CURSOS INTERSEMESTRALES

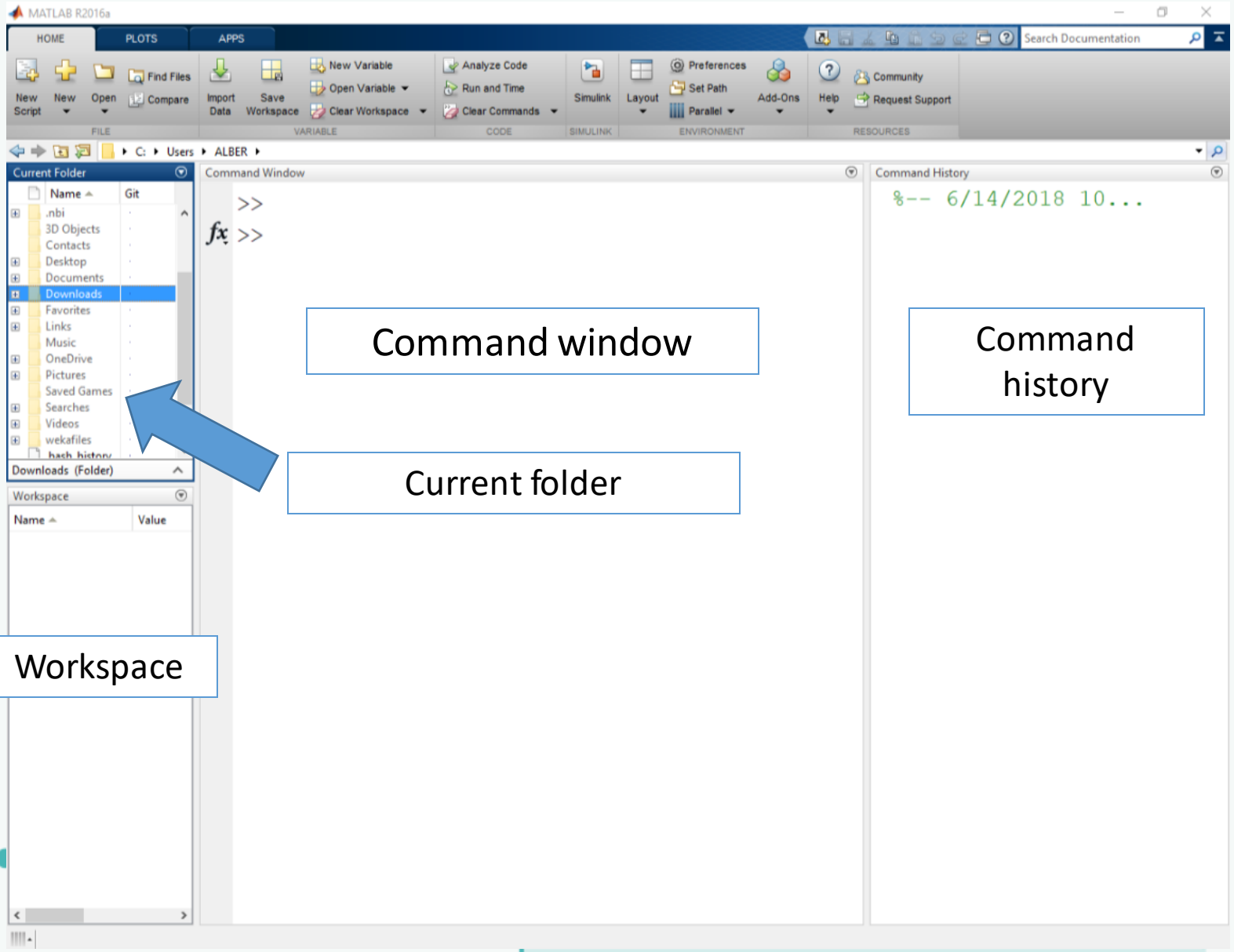
PROTECO

MATLAB

Introducción

MATLAB básico 2019-1

Entorno MATLAB



PROTECO

Introducción

MATLAB es la abreviatura de "matrix laboratory".

Está diseñado para funcionar principalmente con matrices y arreglos completos.

Todas las variables de MATLAB son *arreglos* multidimensionales, sin importar el tipo de datos. Una *matriz* es un arreglo bidimensional.

PROTECO

Operadores aritméticos

Operación	Símbolo	Ejemplo
Suma	+	5 + 4
Resta	-	5 - 4
Multiplicación	*	5 * 4
División derecha	/	5 / 4
División izquierda	\	5 \ 3 = 3 / 5
Exponenciación	^	5 ^ 3

PROTECO

Orden de precedencia

Precedencia	Operación
Primero	Paréntesis, el más interno se ejecuta primero
Segundo	Exponenciación
Tercero	Multiplicación y división
Cuarto	Suma y resta

PROTECO

Formato de impresión en pantalla

Formato	Descripción	Ejemplo usando pi
short (default)	Punto fijo con 4 decimales	3.1416
long	Punto fijo con 15 decimales	3.141592653589793
shortE	Notacion cientifica, 4 dec	3.1416e+00
longE	Notación científica, 15 dec	3.141592653589793+00
shortG	5 digitos	3.1416
longG	15 digitos	3.14159265358979
shortEng	Notación ingenieril exp multiplo de 3, 4 decimales	3.1416e+000
longEng	Notación ingenieril, 15 dec	3.14159265358979e+000
+	Indica si es positivo o negativo	+
bank	Divisas, 2 decimales	3.14
hex	Hexadecimal	400921fb54442d18
rat	Relación entre números enteros pequeños	355/113

PROTECO

Funciones trigonométricas básicas

Función	Descripción
sin(x)	Seno del ángulo, x en radianes
cos(x)	Coseno del ángulo, x en radianes
tan(x)	Tangente del ángulo, x en radianes
cot(x)	Cotangente del ángulo, e en radianes

PROTECO

Funciones de redondeo

Round: redondea al entero más próximo (antes de 2014). Puede recibir distintos argumentos:

```
>>round(pi)                          3
>>round(pi,3)                        3.1420
>>round(3312,2,'significant')        3300
```

Fix: redondea hacia el cero

```
>> fix([-1.9 -3.4 5.8 9.999])
ans =
     -1    -3     5     9
```

PROTECO

Funciones de redondeo

Ceil: redondea hacia el infinito, ejemplo:

>> ceil(45.31)

ans = 46

Floor: redondea hacia menos infinito

>>ceil(-35.6)

ans = -36

PROTECO

Variables especiales

Variable	Valor
pi	Número pi
eps	Mínima distancia entre dos números
inf	Un número muy muy muy grande
NaN	Not a Number
ans	Resultado de lo último tecleado en command window
i	Raíz cuadrada de -1
j	Igual que i

PROTECO

Crear una variable

Para crear una variable debemos de asignarle un nombre representativo, para ello nos situamos sobre la línea de comandos (>>) y tecleamos lo siguiente:

>> a =45 % nombre de variable "a" valor: 45

%Inicia respuesta del intérprete de comandos

a =

45

%Termina respuesta de intérprete de comandos

>> b=36; % notar que en este caso no existe una respuesta

PROTECO

Matrices y arreglos

Creación de arreglos

Para crear un arreglo debemos de separar los elementos con una coma, o un espacio, dentro de dos corchetes, ejemplo:

>>A = [1, 2, 3, 4]

También válido:

>>A = [1 2 3 4]

PROTECO

Matrices y arreglos

Creación de matrices

Para crear una matriz con varias filas debemos separar las filas con punto y coma o solo presionar enter para crear una nueva fila, ejemplo:

>>A= [1 2 3 ; 4 5 6; 7 8 9]

También válido:

>>A= [1 2 3
4 5 6
7 8 9]

PROTECO

Concatenación

Podemos crear nuevas variables ya sean arreglos o matrices a partir de ya existentes, para ello usaremos la concatenación, ejemplo:

>>c=magic(3);

>>a=[c,[456 789 ;859 896;654 784]]

a =

8	1	6	456	789
3	5	7	859	896
4	9	2	654	784

PROTECO

Operador : (dos puntos)

Con este operador podemos hacer múltiples cosas, entre ellas crear arreglos ejemplo:

>>a=[23:32]

a =

23 24 25 26 27 28 29

Podemos definir el incremento, la sintáxis es la siguiente:

Nombre_vector = [inicio : incremento : fin];
Ejemplo:

>> a= [23:3:29]

a =

23 26 29

PROTECO

Función linspace()

Esta función nos regresa un arreglo con elementos equidistantes, debemos de especificar 3 argumentos: inicio, termino y numero de elementos. Ejemplo:

>>a=linspace(4,28,9)

a =

4 7 10 13 16 19 22 25 28

Si se omite el número de elementos ,que en este caso es 9, la cantidad de elementos será 100

PROTECO

Operador transposición ' (comilla simple)

Este operador nos permite transponer un arreglo, ejemplos:

>>a=[1:3];

>>a'

ans =

1

2

3

>>b=[1 2 3; 4 5 6];

>>b'

ans =

1 4

2 5

3 6

PROTECO

Indexación de arreglos

Específica: se especifica el renglón y columna del elemento deseado, ejemplo:

>>a=magic(4);

>>a(2,3)

ans =

10

a =

16	2	3	13
5	11	10	8
9	7	6	12
4	14	15	1

Lineal: se requiere solamente de un índice, la numeración comienza en 1 y se hace en función de las columnas, ejemplo:

>>a=magic(4);

>>a(8)

ans =

14

PROTECO

Indexación de arreglos

Con operador dos puntos:

>>a=magic(4);

>>a(:,2)

```
ans =
     2
    11
     7
    14
```

>> a(:,2:3)

```
ans =
     2     3
    11    10
     7     6
    14    15
```

```
a =
    16     2     3    13
     5    11    10     8
     9     7     6    12
     4    14    15     1
```

>> a(1:2,3:4)

```
ans =
     3    13
    10     8
```

PROTECO

Operadores relacionales

Operador	Funcion equivalente	Descripción
<	lt	Menor que
<=	le	Menor igual que
>	gt	Mayor que
>=	ge	Mayor igual que
==	eq	Igual que
~=	ne	Diferente qe

El resultado de ocupar un operador relacional será una variable de carácter lógico, es decir un 0(false) ó 1(true).

PROTECO

Indexación de arreglos

Lógica: podemos hacer uso de los operadores relacionales para acceder a elementos que cumplan una condición, ejemplo

```
>>a=[1:10];
>>a(a>5)
ans =
    6    7    8    9   10
```

Pero veamos el resultado de hacer esta operación:

```
>>a>5
ans =
    0    0    0    0    0    1    1    1    1    1
```

PROTECO

Strings (cadenas)

Las cadenas son arreglos de caracteres, se utilizan para visualizar mensajes de texto como salida de alguna función, como argumento de función.

Ejemplo:

```
>>a='Hola Mundo';
>>a(4)
ans =
a
```

PROTECO

Matriz de cadenas

Para poder hacer una matriz de caracteres es necesario que cada renglón tenga el mismo número de elementos de lo contrario no se podrá construir la matriz, ejemplo:

>>datos=['Luis' ;'21' ;'Quimica']

Dimensions of matrices being concatenated are not consistent.

Para esto existe una función llamada char(), ejemplo:

>>datos=char('Luis','21','Quimica')

datos=

Luis

21

Quimica

PROTECO

Operaciones matriciales

Suma y resta:

Las matrices deben ser del mismo tamaño, o si el segundo operando es un escalar, este se sumará a cada uno de los elementos:

```
>>A=[8 5 4]; B=[10 2 7];
>>C=A+B
C =
    18     7    11
>> C-5
ans =
    13     2     6
```

PROTECO

Operaciones matriciales

Multiplicación:

La operación se ejucta de acuerdo con las reglas del álgebra ineal, por lo que la operación A*B se ejecuta si solamente el número de columnas de A es igual al número de filas de B, ejemplo:

```
>> A=[1 2 3; 4 5 6; 7 8 9]; B=[9 8 ;7 6; 5 4];
>> C=A*B
C =
    38    32
   101    86
   164   140
```

Si el segundo operando es un escalar, este se multiplicará por cada uno de los elementos

PROTECO

Operaciones matriciales

División: Como vimos anteriormente existe la división por la derecha y por la izquierda, ambas se ocupan para resolver sistemas de ecuaciones lineales del tipo AX=B(div derecha) y XC=D (div izquierda), ejemplo:

$$4x - 2y + 6z = 8$$
$$2x + 8y + 2z = 4$$
$$6x + 10y + 3z = 0$$

```
>>A = [4 –2 6; 2 8 2; 6 10 3]; B=[8;4;0];
>>X=A\B
X =
   -1.8049
    0.2927
    2.6341
```

PROTECO

Operaciones elemento a elemento

En este caso la operación se realiza elemento a elemento, ejemplo:

```
>> A=[2 6 3; 5 8 4]; B=[1 4 10; 3 2 7];
>> A.*B

ans =

 2   24   30
15   16   28
```

Símbolo	Descripción	Símbolo	Descripción
.*	Multiplicación	./	División derecha
.^	Exponenciación	.\	División izquierda

PROTECO