2020

Duván Mejia

UNIVERSIDAD SURCOLOMBIANA

25-11-2020

Manual MATLAB



Tabla de contenido

[Clase semana 1: 2](#_Toc58962327)

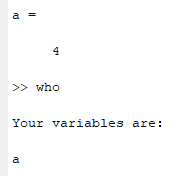
[Clase semana 2 8](#_Toc58962328)

[Clase semana 3 13](#_Toc58962329)

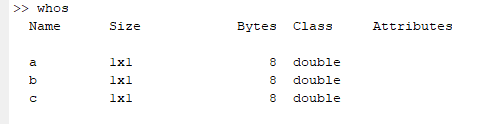
[Clase Semana 4 21](#_Toc58962330)

# **Clase semana 1:**

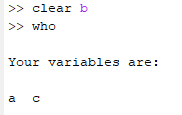
**who:** Aquí podemos observar las variables que tengo.



**whos:** aquí vemos todas las características de las variables.



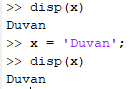
**Clear + Variable**: Borramos la variable seleccionada.



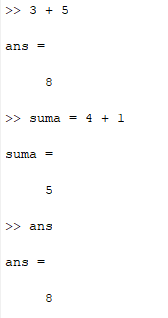
**Clear all**: borramos todas las variables.



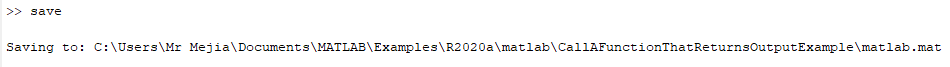
**disp**: Método para Imprimir una variable.



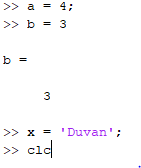
**ans:** Variable temp Operación del dato anterior.



**Save**: guardar archivo.mat y muestra la ruta



**cls** : limpia la pantalla.

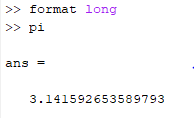




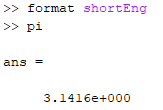
**Load:** cargar el archivo Matlab



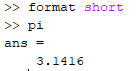
**format –long :** Formato de decimal largo



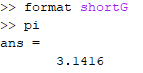
**format -shortEng:** Formato de Ingeniería



**format -short :** Formato de decimales corto



**format- shortG**: Formato más pequeño de decimales



**Eps**: Es el valor mas pequeño y equivalente a eps(1.0) y eps('doble').



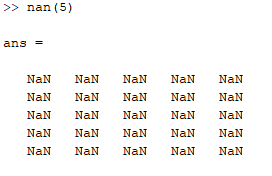
**Pi:** función matematica



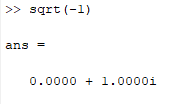
**Inf**: Variable infinita resultante de la división entre Zero



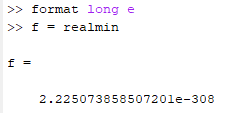
**NaN:** Cuando la variable no tiene ningún dato o indeterminado.



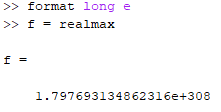
**Compleja(i+j):** utilizamos con valores complejos.



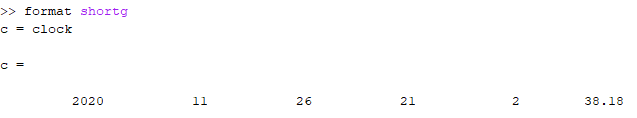
**Realmin:** más pequeño a utilizar.



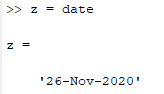
**Realmax:** valor más grande a utilizar.



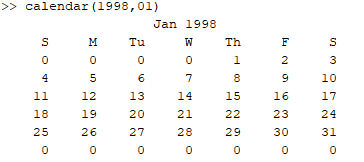
**Clock:** formato de fecha.



**Date:** formato de fecha

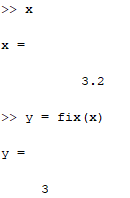


**Calendar:** variable del calendario

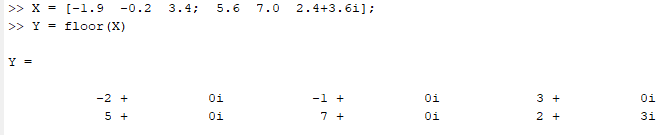


**Funciones de Aproximación**

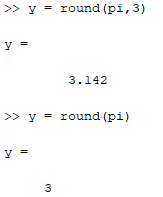
**fix:** Redondea el valor más cercano a cero.



**Floor:** Valor negativo hacia infinito.



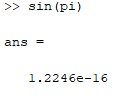
**Round:** Redondea al valor decimal o entero infinito más cercano.



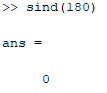
# Clase semana 2

Trigonometría

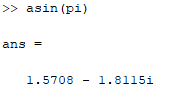
**Sin:** Seno dado en radianes.



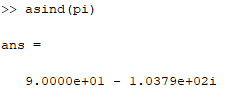
**Sind**: Seno dado en grados.



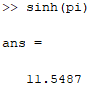
**asin:** seno inverso en radianes.



**Asind:** seno inverso en grados.



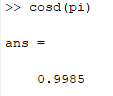
**Sinh:** seno hiperbólico.



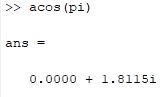
**Cos:** Coseno en radianes.



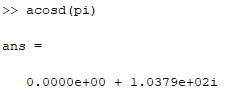
**Cosd:** coseno en grados.



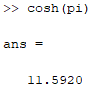
**Acos:** inverso de coseno en radianes.



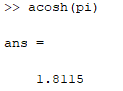
**Acosd:** inverso de coseno en grados.



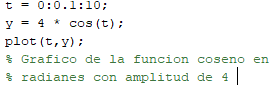
**Cosh:** coseno hiperbólico en radianes.

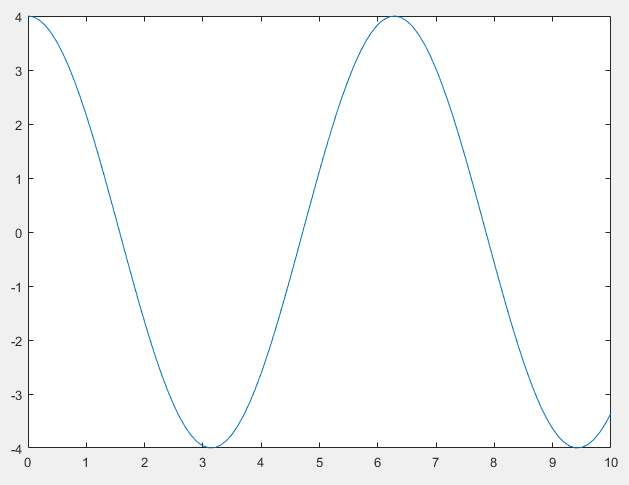


**Acosh:** coseno hiperbólico inverso en radianes.

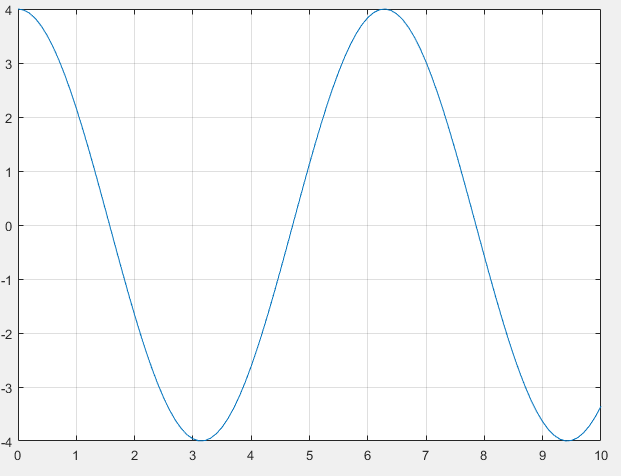


**Plot:** Graficar funciones.

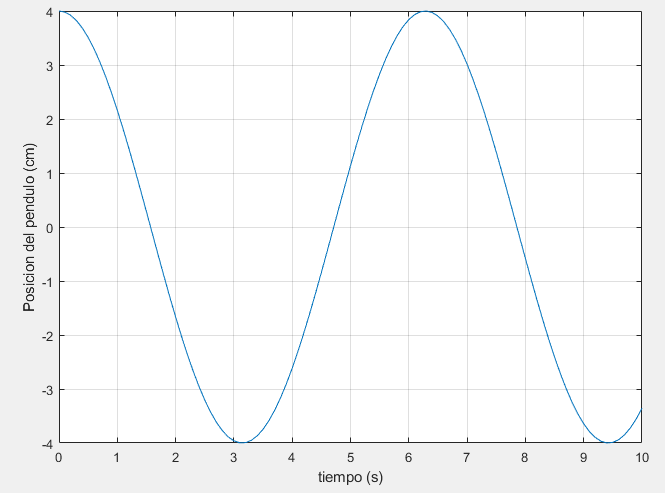




**Grid** – **on - off:** Activar cuadricula o desactivar cuadricula

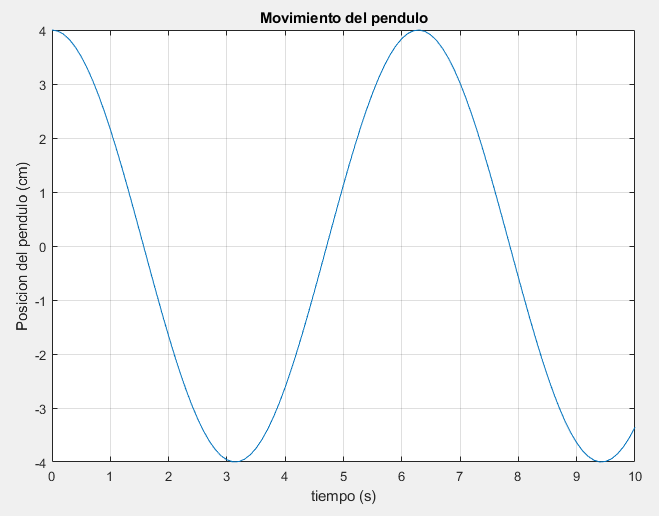


**Label :** Etiqueta de clase.



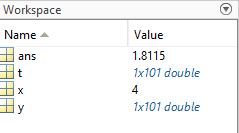
**Title:** Titulo de la gráfica.



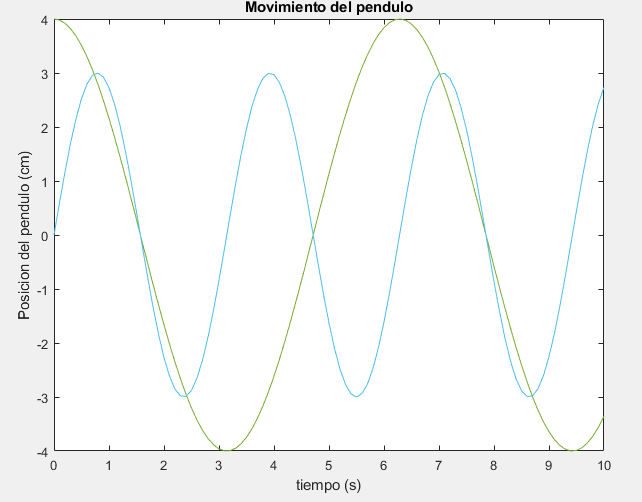


**Workspace:** Muestra la venta de trabajo donde están las variables

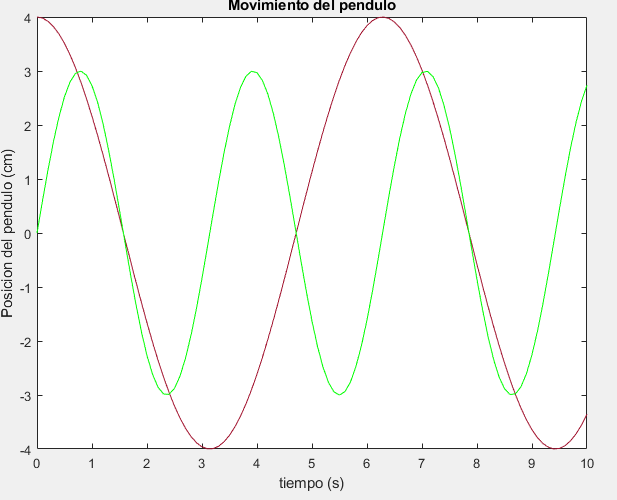




**Hold on :** Muestro dos graficas distintas en una misma figura.



**Paleta de colores:** se selecciona la letra del color inicial para cambiar en la gráfica.

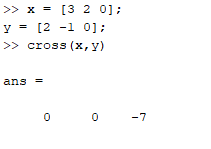


**Subplot:** Graficar subfunciones.

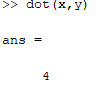


# **Clase semana 3**

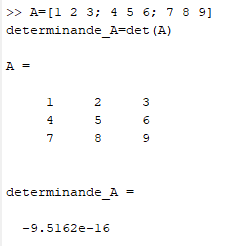
**Cross:** Producto cruz entre dos vectores.



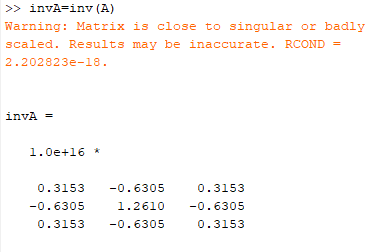
**dot:** Producto escalar entre dos vectores.



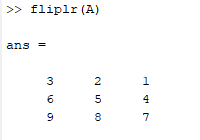
**det:** Hallar la determinante.



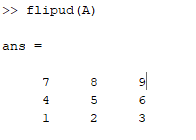
**inv:** Hallar la matriz inversa



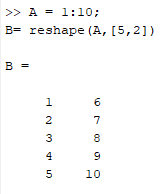
**fliplr:** devuelve A con sus columnas volteadas en la dirección izquierda-derecha



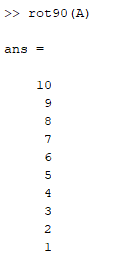
**flipud:** Devuelve las filas en dirección arriba a abajo, alrededor del eje horizontal.



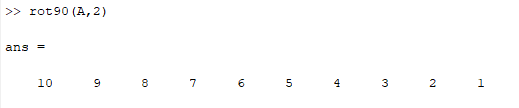
**reshape:** reformula A utilizando el vector de tamaño, sz, para definir size(B)



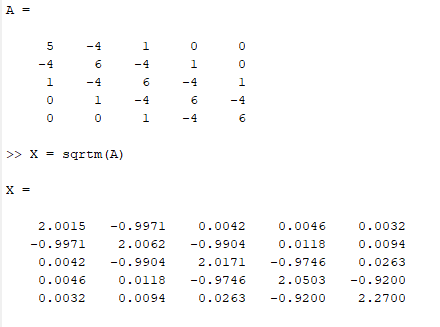
**rot90:** gira la matriz A en sentido contrario a las agujas del reloj en 90 grados.



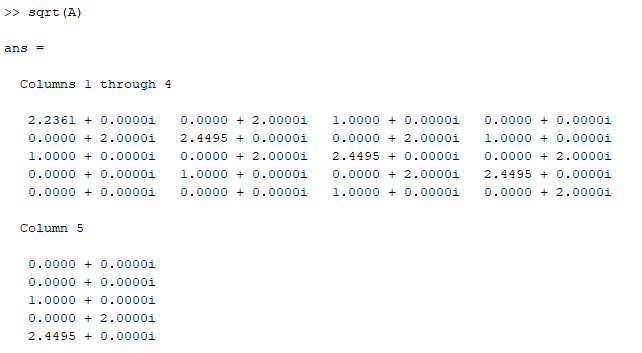
**Rot90 + n:** gira la matriz A n veces en el sentido contrario al reloj.



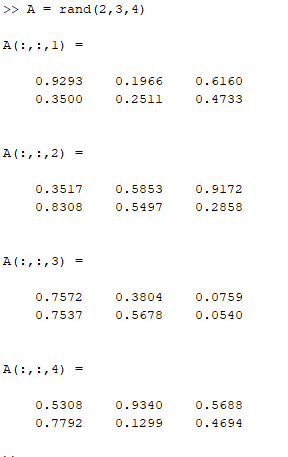
**sqrtm:** devuelve la raíz cuadrada principal de la matriz A.



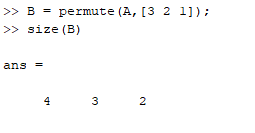
**sqrt:** devuelve la raíz cuadrada de cada elemento de la matriz X. Para los elementos de X que son negativos o complejos, sqrt(X) produce resultados complejos.



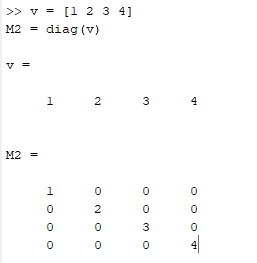
**rand:** Números aleatorios distribuidos uniformemente.



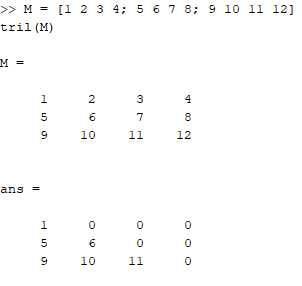
**permute:** Me cambia el orden de las dimensiones en este caso filas, columnas y cantidad.



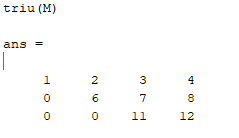
**diag:** Traza Diagonal y el resto 0 en la matriz.



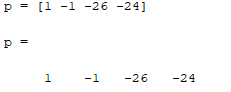
**trill:** La parte triangular inferior de la matriz.



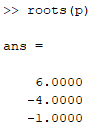
**triu:** Matriz triangular superior.



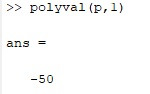
**Guardar un polinomio:**



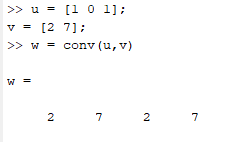
**roots:** Calculamos las raíces del polinomio.



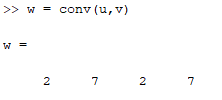
**polyval:** Evaluar el polinomio en un punto dado.



**conv:** devuelve la convolución Circunvolución de los vectores u y v. Si u y v son vectores de coeficientes polinómicos, convolución equivale a multiplicar los dos polinomios.

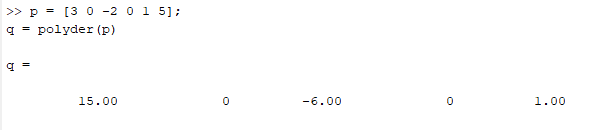


**deconv:** División de un polinomio.

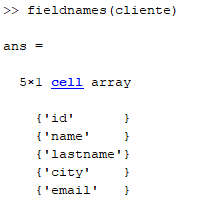


# Clase Semana 4

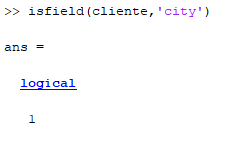
**polyder :** Derivada de un polinomio ejemplo: Y la derivada es :



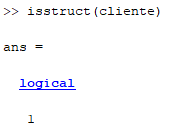
**fieldnames :** Conozco los campos de una estructura.



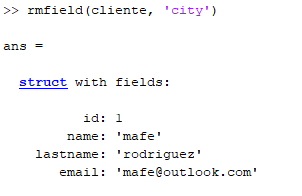
**Isfield:** Devuelve un valor lógico 1 si la clave esta en la estructura, de lo contrario retorna un 0.



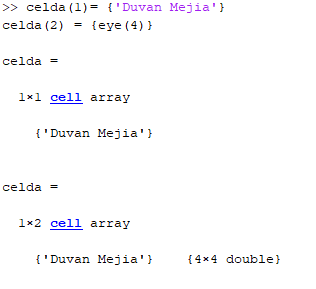
**Isstruct:** Devuelve un valor lógico 1 si el valor es una estructura de lo contrario retorna 0.



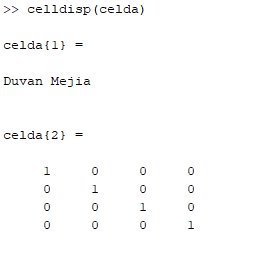
**rmfield:** Elimina un campo de la estructura.



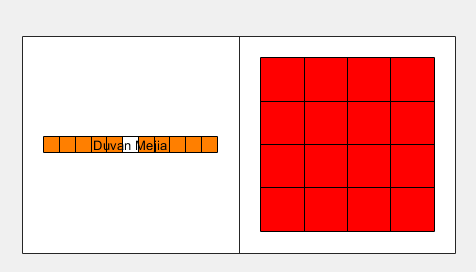
**celda o cell :** devuelve una Matriz de celda



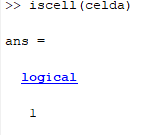
**celldisp:** Me retorna las celdas



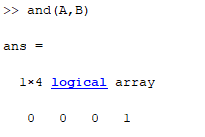
**cellplot:** me grafica la celda.



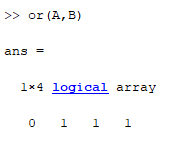
**iscell :** Me retorna un valor lógico de 1 si celda es una cell de lo contrario me retorna un 0



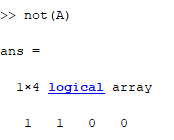
**and:** devuelve un valor lógico entre el vector A y el vector B.



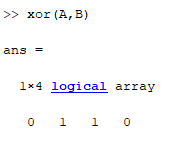
**or:** de vuelve valor lógico si cumple la condición de or



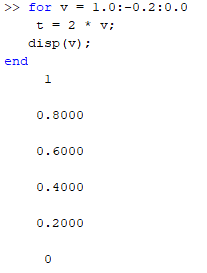
**not:** devuelve un valor lógico 1 si el valor del vector es false y 0 si el vector es true.



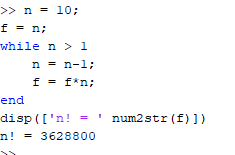
**Xor:** Siempre que las entradas de los vectores sean distinta la salida es 1



**for:** repetición número de veces



**while:** repite la declaración cuantas veces sea necesario si la condición es verdadera.



**if:** ejecuta la instrucción si es verdadera

