

Hoja de consulta de MatLab - Pedro Fortuny Ayuso - Uniovi - 22 de enero de 2022

BÁSICO

`[a:s:b]` — Vector de a a b de s en s .
`linspace(a,b,n)` — Vector de n números equidistribuidos entre a y b .
`pi`, `exp(1)` — Valor de π y de e (3.14159... y 2.71828...).
`clear`, `clear(x)` — Limpia todas las variables o solo la x .
`whos`, `who` — Describe o lista las variables definidas.

OPERACIONES ARITMÉTICAS Y LÓGICAS

`+` `-` `*` `/` — Operaciones básicas con números y matrices.
`^` — Elevar a una potencia: $a^b = a^b$.
`.*` `./` `.^` — Multiplicar, dividir y elevar *elemento a elemento*.
`&` `|` `~` — Operadores lógicos **y** (`&`), **o** (`|`), **negación** (`~`).
`all`, `any` — Verifica si *todos* (`all`) o *alguno de* (`any`) los elementos de un vector cumplen una condición: `any(x>0)` ¿alguno de los elementos de x es > 0 ?

MATRICES Y VECTORES

`linspace(a,b,n)` — Vector de n elementos equidistribuidos entre a y b .
`[a:s:b]` — Vector de números de a hasta b yendo de s en s .
`length(v)` — Longitud del vector v .
`size(M)` — Tamaño (`filas` \times `columnas`) de M .
`zeros(n,m)` — Matriz llena de 0s de n filas y m columnas. Si m no está, matriz cuadrada $n \times n$.
`ones(n,m)` — Matriz llena de 1s de n filas y m columnas.
`eye(n)` — Matriz cuadrada identidad de tamaño n .
`eye(n,m)` — Matriz $n \times m$ en la que la matriz $n \times n$ más a la izquierda es la identidad y el resto son ceros.
`diag(v)` — Matriz cuadrada cuya diagonal es el vector v .
`diag(v,n)` — Matriz cuadrada cuya diagonal n es el vector v . La diagonal 0 es la principal. Si $n < 0$, se cuenta debajo de la diag. principal, si $n > 0$, encima.
`rand(n,m)` — Matriz de números aleatorios equidistribuidos entre 0 y 1 de n filas y m columnas. Si m no está, matriz cuadrada.

ACCESO A ELEMENTOS

Si x es un vector

`x(i)` — El elemento i de x .
`x(i:j)` — Elementos desde el i hasta el j del vector x .
`x(i:end)` — Elementos desde el i hasta el final del vector x .
`x(:)` — Todos los elementos de x .

Si A es una matriz

`A(i,j)` — El elemento i, j de A .
`A(a:b,c:d)` — Los elementos de las filas a hasta la b que están en las columnas c hasta la d (es una submatriz).
`diag(A)` — Los elementos de la diagonal principal de A .

FUNCIONES Y UTILIDADES

`sqrt` — Raíz cuadrada. `sqrt(x)`: raíz cuadrada de cada elemento de x .
`sin`, `cos`, `tan` — Seno, coseno y tangente. P.ej. `sin(x)`, seno de cada elemento de x .
`asin`, `acos`, `atan` — Funciones inversas de las anteriores (arco seno, etc.)
`exp` — Exponencial. `exp(x)`: exponencial de *cada elemento de x*.
`log`, `log10` — Logaritmo natural (base e) y decimal (base 10). `log(x)`: logaritmo natural de cada elemento de x .

DEFINICIÓN DE FUNCIONES

`f = @(x) sin(x) - exp(x) .* x.^2` — Define la función f , de una variable, que vale $\sin(x) - e^x x^2$ (Úsense siempre `.*`, `./` y `.^`).
`f = @(x,y,z) x .* y - z.^2 .* y ./ x` — Define la función f de 3 variables que vale $xy - z^2 y/x$. (Úsense siempre `.*`, `./` y `.^`).
`function [z] = patata(a, b, c) end` — En el fichero `patata.m`, definición de la función `patata`, que devuelve un valor z y requiere 3 parámetros (a , b y c).
`function [Sol N M] = pototo(a, b, c) end` — En el fichero `pototo.m`, definición de la función `pototo`, que devuelve tres valores (`Sol`, `N` y `M`) y requiere tres parámetros (a , b y c).

DIBUJOS

`clf` — Limpia la ventana gráfica.
`hold on/hold off` — Permite (`hold on`) o impide (`hold off`) dibujar sobre la misma gráfica.
`plot(y)` — Si y es un vector, dibuja la secuencia de valores de y .
`plot(x,y)` — Si x e y son vectores de la misma longitud, dibuja la gráfica de x frente a y dada por esos valores.
`plot(x,f(x))` — Si x es un vector y f una función, dibuja los puntos $(x, f(x))$.
`fplot(f,[x0 x1 y0 y1])` — Si f es una función, dibuja f en el rectángulo $[x0, x1] \times [y0, y1]$.
`axis([x0 x1 y0 y1])` — Redibuja una gráfica en el rectángulo $[x0, x1] \times [y0, y1]$.
`xlim([x0 x1]), ylim([y0 y1])` — Redibuja una gráfica con la variable x entre $x0$ y $x1$ (o con y entre $y0$ y $y1$).

Hoja de consulta de MatLab - Pedro Fortuny Ayuso - Uniovi - 22 de enero de 2022

PROGRAMACIÓN

return — Vuelve *inmediatamente* de una función.

```
if(COND)
    ...
end
```

— Si se cumple COND, se hacen las instrucciones

```
if(COND)
    ...1
else
    ...2
end
```

— Si se cumple COND, se hace ...1; si no, se hace ...2.

```
for k = a:s:b
    ...
end
```

— Bucle desde a hasta b de s en s, en la variable k.

```
for k = v
    ...
end
```

— Si v es un vector, bucle en la variable k que recorre los elementos de v.

```
while(COND)
    ...
end
```

— Repetir las instrucciones de ... mientras COND sea verdad.

Si nombreF.m es un fichero .m

```
function [x1 x2] = nombreF(a,b,c)
    ...
end
```

Define la función nombreF, que — recibe como entrada a, b y c y devuelve como salida x1 y x2.

OPERACIONES MATRICIALES AVANZADAS

hilb(n) — Matriz de Hilbert de orden n. Cada entrada (i, j) vale $1/(i + j)$.

rref(M) — Matriz *reducida escalonada* asociada a M.

cond(M) — Número de condición de M.

inv(M) — Matriz inversa *de una matriz cuadrada* (si es que tiene).

**** — Resolución de un sistema lineal: $A \setminus b$ resuelve el sistema $Ax = b$.

[A B] = lu(M) — Factorización LU de la matriz M: Se tiene que $AB = M$ y que A es triangular inferior y B triangular superior.

chol(A) — Factor de Cholesky de A (para A simétrica y definida positiva). Una matriz R tal que $R'R=A$.

OPERACIONES CON POLINOMIOS

Se supone que v es un vector $v=[v_1, \dots, v_n]$.

polyval(v, x) — Valor de la expresión polinomial $v_1x^{n-1} + \dots + v_{n-1}x + v_n$ (el grado *decrece*).

roots(v) — Raíces del polinomio $v_1x^{n-1} + \dots + v_{n-1}x + v_n$.

polyderiv(v) — Polinomio (en forma de vector) derivada de $v_1x^{n-1} + \dots + v_{n-1}x + v_n$, es decir, $[v_1/(n-1), \dots, v_{n-2}/2, v_{n-1}]$.

OPERACIONES NUMÉRICAS

Se supone que v es un vector $v=[v_1, \dots, v_n]$.

sum(v), prod(v) — Suma y producto de todos los elementos de v.

max(v), min(v) — Máximo y mínimo de los elementos de v.

mean(v), var(v) — Media y varianza de los elementos de v.

diff(v) — Vector de *diferencias sucesivas* de v. Es decir, $[v_2-v_1, v_3-v_2, \dots, v_n-v_{n-1}]$. Se tiene $\text{length}(\text{diff}(v))=\text{length}(v)-1$.

diff(v,k) — Vector de *k-diferencias sucesivas* de v. Se tiene $\text{length}(\text{diff}(v,k))=\text{length}(v)-k$.

INTERPOLACIÓN

interp1(x,y) — Dados dos vectores x e y de misma longitud n, devuelve el polinomio interpolador de los puntos (x, y) correspondientes. Se devuelve un vector $[v_1, \dots, v_n]$ (que se interpreta como polinomio de grado $n - 1$).

polyfit(x,y,n) — Devuelve el polinomio (como vector) de grado n que minimiza el error cuadrático respecto de la nube dada por x e y (vectores de la misma longitud).

spline(x,y) — Interpolante de spline cúbico por los puntos (x, y) dados por x e y (vectores de la misma longitud). Devuelve un polinomio “a trozos” que se puede evaluar con **ppval**.