Vectores

- Almacenamento de comandos en ficheiro: *diary ficheiro.txt*; *diary on*; *diary off*.
- Definición entre corchetes: *v = [1 2 3]*: elementos separados por espacios ou comas. Separación entre filas mediante ;
- Vector columna: *v = [1;2;3]*
- Trasposición dun vector: *v*'
- Definición con compoñentes equiespaciadas ($v_{i+1}-v_i=cte$) nun intervalo [a,b]: *v=a:paso:b* (por defecto *paso*=1): *v= 0:0.1:1*: elementos de 0 a 1 separados 0.1
- *linspace(a,b,n)*   *n*=lonxitude

$$\log_{10} v_i = \frac{(b-an)+i(a-b)}{1-n};\ i=1,\ldots,n$$

- Vector con compoñentes logarítmicamente espaciadas: *v = logspace(a,b,n)*: *n* mostras logarítmicamente equiespaciada entre 10^a e 10^b: ($\log_{10}x_{i+1} - \log_{10}x_i=cte$ independente de *i*):

Acceso e edición dun vector

- Acceso a elementos dun vector:
  - v(1) elemento nº 1
  - v(*end*) último elemento
  - v(1:5) elementos de 1 a 5
  - v(1:2:10) elementos de 1 a 10 de 2 en 2
  - v(:) o vector completo
  - v(*1*:*end*~=*k*): o vector menos o elemento *k*-ésimo
- Adición / supresión de elementos:
  - Adición de elementos: *v* = [*v* 5 6]. Tamén: *v*=1:3; *v*(6)=9
  - Concatenación de vectores; *v*=[1 2 3];*w*=[4 5 6]; *z=[v w]* ou *z=[v*' *w*'*]*
  - Borrar elementos: *v*(5:8)=[]; *v*(*v*>2)=[]; *v*(*rem*(*v*,2)==0)=[];

Funcións con vectores (I)

- Lonxitude dun vector: *length(v)*; nº elementos: *numel(v)*
- Producto escalar de 2 vectores: *dot(v,w)*, *v***w*' ou *sum(v.***w)*
- Lectura de vector/matriz dende arquivo: *load datos.dat;* ou ben v=*load('datos.dat')*.
- Almacena en vector/matriz *datos* ou *v*. O arquivo debe conter unha matriz numérica (non *char*). Tódalas liñas coa mesma cantidade de valores.
- Suma/producto de elementos dun vector: *sum(v)*, *prod(v)*
- *min(v)* e *max(v)*: valores mínimo e máximo dun vector.

[vmax imax] = max(v): valor máximo e índice do máximo

[~,imax]=max(v): só o índice do máximo

Funcións con vectores (II)

- *sort(v): ordea un vector por orde crecente (con matrices, ordea cada columna); sort(v, 'descend')* -> orde decrecente;
- [v2,i]=sort(v): v2=vector ordeado, i=vector cos índices dos elementos de v ordeados
- *mean(v)*, *var(v)*, *std(v), median(v)*: media, varianza, desviación típica e mediana.
- *unique(v)*: elementos non repetidos de *v* ordeados (crecente).
- Invertir un vector: *flip*(1:4) -> 4 3 2 1
- Intersección entre dous vectores: *intersect*([1 2 3 4],[1 3 7]) -> [1 3]
- Unión de dous vectores: *union*([1 3],[1 2]) -> [1 2 3]

Matrices

matriz=[elem 1ª fila; ...; elems. nª fila]

- *a = [1 2 3; 4 5 6; 7 8 9]* : matriz 3x3: columnas separadas por ;

Ollo: podes escribir a(1,2) ou a(5), onde o índice incrémentase por columnas

- Tipo da matriz (nº filas e cols): *whos a*
  Name Size Bytes Class
  a 3x3 72 double array
- *eye(m, n)*: matriz identidade; *eye(n)*: cadrada identidade
- *zeros(m, n)*: matriz *mxn* con ceros; *3+zeros*(*m*,*n*): con 3s
- *ones(m, n)*: matriz *mxn* con 1s; *5*ones(m, n)*: con 5s
- *rand(m, n)*: con valores reais aleatorios en [0, 1]

  *a + (b – a)*rand(m, n)* : aleatorios en [a, b]

- *randi([m n],nf,nc)*: matriz *nf* x *nc* con valores enteiros aleatorios entre *m* e *n; randi(n,nf,nc): valores entre 1 e n*
- Inicializa xerador de números aleatorios:
  - *rng('default')* ← Sempre igual (reproducible)
  - *rng('shuffle')* ← Distinto (co tempo) (non reproducible)

Octave:
rand('*seed*',0)
rand('*seed*','*reset*')

Acceso a elementos dunha matriz e inserción / borrado de filas e columnas

- a(1,2): elemento 1ª fila e 2ª columna
- a(5): elemento 5º percorrendo por columnas
- a(1,:): elementos da 1ª fila
- a(:, 2): elementos da 2ª columna
- *a(1:2,2:3)*: [a_{12} a_{13}; a_{22} a_{23}]
- *a = rand(10,10); a(1:2:5, 2:3:10)*

$$\begin{bmatrix} a_{12} & a_{15} & a_{18} \\ a_{32} & a_{35} & a_{38} \\ a_{52} & a_{55} & a_{58} \end{bmatrix}$$

- Adición dunha fila: *a=ones(3);a=[a; [1 2 3]]*. Ou ben: a=ones(3);a(5,:)=[1 1 1]
- Adición de columna: *a=[a [1;2;3]]* ou a(:,6)=zeros(3,1)
- Supresión dunha fila: *a(1,:)=[]*
- Supresión dunha columna: *a(:,1)=[]*

Funcións con matrices (I)

- Tamano: *[nf nc] = size(a); nf = size(a,1); nc=size(a,2);*
- *numel(a)*: nº elementos de matriz; *length(a)*: *max*(*nf*,*nc*)
- Matriz diagonal cun vector: *v=[1 2 3];a= diag(v)*
- Vector coa diagonal dunha matriz: *v = diag(a)*
- *diag(diag(a))*: matriz diagonal coa diagonal de *a*
- *magic(n)*: matriz cadrada máxica de orde *n* (igual suma de filas e columnas).
- Suma de elementos <u>por columnas</u>: *sum(a)* ou *sum(a,1);* <u>*por filas*</u>: *sum(a,2)* ou *sum(a')*; <u>suma completa</u>: *sum(a(:))* ou *sum(sum(a))*
- Producto de elementos por columnas/filas: *prod(a)/prod(a,2)*
- Triángulo superior / inferior: *triu(a)* / *tril(a)*

Transformación de matriz en vector ou matriz doutra orde

- Útil para transformar unha matriz nun vector e procesar os seus elementos cun so bucle, evitando bucles dobres.
- Conversión de matriz *a* en vector fila <u>por columnas</u>: *v=a(:)’*. Se queres <u>por filas</u>: *b=a’;b*(:)’
- Función *reshape(matriz,nf,nc)*: transforma a matriz noutra de orde *nf* x *nc* <u>por columnas</u>. O número *nf* x *nc* debe ser igual ao nº de elementos de *a*.
- Se queres que sexa <u>por filas</u>: *reshape(a’,nf,nc);*
- Transformar de matriz a vector <u>por filas</u>: *v=reshape(a,1,nf * nc);* Se queres <u>por columnas</u>: *v=reshape(a’,1,nf * nc);*
- Transformar *a* -> *b* (con *n* filas): *b=reshape(a,n,[]);* o número *n* debe ser divisor do nº de elementos de *a*; o nº de columnas será o necesario para almacenar os *nf* x *nc* elementos de *a* nunha matriz *b* de *n* filas.

Repetición dunha matriz

- Función *repmat*(*a*,*n*,*m*): repite a matriz a *n* veces verticalmente e *m* veces horizontalmente
- Ex: *a=[1 2;3 4]*

  repmat(a,2,3): repite a matriz dúas veces verticalmente e 3 horizontalmente:

  1 2 1 2 1 2
  3 4 3 4 3 4
  1 2 1 2 1 2
  3 4 3 4 3 4

Operacións con matrices (I)

- Operacións por compoñentes: punto antes do operador: *a.*b, a./b, a.^b*: ambas matrices deben coincidir en nº de filas e de columnas
- Operacións matriz-escalar:
  - Suma / resta / produto / cociente con escalar: tódolos elementos da matriz se operan co escalar
  - Cociente escalar-matriz por compoñentes: $b=k./a \rightarrow b_{ij} = k/a_{ij}$
  - Potencia escalar-matriz por compoñentes: $b=k.\hat{}a \rightarrow b_{ij} = k^{a_{ij}}$
  - Potencia matriz-escalar por compoñentes: $b=a.\hat{}k \rightarrow b_{ij} = a_{ij}^{k}$
  - Potencia matriz-escalar matricial: $b=a\hat{}k$ ($a\cdot\ldots\cdot a$, a debe ser cadrada)

Operacións con matrices (II)

- Operacións entre matrices:
  - Suma *a* + *b* e resta *a* - *b*: *a* e *b* deben coincidir en nº de filas e de columnas
  - Producto matricial: *a***b*: o nº de columnas de *a* debe coincidir co nº de filas de *b*
  - Producto por compoñentes: *c=a.***b*: *a* e *b* deben coincidir en nº filas e de columnas, e $c_{ij} = a_{ij} \cdot b_{ij}$
  - División matricial a esquerda: *a\b* → $a^{-1} \cdot b$ → *pinv(a)***b*
  - División matricial a dereita: *a/b* → $a \cdot b^{-1}$ → *a***pinv(b)*
  - Cociente por compoñentes: *c=a./b* → $c_{ij} = a_{ij} / b_{ij}$
  - Potencia por compoñentes: *c=a.^b* → $c_{ij} = a_{ij}\hat{}\, b_{ij}$

Funcións con matrices (II)

- *unique(a)*: elementos non repetidos de *a* ordeados (crecente) como vector columna.
- Determinante dunha matriz cadrada: *det(a)*.
- Inversa dunha matriz cadrada: *inv(a)*, so cando $det(a) \neq 0$.
- Pseudoinversa de Moore-Penrose $a^{\dagger}$: *pinv(a)*, existe para matrices non cadradas e cadradas con $det(a)=0$.
- Autovalores dunha matriz cadrada: *v=eig(a)*
- Autovectores: *[v d] = eig(a)*

  v= matriz con autovectores de matriz *a* por columnas

  d=matriz diagonal con autovalores de matriz *x*: $det(a - d_{ii}\mathbf{1}) = 0$; $a \cdot v_i = d_{ii} v_i$ (v_i= columna *i* de *v*), $i=1,\ldots,n$

Funcións con matrices (III)

- Mínimo e máximo:
  - Por columnas: *min(a)* e *max(a)*
  - Por filas: *min(a,[],2)* ou *min(a')*, menos eficiente
  - Matriz completa: *min(a(:))* ou *min(min(a))*
- Valores mínimos/máximos e índices dos elementos mín/máx:
  - Por columnas: *[v,i]=min(a)*
  - Por filas: *[v,i]=min(a,[],2)*
  - Matriz completa: *[v,i]=min(a(:))*
  - *v*: vector con valores mínimos
  - *i*=vector con índices de elementos mínimos
- Índices de fila e columna do elemento mínimo/máximo dunha matriz:

[~,i]=min(a(:)); [f,c]=ind2sub(size(a),i)

Funcións con matrices (IV)

- Media, varianza desviación típica e mediana:
  - Por columnas: *mean(a), var(a)*, *std(a)*, *median(a)*:
  - Por filas: *mean(a,2), var(a,[],2)*, *std(a,[],2)*, *median(a,2)*
  - Matriz completa: *mean(a(:)), var(a(:)), std(a(:)), median(a(:))*
- Ordeamento:
  - Por columnas: *sort(a)*, *sort(a,'descend')*
  - Por filas: *sort(a,2), sort(a,2,'descend')*
  - Matriz completa: *sort(a(:))*, *sort(a(:),'descend')*
- Matrices dispersas (moitos elementos nulos): *a = sparse(i, j, c, m, n); full(a)*: mostra matriz; i(j) = índices de filas (columnas) de elementos non nulos; c= vector con valores de elementos non nulos; m(n)= nº filas(columnas)

Sistema de ecuacións lineais

- Sexa o sistema en forma matricial **b** = **Ax**, con *n* ecuacións e *n* incógnitas

- Resolución en Matlab:
  $a = [a_{11} \ldots a_{1n}; \ldots; a_{n1} \ldots a_{nn}];$
  $b = [b_1; \ldots; b_n];$
  rank(a)
  rank([a b])
  x = a\b % se rank(a)==rank([a b])
  *x = inv(a)*b % alternativa*

rango dunha matriz: se *rango(a)==n*, o sistema é compatíbel determinado; se *m=rango(a)<n*, o sistema é compatíbel indeterminado (se *rango([a b])=m*) ou incompatíbel (se *rango([a b])≠m*)

- Se o sistema é **incompatíbel**, a pseudoinversa *x=pinv(a)*b* permite atopar unha solución de norma mínima, é dicir, *norm(a*x-b)* é mínima, aínda que ≠0 e pode ser elevada

$$\begin{aligned} x+2y+3z&=0 \\ 2x+4y+6z&=5 \\ 3x+6y+9z&=2 \end{aligned}$$

rank(*a*)=1
rank([*a b*])=3

Non existe **x** con **Ax**-**b**=**0**
$\mathbf{x}=\mathbf{A}^{+}\mathbf{b}$ verifica |**Ax**-**b**| é mínima

Sistema compatíbel indeterminado (I)

- As infinitas solucións pódense escribir como unha **solución individual do sistema** *(x0)* máis unha **combinación linear de solucións do sistema homoxéneo** *(k·c)* asociado.
- **Solución individual**: *x0=pinv(a)*b*. Podes comprobar que é solución calculando *norm(A*x0-b)*.
- **Solucións do sistema homoxéneo**: cerne da aplicación linear asociada á matriz dos coeficientes: *k=null(a)* retorna unha matriz onde cada columna é un vector dunha base ortonormal deste subespazo linear (nulo).
- **Solución xeral** do sistema indeterminado: *x0+k*c*, sendo *c* o vector de coeficientes da combinación linear (p.ex. *c=ones(r,1)*, sendo *r=size(k,2)* a dimensión do espazo solución (nº columnas de *k*).

Sistema compatíbel indeterminado (II)

- Sistema *ax=b c*on *a=[1 2 3;4 5 6;7 8 9]; b=[1;2;3]*
- *rank(a)=2,rank(b)=2<3: s*istema compatíbel indeterminado
- A solución ten dimensión 3-2=1 (é unha recta en $\mathbb{R}^3$)
- *x0=pinv(a)*b* : solución individual
- *k=null(a)*: *k*=vector columna director da recta
- Solucións da forma *x=x0+c*k* onde *c*=escalar
- O vector *x* é solución porque *norm(a*x-b)* $\simeq$ 0

Exercicios

1) Define un vector x con 10 compoñentes espaciadas logarítmicamente entre 1 e 100; suprímelle as compoñentes 3-5; engádelle o vector [3 4 5] polo comezo; selecciona os elementos de índice múltiplo de 3; calcula a lonxitude de x

2) Calcula a suma, producto, máximo e mínimo, media e desviación típica do vector x do exercicio anterior

3) Crea co editor de Matlab un arquivo de *datos.dat*. Cárgao en Matlab ao vector x e representa as dúas columnas de x

```
1 3
2 4
3 3
4 5
5 4
```

4) Define os vectores (1,2,3,4,5) e (5,4,3,2,1) e calcula o seu producto escalar

Exercicios

5) Crea unha matriz 3x3 con elementos=1 e outra 2x2 con elementos=5. Logo pégaas e obtén a seguinte matriz:

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 0 & 0 & 5 & 5 \end{bmatrix}$$

6) Define unha matriz de orde 3x4 con números aleatorios no intervalo $[-1, 1]$:

7) Dada unha matriz A cadrada de orde 5: selecciona a submatriz de A coas filas 2-3 e as columnas 1-3; amplía a matriz engadíndolle unha fila ao comezo da matriz; bórralle as filas 1 e 4

8) Dadas as matrices A e B:

$$A=\begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} \qquad B=\begin{bmatrix} 1 & 4 & 5 \\ -3 & 2 & 1 \\ 0 & 5 & 4 \end{bmatrix}$$

calcular $A \cdot B$, $A^{-1} \cdot B$, $A \cdot B^{-1}$, $|A|$, suma,min e max por columnas de A; triángulo superior e inferior de B

Exercicios

9) Define unha matriz dispersa 10x10 con valores non nulos (8,1)=-3 e (3,4)=-9

10) Define unha matriz 5x5 con valores aleatorios no intervalo [-3,1]

11) Partindo da matriz identidade 7x7 e usando o : obtén a seguinte matriz:

$$\begin{bmatrix} 2 & 2 & 2 & 0 & 5 & 5 & 5 \\ 2 & 2 & 2 & 0 & 5 & 5 & 5 \\ 3 & 3 & 3 & 0 & 5 & 5 & 5 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 4 & 4 & 7 & 0 & 9 & 9 & 9 \\ 4 & 4 & 7 & 0 & 9 & 9 & 9 \\ 4 & 4 & 7 & 0 & 9 & 9 & 9 \end{bmatrix}$$

12) Crea unha matriz 5x7 coa 1ª fila 1 2 3 4 5 6 7, 2º fila 8 9 10 11 12 13 14, 3ª fila 15-21, etc. A partir dela crea outra matriz 3x4 coas filas 2-4 e columnas 3-6 da matriz orixinal

13) Resolve este sistema de ecuacións:

$$\begin{array}{c} -44x+10y+16z=20 \\ 10x-43y+6z+12t=0 \\ 16x+6y-30z+8t=12 \\ 12y+8z-34t=-40 \end{array}$$

Solucións aos exercicios (I)

1) *x=logspace(1,2,10);x(3:5)=[];x=[3 4 5 x];
x(3:3:10);length(x) ou size(x,2)*

2) *sum(x);prod(x);max(x);min(x);mean(x);
std(x)*

3) *x=load('datos.dat');plot(x(:,1),x(:,2),'o-')*

4) *x=[1 2 3 4 5]; y=[5 4 3 2 1];dot(x,y);x*y'*

5) *a=ones(3,3);b=5*ones(2);[[a zeros(3,2)] ;
[zeros(2,3) b]]*

6) *a=-1+2*rand(3,4)*

7) *a=magic(5);a(2:3,1:3);a=[ones(1,5); a];a(1:3:4,:)=[];*

Solucións aos exercicios (II)

*8)a=[1 2 3; 0 1 2; 0 0 1]; b=[1 4 5; -3 2 1; 0 5 4];a*b; a\b ou inv(a)*b; a/b ou a*inv(b); det(a);sum(a);min(a);max(a);a-tril(a);a-triu(a)*

9)a=sparse([8 3],[1 4],[-3 -9],10,10); full(a)

*10)a=-3+4*rand(5)*

11)a=eye(7);a(1:2,1:3)=2; a(3,1:3)=3; a(1:3,5:7)=5; a(5:7,1:2)=4; a(5:7,3)=7; a(5:7,5:7)=9

12)a=zeros(5,7);x=1;for i=1:5; for j=1:7; a(i,j)=x; x=1+1;end; end; b=a(2:4,3:6)

13)a = [-44 10 16 0; 10 -43 6 12; 16 6 -30 8; 0 12 8 -34]; b = [20;0;12;40];a\b