Lecture 4

- For-looping
- Line fitting
- Thresholding
- Find, mean
- Points within a circle
- Quadratic curve fitting

Line fitting

x =

1 2 3 4 5

y =

3 5 7 9 11

Fitting line y=ax+b to vectors x and y
Find a and b

a=2, b=1

Methodology

$$C\begin{bmatrix} a \\ b \end{bmatrix} = d$$

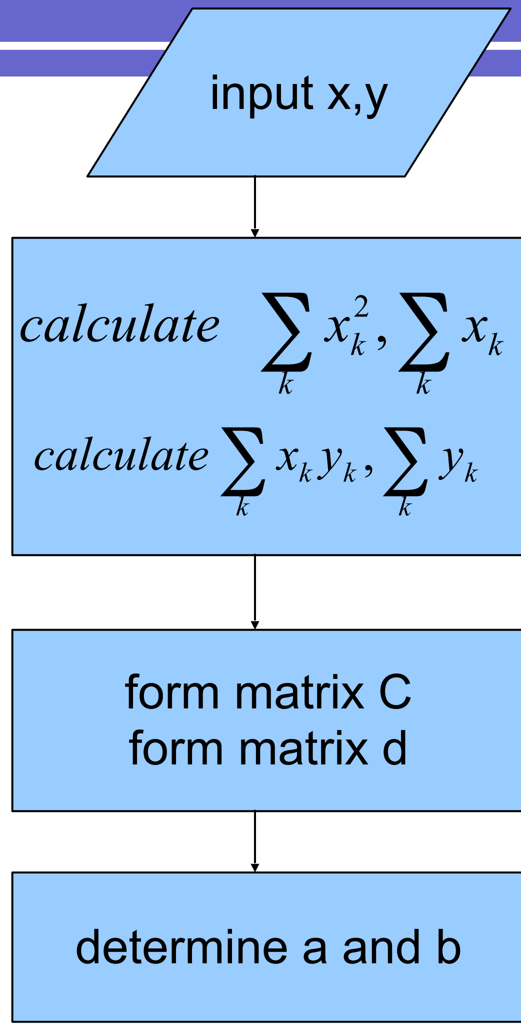
$$\begin{bmatrix} a \\ b \end{bmatrix} = C^{-1}d$$

$$C = \begin{bmatrix} \sum_k x_k^2 & \sum_k x_k \\ \sum_k x_k & \mathrm{N} \end{bmatrix}$$

$$d = \begin{bmatrix} \sum_k x_k y_k \\ \sum_k y_k \end{bmatrix}$$

Flow chart

function [a,b]=find_ab(x,y)

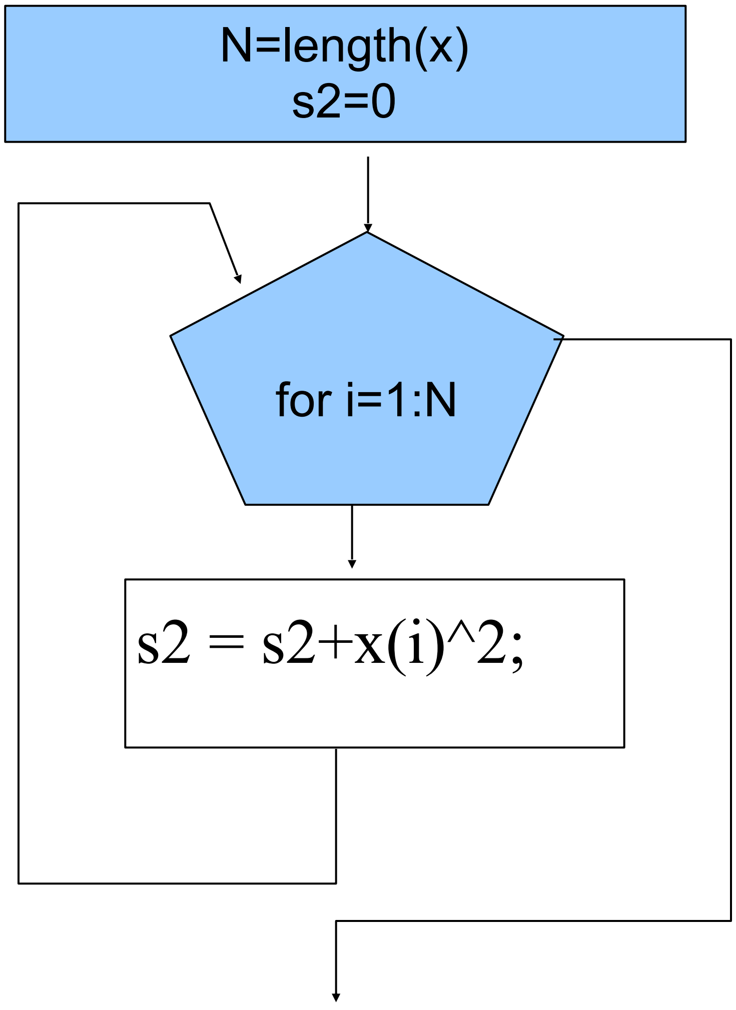


Apply for-loops to calculate four terms

C(1,1)=s2;
C(1,2)=sxy;
C(2,1)=sxy;

Flow chart

$calculate \quad \sum_k x_k^2$

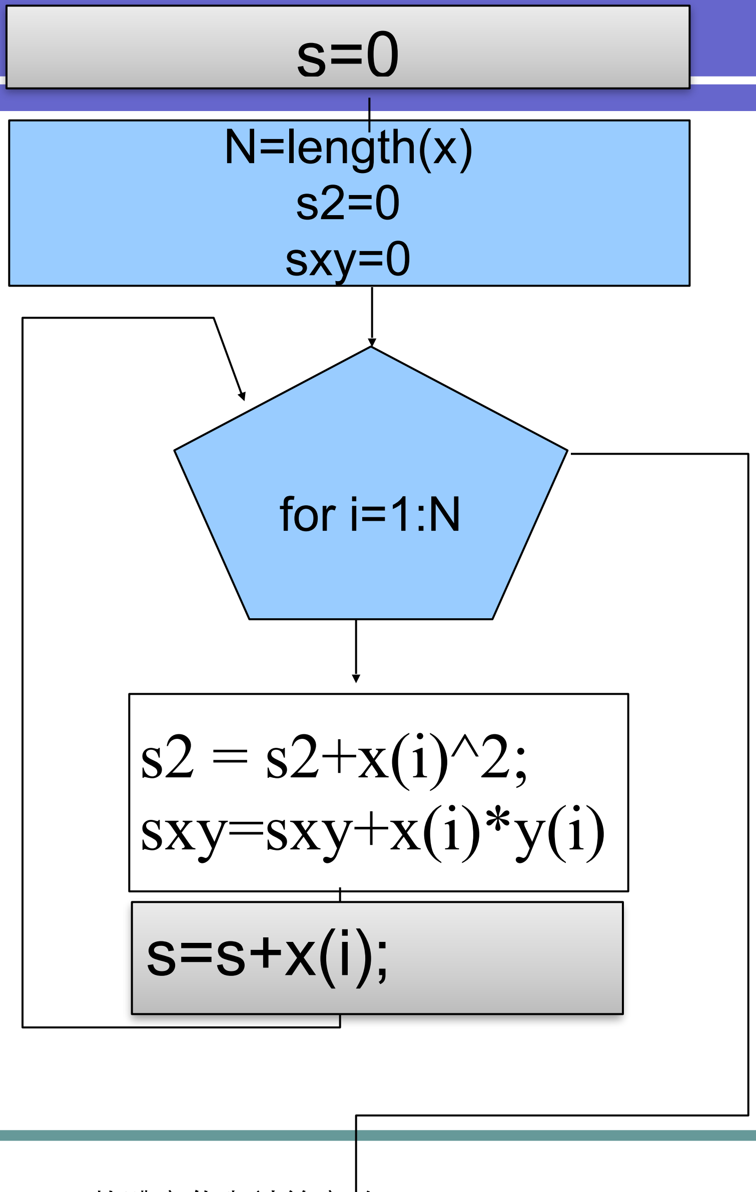


Flow chart

calculate $\sum_k x_k^2$

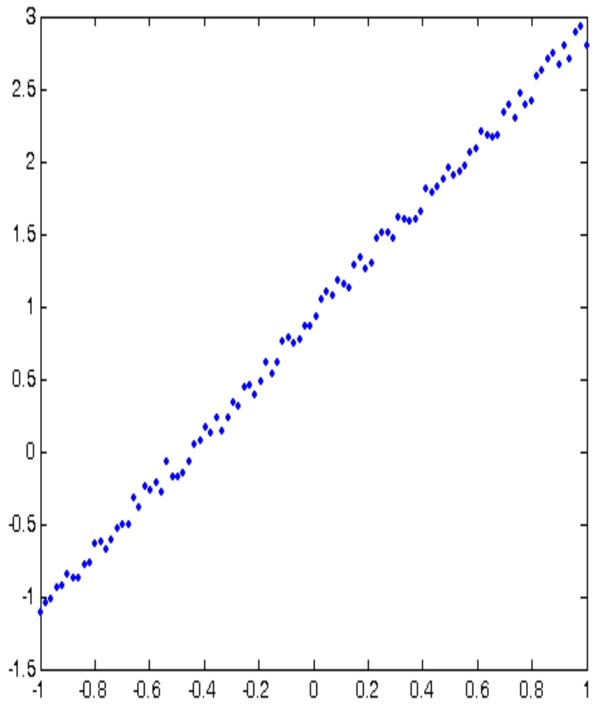
and $\sum_k x_k y_k$, $\sum x_i$

at the same loop



Noisy sample from a line

```
>> x=linspace(-1,1,100);
>> y=2*x+1+rand(1,100)*0.2-.01;
>> plot(x,y,'.');
```



Errors

$$E(a,b) = \sum_i (ax_i + b - y_i)^2$$

$$E \geq 0$$

$$E(a = 2, b = 1) = ?\ given\ \mathrm{x, y}$$

Mean square fitting error

x =

1 2 3 4 5

y =

3 5 7 9 11

a=2, b=1

Find the mean square error
of fitting y=ax+b to vectors x and y

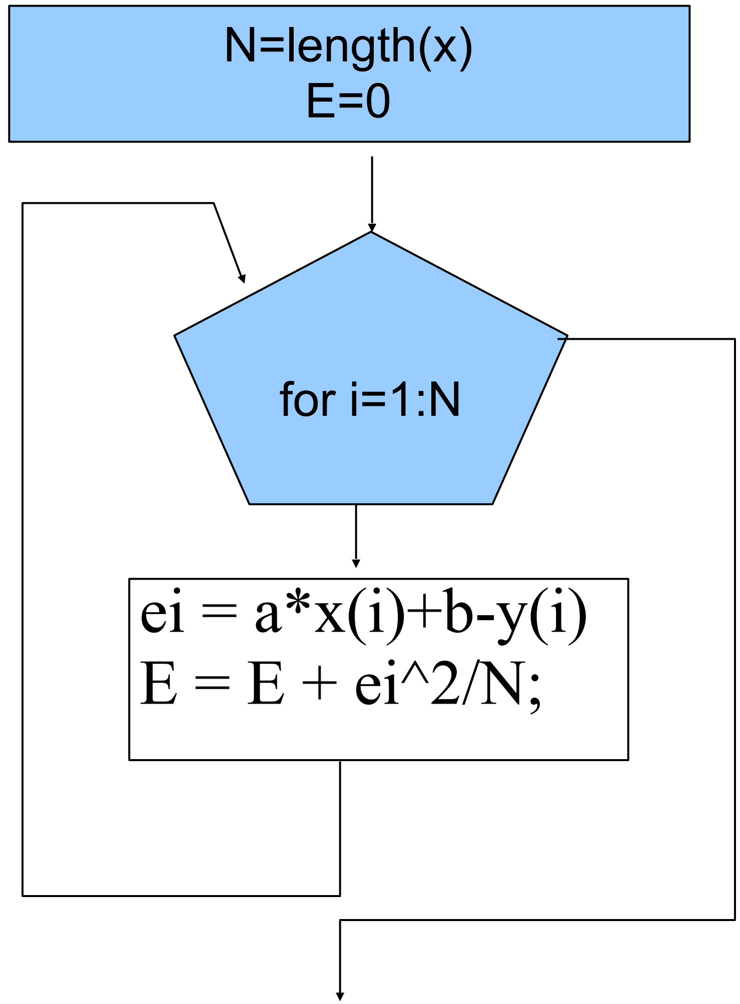
$$E(a,b) = \frac{1}{N}\sum_i (ax_i + b - y_i)^2$$

Flow chart

function E=find_E(x,t,a,b)

calculate

$$E(a,b) = \frac{1}{N}\sum_i (ax_i + b - y_i)^2$$



Find

function ind = my_find(x)

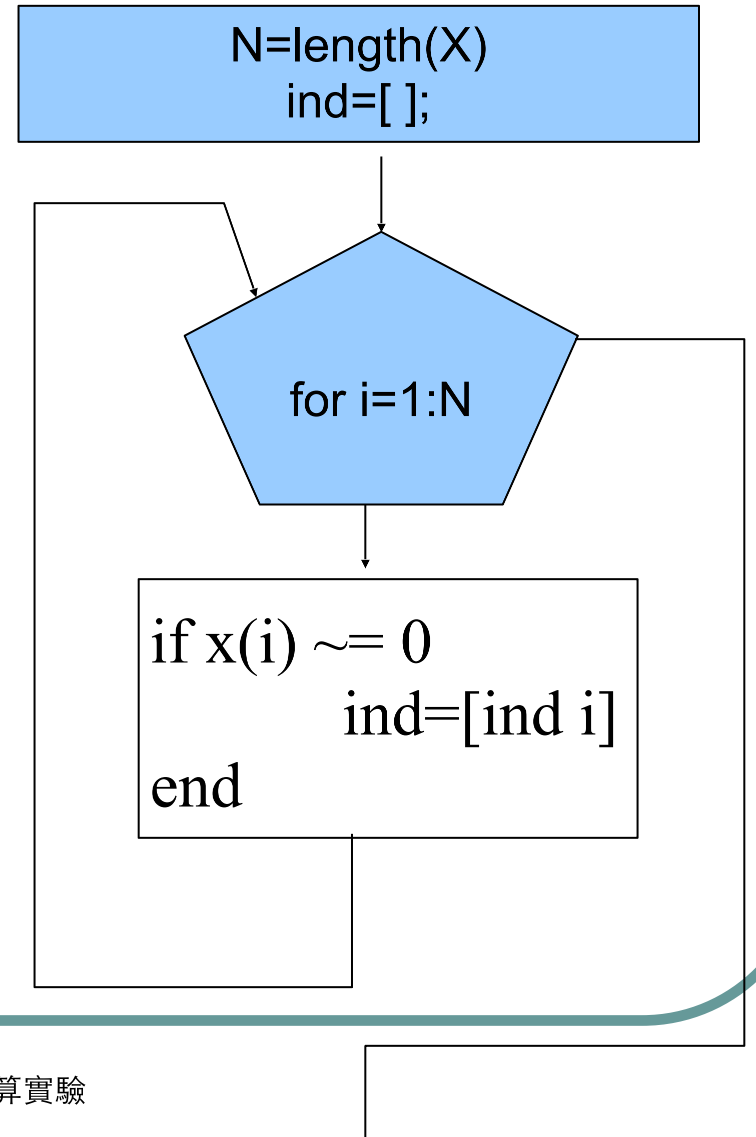
x =

-1 -2 0 1 2 3

ind=find(x)

ind =

1 2 4 5 6



```
function ind=myfind(x)
N=length(X)
ind=[ ];
for i=1:N
    if x(i) ~= 0
        ind=[ind i]
    end
end
```

Thresholding

x =

0.9501 0.2311 0.6068 0.4860 0.8913 0.7621

z =

0 0 0 0 0 0

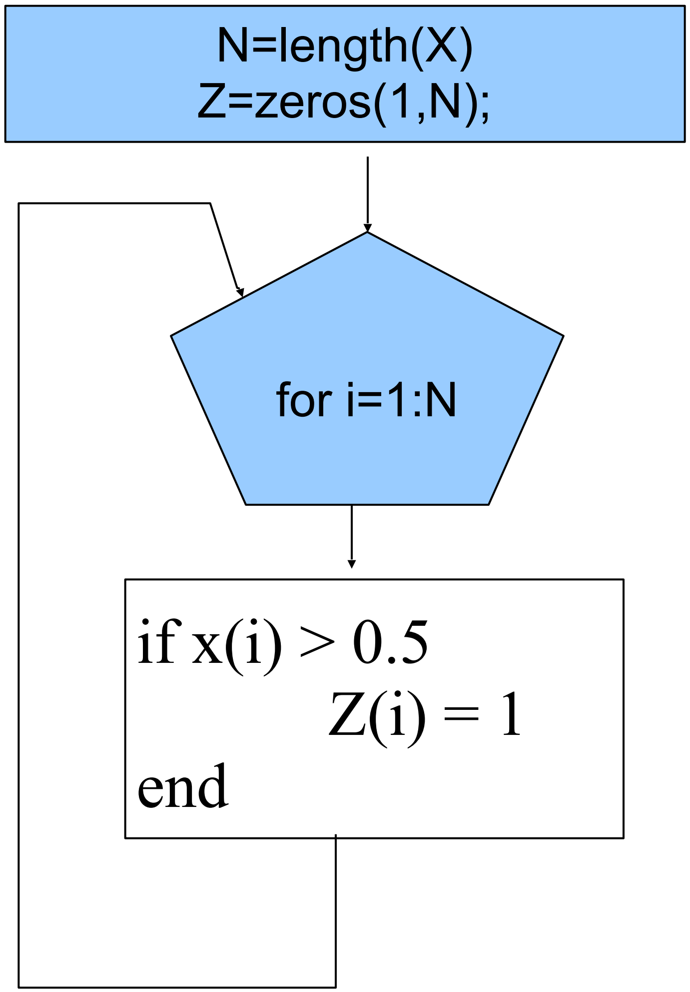
```
ind = find(x > 0.5);
z(ind)=1;
```

z =

1 0 1 0 1 1

For looping

function Z=myfind(x)



X =

0.8709	-0.1795	-0.8842	0.6263	-0.7222
0.8338	0.7873	-0.2943	-0.9803	-0.5945

```
RD=(X(1,:).^2+X(2,:).^2);
```

RD =

1.4537	0.6521	0.8684	1.3532	0.8750

```
Ind = find(RD < 1)
```

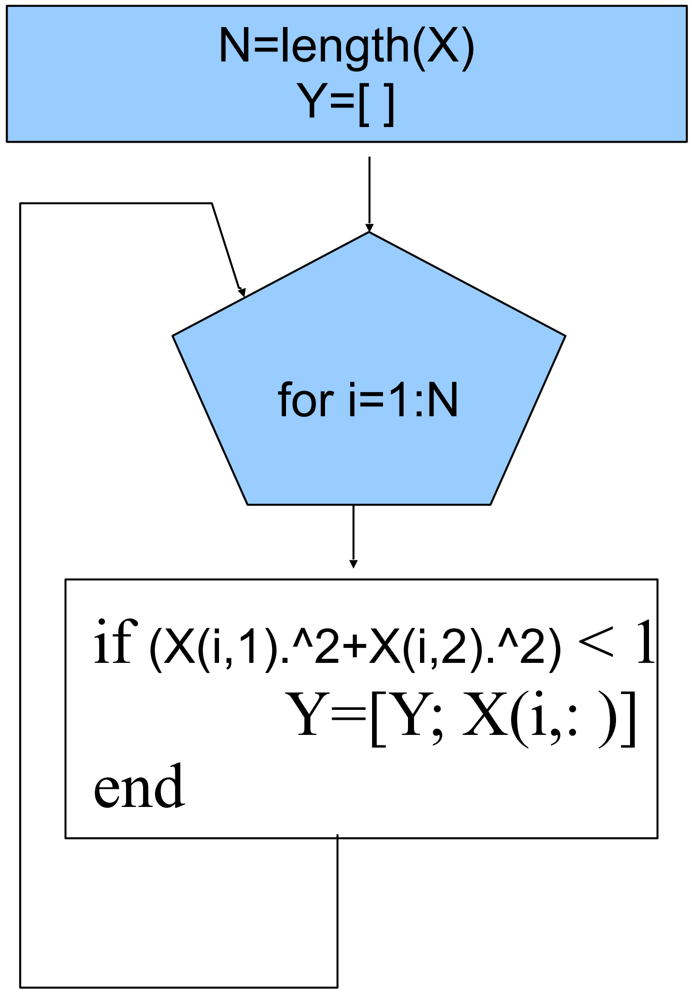
ind =

2 3 5

```
y = x(:,Ind)
```

-0.1795	-0.8842	-0.7222
0.7873	-0.2943	-0.5945

function Y=unit_circle(X)



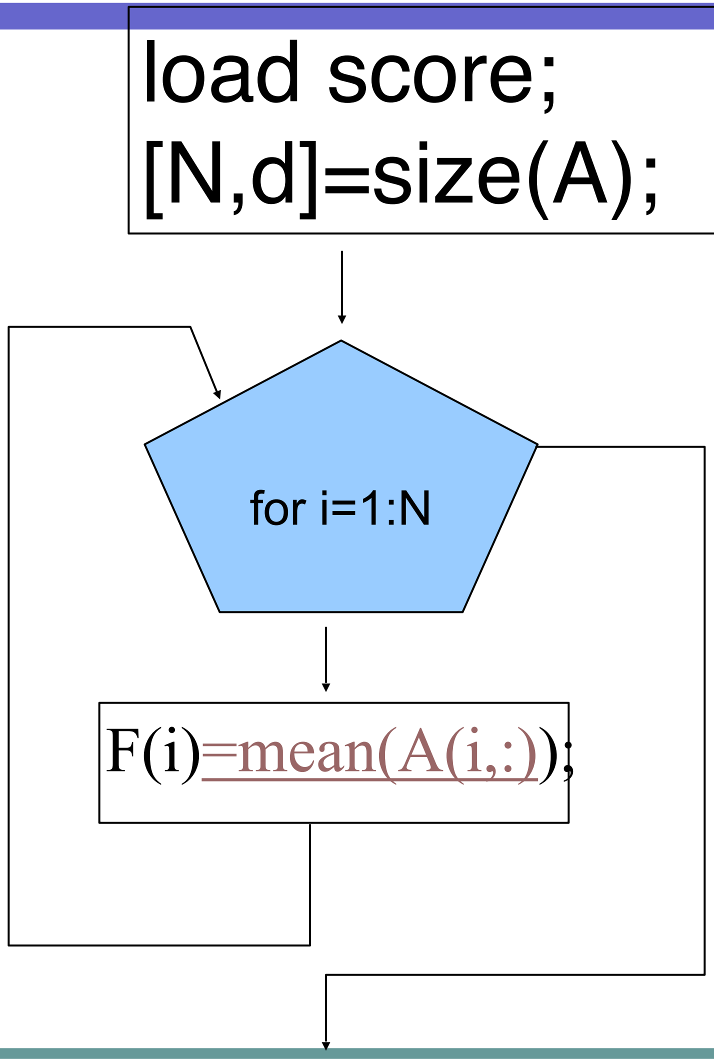
Generation of scores

- N=10;d=3;
- A=ceil(rand(N,d)*100);
- save score.mat A;

Load scores

- load score.mat;
- [N,d]=size(score);

Flow chart: Means of scores



function F=mean_sr(A)

- Increasing index
- Execute same instructions iteratively with an increasing looping index

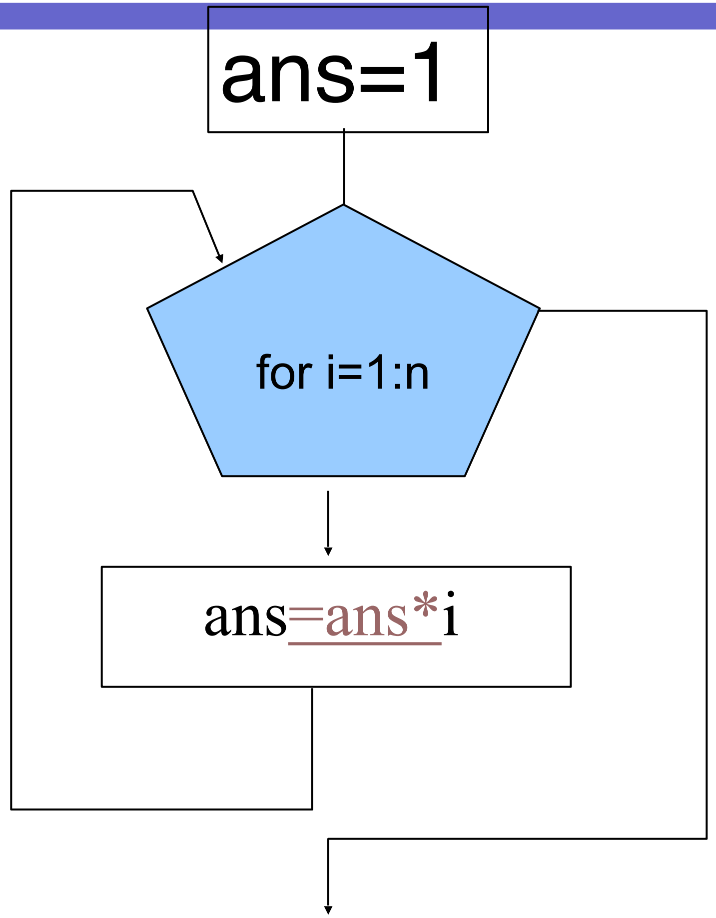
```
load score;
[N,d]=size(A);
for i=1:N
        F(i)=mean(A(i,:));
end
```

Factorial

- Write a matlab function to calculate n!
- Input: a positive integer, n
- Output: n!

Flow chart

function ans=mt_fact(n)



n!

- Increasing index
- Execute same instructions iteratively with an increasing looping index

Increasing index

1 : n

1 : 1 : n

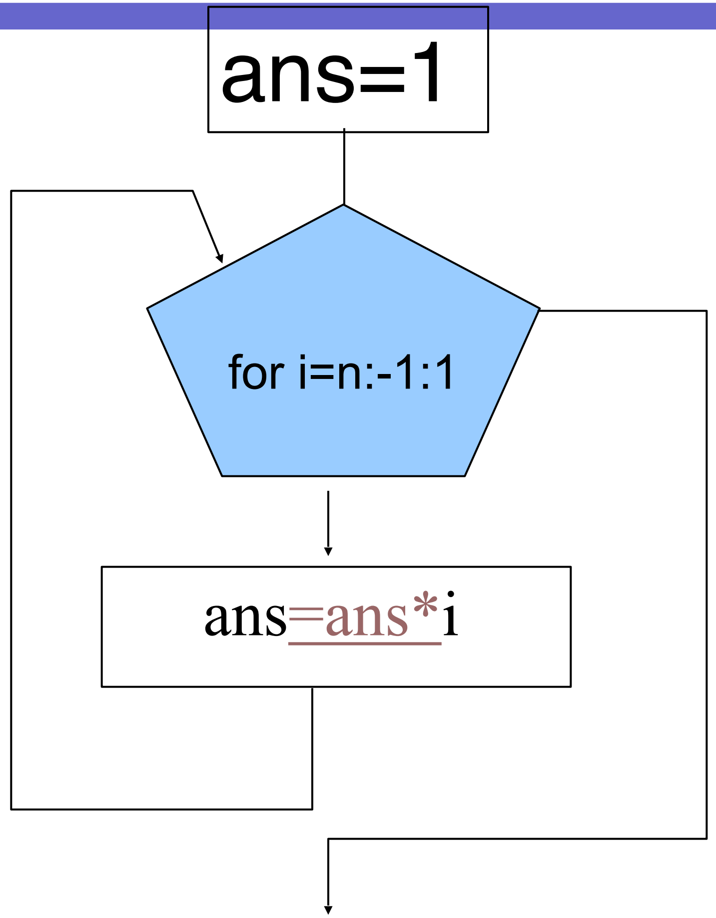
1 : 3 : n

end

Step size

start

Decreasing



n!

- Decreasing index
- Iterative execution with a decreasing index

Decreasing index

n : -1 : 1

n : -3 : 1

n1: -2 : n2

end

negative step size

start

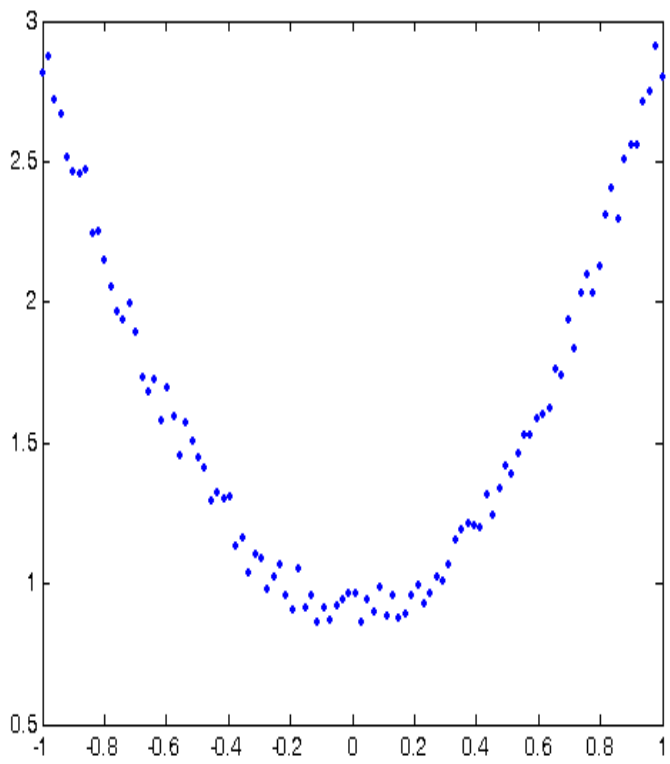
Factorial function

source codes

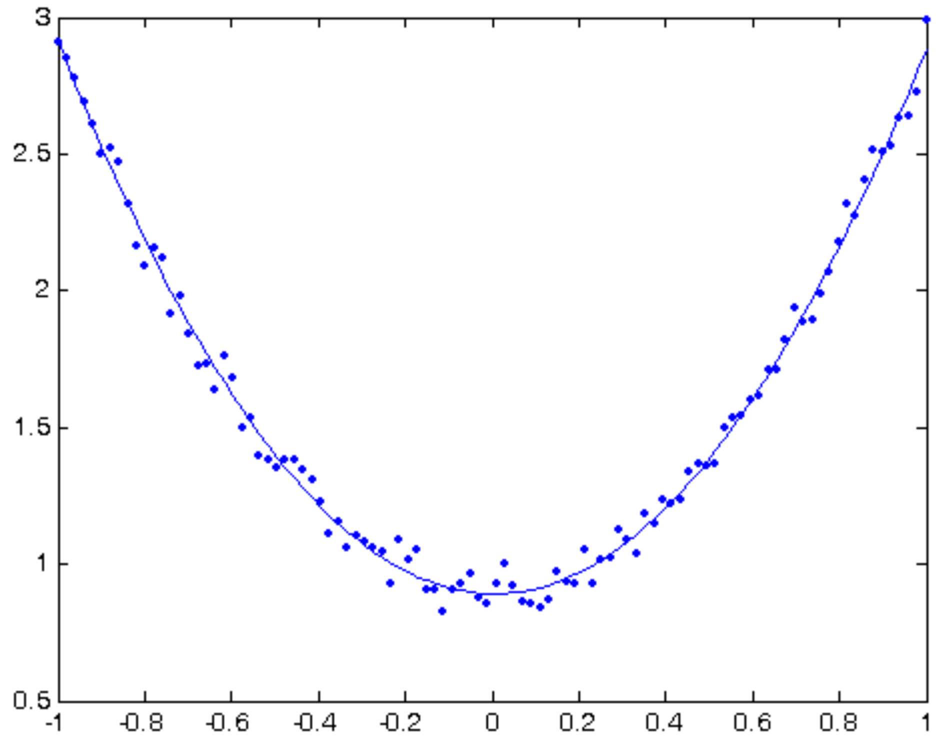
```
function ans=factorial(n)
    ans=1;
    for i=1:n
        ans=ans*i;
    end
return
```

Noisy sample from a quadratic curve

```
x=linspace(-1,1,100);
y=2*x.^2+1-rand(1,100)*0.2;
plot(x,y,'.');
```



Mean square error



Mean square error

$$E(a,b,c) = \frac{1}{N}\sum_i (ax_i^2 + bx_i + c - y_i)^2$$

x =

-1 0 1 2 3 4

y =

2 1 4 11 22 37

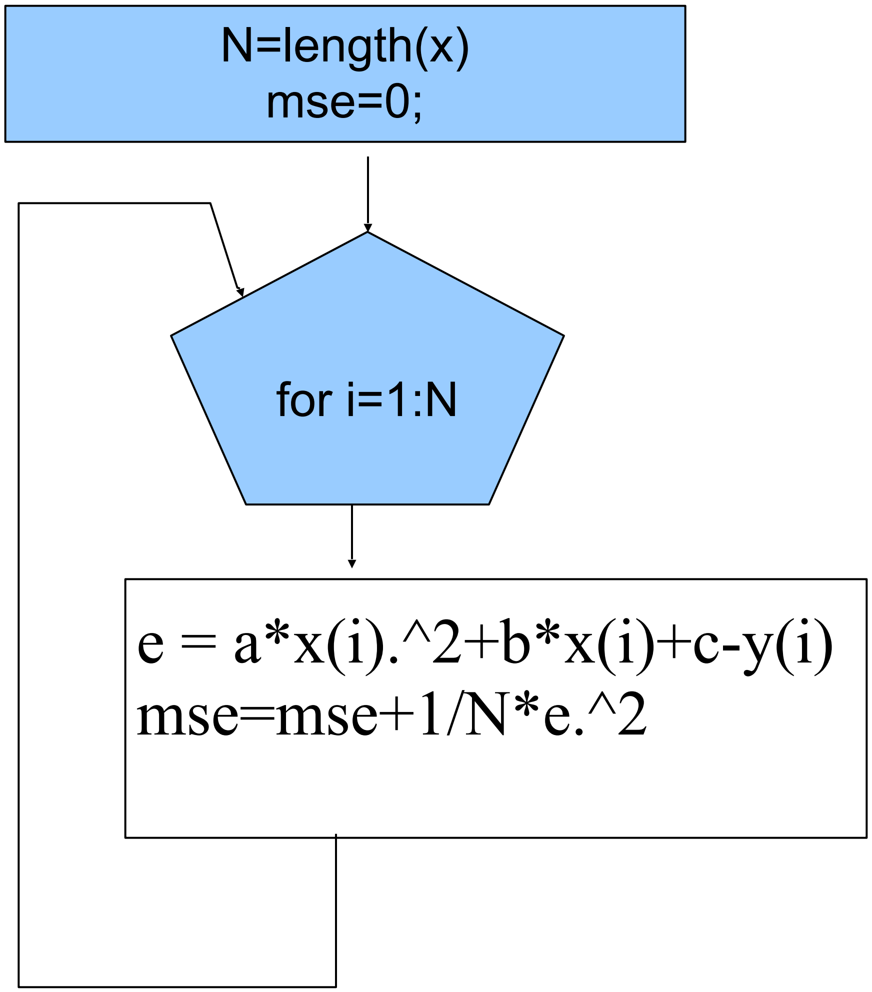
```
a=2;b=1;c=1
e=a*x.^2+b*x+c-y
```

e =

0 0 0 0 0 0 → mean(e.^2)

For looping

function mse=mse_qf(x,y,a,b,c)



```
N=length(X)
mse=0;
for i=1:N
    e = a*x(i).^2+b*x(i)+c-y(i)
    mse=mse+1/N*e.^2
end
```

Unconstrained optimization

$$\min_{a,b,c} E(a,b,c)$$

$$E(a,b,c) = \sum_i \left(y_i - (ax_i^2 + bx_i + c)\right)^2$$

Fitting quadratic curves

Find coefficients of a quadratic Polynomial that well fits given paired data, $\{(x_i,y_i)\}$.

Quadratic form

- E is non-negative and quadratic
- The minimum can be determined by

$$\frac{\partial E}{\partial a} = 0, \frac{\partial E}{\partial b} = 0, \frac{\partial E}{\partial c} = 0$$

Normal equation I

$$\frac{\partial E}{\partial a} = 0$$

$$\Updownarrow$$

$$-2\sum_i (y_i - (ax_i^2 + bx_i + c))x_i^2 = 0$$

$$(\sum_i x_i^4)a + (\sum_i x_i^3)b + (\sum_i x_i^2)c = \sum_i y_i x_i^2$$

Normal Equation II

$$\frac{\partial E}{\partial b} = 0$$

$$-2\sum_i (y_i - (ax_i^2 + bx_i + c))x_i = 0$$

$$(\sum_i x_i^3)a + (\sum_i x_i^2)b + (\sum_i x_i)c = \sum_i y_i x_i$$

Normal Equation III

$$\frac{\partial E}{\partial c} = 0$$

$$\Updownarrow$$

$$-2\sum_i (y_i - (ax_i^2 + bx_i + c)) = 0$$

$$\left(\sum_i x_i^2\right)a + \left(\sum_i x_i\right)b + \left(\sum_i 1\right)c = \sum_i y_i$$

Linear system

$$(\sum_i x_i^4)a + (\sum_i x_i^3)b + (\sum_i x_i^2)c = \sum_i y_i x_i^2$$

$$(\sum_i x_i^3)a + (\sum_i x_i^2)b + (\sum_i x_i)c = \sum_i y_i x_i$$

$$(\sum_i x_i^2)a + (\sum_i x_i)b + (\sum_i 1)c = \sum_i y_i$$

Matrix form

$$\begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix}$$

```
E(1,:)=[sum(x.^4) sum(x.^3) sum(x.^2)];
E(2,:)=[sum(x.^3) sum(x.^2) sum(x)];
E(3,:)=[sum(x.^2) sum(x) length(x)];
D=[sum(x.^2.*y) sum(x.*y) sum(y)]';
```

Quadratic curve fitting

- Input paired data
- Calculate e_{ij} and d_i for i,j=1,2,3
- Apply inv instruction to determine a,b,c

Mean square error

$$E(a,b,c) = \frac{1}{N}\sum_i (ax_i^2 + bx_i + c - y_i)^2$$

x =

-1 0 1 2 3 4

y =

2 1 4 11 22 37

```
a=2;b=1;c=1
e=a*x.^2+b*x+c-y
```

e =

0 0 0 0 0 0 → mean(e.^2)

Mean square error

$$E(a,b,c)=\frac{1}{N}\sum_i (ax_i^2+bx_i+c-y_i)^2$$

x =

-1 0 1 2 3 4

y =

2 1 4 11 22 37

```
v=rand(1,3);
a_est=v(1); b_est=v(2); c_est=v(3)
e=a_est*x.^2+b_est*x+c_est-y
```

e =

-1.5000 -0.5000 -2.5000 -7.5000 -15.5000 -26.5000

→ mean(e.^2)

Matrix form

$$\begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix}$$

```
E(1,:)=[sum(x.^4) sum(x.^3) sum(x.^2)];
E(2,:)=[sum(x.^3) sum(x.^2) sum(x)];
E(3,:)=[sum(x.^2) sum(x) length(x)];
D=[sum(x.^2.*y) sum(x.*y) sum(y)]';
```

Mean square error

$$E(a,b,c) = \frac{1}{N}\sum_i (ax_i^2 + bx_i + c - y_i)^2$$

x =

-1 0 1 2 3 4

y =

2 1 4 11 22 37

```
v=inv(E)*D;
a_est=v(1); b_est=v(2); c_est=v(3)
e=a_est*x.^2+b_est*x+c_est-y
```

e =

0 0 0 0 0 0 → mean(e.^2)

Demo_QCF

source code

```
function demo_QCF()
x=linspace(-1,1,100);
y=2*x.^2+1-rand(1,100)*0.2;
plot(x,y,'.');hold on
v=QCF(x,y)
y_hat=v(1)*x.^2+v(2)*x+v(3);
plot(x,y_hat);
return
function v=QCF(x,y)
E(1,:)=[sum(x.^4) sum(x.^3) sum(x.^2)];
E(2,:)=[sum(x.^3) sum(x.^2) sum(x)];
E(3,:)=[sum(x.^2) sum(x) length(x)];
D=[sum(x.^2.*y) sum(x.*y) sum(y)]';
v=inv(E)*D;
return
```