$$

$$\frac{dE_1}{db} = \frac{2}{n}\sum_{i=1}^{n}(ax_i + b - y_i) = 0$$

$$\frac{dE_1}{da} = \frac{2}{n}\sum_{i=1}^{n}(ax_i + b - y_i)x_i = 0$$

$$\frac{dE_1}{db} = \frac{2}{n}\sum_{i=1}^{n}(ax_i + b - y_i) = 0$$

$$\left\{\begin{aligned} &\sum_{i=1}^{n}(ax_i^2 + bx_i - y_i x_i) = 0 \\ &\sum_{i=1}^{n}(ax_i + b - y_i) = 0 \end{aligned}\right\} \Rightarrow \left\{\begin{aligned} &\sum_{i=1}^{n} x_i^2 a + \sum_{i=1}^{n} x_i b = \sum_{i=1}^{n} x_i y_i \\ &\sum_{i=1}^{n} ax_i + nb = \sum_{i=1}^{n} y_i \end{aligned}\right\}$$

$$\left\{\begin{array}{l} \sum_{i=1}^{n} x_i^2 a + \sum_{i=1}^{n} x_i b = \sum_{i=1}^{n} x_i y_i \\ \sum_{i=1}^{n} x_i a + nb = \sum_{i=1}^{n} y_i \end{array}\right\}$$

$$\begin{pmatrix} \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & n \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{n} x_i y_i \\ \sum_{i=1}^{n} y_i \end{pmatrix}$$

$$\begin{pmatrix} \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & n \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{n} x_i \, y_i \\ \sum_{i=1}^{n} y_i \end{pmatrix}$$

$$A(1,1) = \sum_{i=1}^{n} x_i^2 \quad A(1,2) = \sum_{i=1}^{n} x_i \quad c(1) = \sum_{i} x_i y_i$$

$$A(2,1) = \sum_{i=1}^{n} x_i \quad A(2,2) = n \quad c(2) = \sum_{i} y_i$$

Try to calculate A(1, 1), A(1, 2) and d(1) and estimate polynomial coefficients for line fitting

```
n = 100;
x = rand(1, n) * 10 - 5;
noise = randn(1,n) ;
y = polyval([1.5 1],x) + noise;
plot(x, y, '.'); hold on;

A(1, 1) = ________ ;
A(1, 2) = ________ ;
A(2, 1) = A(1, 2);
A(2, 2) = n;
d(1) = ________ ;
d(2) = sum(y);
inv(A) * d'
```

ans = 1.47900 1.0776

$$A(1,1) = \sum_{i=1}^{n} x_i^2$$

$$A(2,1) = \sum_{i=1}^{n} x_i$$

$$d(1) = \sum_{i} x_i y_i$$

$$d(2) = \sum_{i} y_i$$

Quadratic polynomial fitting

$$E_2(\theta) \equiv E_{\text{QuadraticCurveFitting}}(a, b, c)$$

$$= \frac{1}{n}\sum_{i=1}^{n}(ax_i^2 + bx_i + c - y_i)^2$$

$$y = ax^2 + bx + c$$

$$\mathrm{E}_2(\theta) \equiv \mathrm{E}_{\mathrm{QuadraticCurveFitting}}(\mathrm{a}, \mathrm{b}, \mathrm{c}) = \frac{1}{n}\sum_{i=1}^{n}(ax_i^2 + bx_i + c - y_i)^2$$

$$\frac{dE_2}{d\theta} = 0$$

$$\frac{dE_2}{da} = \frac{2}{n}\sum_{i=1}^{n}(ax_i^2 + bx_i + c - y_i)x_i^2 = 0$$

$$\frac{dE_2}{db} = \frac{2}{n}\sum_{i=1}^{n}(ax_i^2 + bx_i + c - y_i)x_i = 0$$

$$\frac{dE_3}{dc} = \frac{2}{n}\sum_{i=1}^{n}(ax_i^2 + bx_i + c - y_i) = 0$$

$$\begin{pmatrix} \sum_{i=1}^{n}(ax_i^2 + bx_i + c - y_i)x_i^2 = 0 \\ \sum_{i=1}^{n}(ax_i^2 + bx_i + c - y_i)x_i = 0 \\ \sum_{i=1}^{n}(ax_i^2 + bx_i + c - y_i) = 0 \end{pmatrix}$$

$$\begin{pmatrix} \sum_{i=1}^{n}(ax_i^4 + bx_i^3 + cx_i^2 - y_i x_i^2) = 0 \\ \sum_{i=1}^{n}(ax_i^3 + bx_i^2 + cx_i - y_i x_i) = 0 \\ \sum_{i=1}^{n}(ax_i^2 + bx_i + c - y_i) = 0 \end{pmatrix}$$

$$\begin{cases} \sum_{i=1}^{n}(ax_i^4 + bx_i^3 + cx_i^2 - y_i x_i^2) = 0 \\ \sum_{i=1}^{n}(ax_i^3 + bx_i^2 + cx_i - y_i x_i) = 0 \\ \sum_{i=1}^{n}(ax_i^2 + bx_i + c - y_i) = 0 \end{cases}$$

$$\begin{cases} \sum_{i=1}^{n} ax_i^4 + \sum_{i=1}^{n} bx_i^3 + \sum_{i=1}^{n} cx_i^2 = \sum_{i=1}^{n} y_i x_i^2 \\ \sum_{i=1}^{n} ax_i^3 + \sum_{i=1}^{n} bx_i^2 + \sum_{i=1}^{n} cx_i = \sum_{i=1}^{n} y_i x_i \\ \sum_{i=1}^{n} ax_i^2 + \sum_{i=1}^{n} bx_i + nc = \sum_{i=1}^{n} y_i \end{cases}$$

$$\begin{pmatrix} \sum_{i=1}^{n} ax_i^4 + \sum_{i=1}^{n} bx_i^3 + \sum_{i=1}^{n} cx_i^2 = \sum_{i=1}^{n} y_i x_i^2 \\ \sum_{i=1}^{n} ax_i^3 + \sum_{i=1}^{n} bx_i^2 + \sum_{i=1}^{n} cx_i = \sum_{i=1}^{n} y_i x_i \\ \sum_{i=1}^{n} ax_i^2 + \sum_{i=1}^{n} bx_i + nc = \sum_{i=1}^{n} y_i \end{pmatrix}$$

$$\begin{pmatrix} \sum_{i=1}^{n} x_i^4 & \sum_{i=1}^{n} x_i^3 & \sum_{i=1}^{n} x_i^2 \\ \sum_{i=1}^{n} x_i^3 & \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i & n \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{n} y_i x_i^2 \\ \sum_{i=1}^{n} y_i x_i \\ \sum_{i=1}^{n} y_i \end{pmatrix}$$

Try to calculate A(1, 1), A(1, 2) and d(1) for estimating quadratic curve coefficients

```
n = 100;
x = rand(1, n) * 10 - 5;
noise = randn(1,n) ;
y = polyval([1.5 -1 1],x) + noise;
plot(x, y, '.'); hold on;

A(1, 1) = ____________ ;
A(1, 2) = ____________ ;  A(2, 1) = A(1, 2);
A(1, 3) = sum(x.^2); A(3, 1) = A(1, 3);
A(2, 2) = sum(x.^2);
A(2, 3) = sum(x); A(3, 2) = A(2, 3);
A(3, 3) = n;
d(1) = ____________ ;
d(2) = sum(y.*x);
d(3) = sum(y);
p_hat = inv(A) * d'
```

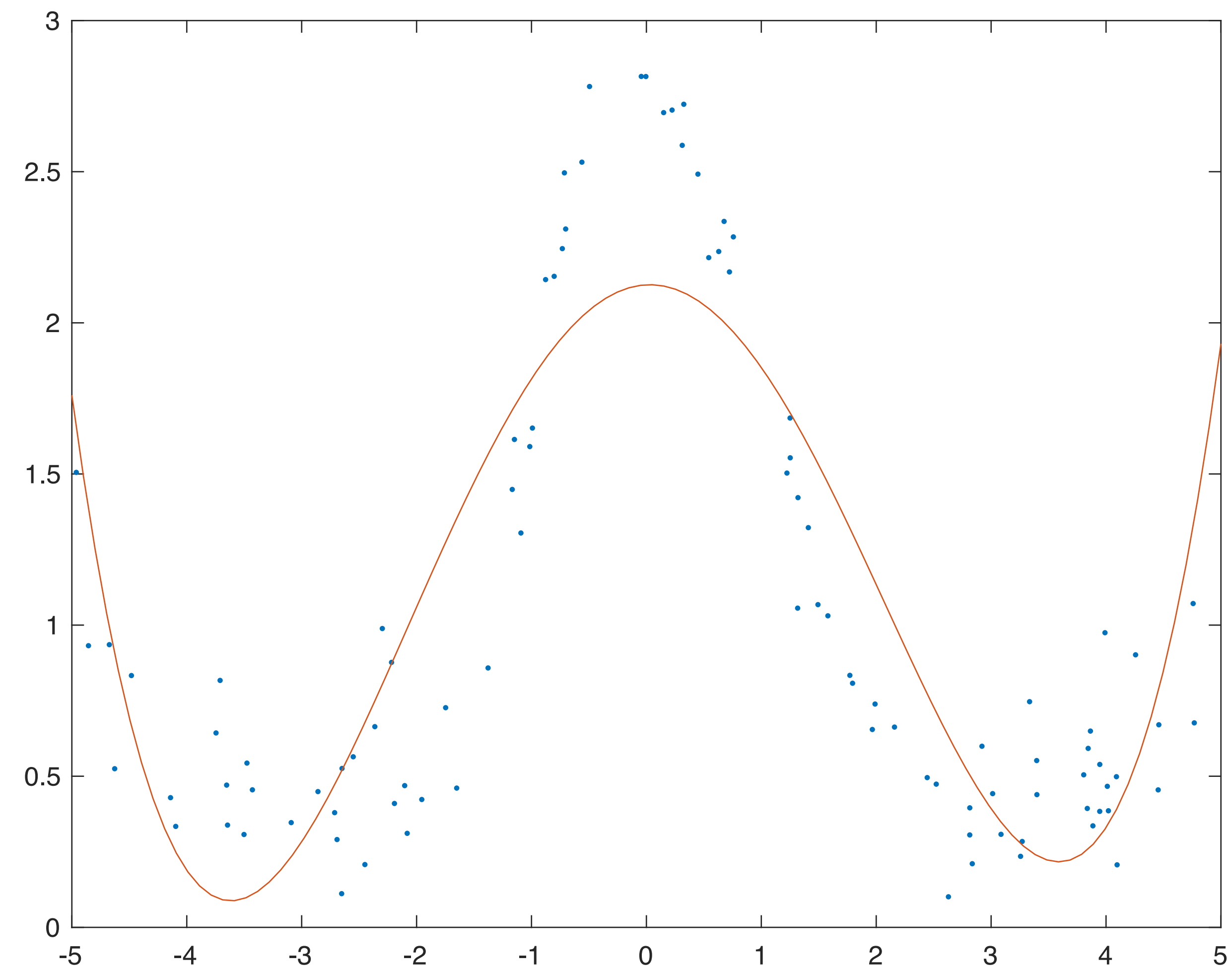
p_hat =

1.4999
-1.0617
1.0099

$$\begin{pmatrix} \sum_{i=1}^{n} x_i^4 & \sum_{i=1}^{n} x_i^3 & \sum_{i=1}^{n} x_i^2 \\ \sum_{i=1}^{n} x_i^3 & \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i & n \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{n} y_i x_i^2 \\ \sum_{i=1}^{n} y_i x_i \\ \sum_{i=1}^{n} y_i \end{pmatrix}$$

Apply polyfit to estimate coefficients of a 4-degree polynomial subject to given data for approximating exp(cos(x))

```
n = 100;
x = rand(1, n) * 10 - 5;
noise = randn(1,n) * 0.2;
str = '[redacted]';
f = inline(str);
y = [redacted] + noise;
plot(x, y, '.'); hold on;

p = [redacted];
x_new = linspace(-5,5);
y_new = polyval(p, x_new);

plot(x_new, y_new)
```



p =

0.0117 -0.0001 -0.3041 0.0189 2.1259