# **Octave**

## Carga de paquete simbolico

**pkg load symbolic**

**syms x**

## Comandos de axuda

**doc, help, letras y tabulador** para ver que comandos tienen esas letras

## Comandos útiles

Usamos **;** al final para realizar una operación pero no mostrarla

## Comandos de limpeza

**clc, home, clear all, clear** \*variable\*

## Formatos

**format short, format rat, format long, format long e**

## Operaciones

**+**, **-**, **\***, **/**, **^**,

## Funciones

**sqrt(n)** raíz cuadraza, **nthroot(x,n)**, **exp(n)** exponencial con base e de su argumento, **abs(n)** valor absoluto, **log(n)** logaritmo neperiano,

**log10(n)** log base 10; **sin(n), cos(n), tan(n), sec(n), csc(n), cot(n), asin(n), acos(n), atan(n), asec(n), acsc(n), acot(n), factor(n)** divide un numero factorialmente**,** **factorial(n)**

## Aproximar valor a unha fraccion

fraccion = **rats(n)**

## Partes enteiras e redondeo

**fix(1.5)** quita decimales, **floor(1.5)** aproxima al valor más pequeño, **round(1.5)** redondea

## Vector de ceros

vector\_ceros = zeros(filas,columnas)

**vector = [1 2 3 4 5]** %en fila%

**vector = [1; 2; 3; 4; 5];** %en columna%

**vector = [1 2 3 4 5]';** %en columna%

## Mostrar elementos dun vector

**vector(2)**

**vector([1,3])**

**vector([1:3])**

**zeros(a,b)**

a:b:c

a:b

% Crear vector con saltos

vector1 = linspace(1,10,5) % 5 numeros do 1 ao 10

vector2 = [1:2:10] % Desde o 1 ata o 10 sumando 2

% Lonxitude dun vector

longitud\_vector = length(vector)

% Operacions con vectores

v = vector.^5

v = vector(5).\*5

v = v./vector

v = vector + [1 2 3 4 5]

v = vector + 5

suma = suma(vector)

producto = prod(vector)

maximo = max(vector)

minimo = min(vector)

% Arquivo de funcion (co nome funcion)

pkg load symbolic

syms x

function y = funcion(x) % Chamar a funcion utilizando funcion(1)

y = cos(4\*x) + sin(2\*x)

endfunction

% Constantes simbolicas

pi = sym('pi')

e = sym('E')

% Constante simbolica a valor numerico

alpha = sym('pi')

beta = double(alpha)

vpa(1/3,5) % Aproximacion con polo menos 5 decimais

% Operar con expresions simbolicas

funcion = x^2 + 5;

findsym(funcion) % Mostra a variable simbolica da funcion

solve(funcion == 0, x) % Raices da funcion

[x,y] = solve(2\*x + y == 0, x + y == 3) % Sistema de ecuacions

subs(funcion,x,0) % Resolve a funcion para x = 0

% Exemplo con varias variables

syms a b

funcion = a\*x^2 + 5\*b\*x;

subs(funcion,{a,b},{1,2}) % Substitue as variables a e b por 1 e 2

% Textos

disp('Este texto se muestra en pantalla')

disp(' ') % Lina en blanco

a = 5;

disp (['El resultado es', num2str(a)])%Mostra texto cun numero cambiado a cadena

error('Error')

fprintf('El resultado es %8.3f', a)

fprintf('\n') % Salto de lina

variable = input('Introduzca un numero : ')

cadena = input('Introduzca su nombre : ','s')

opcion = menu('MENU','Introducir nombre','Introducir edad')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% GRAFICAS %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

pkg load symbolic

syms x

% Graficar poligonal que une varios puntos (x,y)

x = [1 3 5];

y = [2 1 4];

plot(x,y)

% Dotar de informacion a grafica

title('Titulo grafica')

xlabel('Eje de abscisas')

ylabel('Eje de ordenadas')

text(1,2, 'Este es el punto (1,2)')

legend('texto linea')

axis equal % Misma escala en ambos eixos

vector\_ejes = axis % Formados polos valores minimos e maximos de cada eixo

axis([1 10 1 10]) % Modifica os valores minimos e maximos de cada eixo