



Universidad
de Alcalá

Práctica 0. Introducción a Matlab (I)

Sistemas de Control Inteligente

Grado en Ingeniería Computadores

Grado en Ingeniería Informática

Grado en Sistemas de Información

Universidad de Alcalá

1. INTRODUCCIÓN

MATLAB es un entorno de programación y lenguaje de programación de alto nivel desarrollado por MathWorks. Su nombre proviene de "MATrix LABoratory" ya que fue diseñado originalmente para facilitar operaciones matriciales. Es un software ampliamente utilizado en ingeniería, matemáticas, y ciencias aplicadas debido a sus capacidades avanzadas de análisis numérico y visualización que utilizaremos en el laboratorio para la realización de las prácticas.

Algunas características y usos de MATLAB incluyen:

1. **Cálculos matriciales:** siendo la función para la que fue originalmente concebido, MATLAB facilita operaciones con matrices y vectores, lo que lo hace útil para álgebra lineal, cálculos diferenciados e integrados, y transformadas.
2. **Gráficos y visualización:** MATLAB tiene herramientas extensas para trazar gráficos 2D y 3D, visualizar datos y construir interfaces gráficas de usuario.
3. **Programación:** Además de ser un entorno para cálculos matemáticos, MATLAB es un lenguaje de programación completo, permitiendo a los usuarios escribir sus propias funciones y scripts.
4. **Toolboxes:** como complemento a la funcionalidad básica de MATLAB, existe una amplia variedad de colecciones de funciones especializadas en áreas específicas, denominadas toolboxes, tales como procesamiento de señales, control de sistemas, electrónica de potencia, entre otros muchas.

Junto con MATLAB, utilizaremos también una herramienta complementaria denominada **SIMULINK**. Simulink es una herramienta gráfica de modelado, simulación y análisis de sistemas dinámicos integrada con MATLAB que permite la simulación y modelado basados en diagramas de bloques. Algunas características interesantes de este software son:

1. **Entorno gráfico:** Simulink utiliza un entorno basado en diagramas de bloques. Los usuarios pueden arrastrar y soltar bloques de una amplia biblioteca y conectarlos para construir modelos complejos.
2. **Amplia biblioteca de bloques:** Simulink tiene una extensa biblioteca de bloques predefinidos para modelar sistemas continuos, discretos, híbridos, y eventos discretos. Esto incluye bloques para operaciones matemáticas, funciones de entrada/salida, y controladores, entre otros.
3. **Toolboxes especializadas:** Al igual que MATLAB, Simulink también se puede extender con toolboxes especializados. Estos pueden ser para control automático, procesamiento de señales, comunicaciones, sistemas embebidos y muchas otras áreas.

Existen infinidad de manuales y cursos gratuitos sobre el uso de MATLAB y Simulink, pero, sin duda, la documentación más completa está en la página web de Mathworks:

<https://es.mathworks.com/help/matlab/>

<https://es.mathworks.com/help/simulink/>

2. EJERCICIOS

En esta primera sesión de la práctica 0, se proponen algunos ejercicios básicos para la familiarización con el uso de MATLAB. En la segunda sesión se trabajará con Simulink.

Ejercicio 1. Matrices y vectores.

Realice un script de Matlab que permita desarrollar una serie de operaciones con una matriz:

1. Cree la siguiente matriz A y el vector v:

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \\ 7 & 8 \end{bmatrix} \quad v = \begin{bmatrix} 14 \\ 16 \\ 18 \\ 20 \end{bmatrix}$$

2. Obtenga y visualice una matriz B concatenando la matriz A y el vector v.
3. Obtenga y visualice un vector fila resultado de concatenar las filas de la matriz B.
4. Obtenga y visualice un vector columna resultado de concatenar las columnas de la matriz B.

Ejercicio 2. Matrices y vectores.

Realice un script de Matlab que permita desarrollar una serie de operaciones con una matriz:

1. El script ha de generar una matriz, cuadrada y aleatoria de tamaño indicado por el usuario. En la línea de comandos se ha de visualizar el mensaje: "Indique el tamaño de la matriz".

Algunas funciones útiles: input, rand.

2. A partir de la matriz construida, el script deberá calcular y presentar por pantalla los siguientes datos:
 - a) Matriz generada.
 - b) Una segunda matriz formada por las columnas impares de la matriz inicial
 - c) El valor de los elementos de la diagonal de la matriz generada.
 - d) Valor máximo, mínimo, medio y varianza de cada fila. Estos valores se han de representar gráficamente, indicando en el eje de abscisas el número de fila

Algunas funciones útiles: disp, for, diag, det, max, min, mean, var, figure, plot, hold on, hold off, title, xlabel, ylabel, size.

Ejercicio 3. Matrices y vectores.

Programe un script en Matlab que permita realizar una serie de operaciones con dos matrices (**A** y **B**) que se introducirán por teclado. Para ello:

1. Solicite al usuario las dimensiones de las matrices en formato [filas cols], (si se introduce un único número, la matriz será cuadrada).
2. Genere dos matrices (**A** y **B**) de las dimensiones elegidas. Para rellenar las matrices, escriba una función en Matlab (en un fichero diferente) que reciba como parámetro las dimensiones deseadas [filas cols], y devuelva la matriz rellena.

function Matriz = IntroducirMatriz(Dimensiones);

3. La función debe pedir datos al usuario para cada posición de la matriz. En caso de que el usuario escriba 'r', la matriz se rellenará de valores aleatorios

4. Calcule y muestre por pantalla:

- Las matrices generadas A y B
- La transpuesta e inversa de cada una de las matrices
- El valor del determinante y el rango de cada una de las matrices
- El producto de A y B (matricial y elemento a elemento)
- Un vector fila obtenido concatenando la primera fila de cada una de las matrices
- Un vector columna obtenido concatenando la primera columna de cada una de las matrices

En caso de que no sea posible realizar alguno de los cálculos solicitados, indíquelo por pantalla.

Algunas funciones útiles: input, for, if..else, det, inv, pinv, rand, randn, rank, size.

Ejercicio 4. Representación gráfica en 3D

Realice un script en Matlab que dibuje sobre el área $-5 \leq x, y \leq 5$ la superficie, la superficie en forma de malla y el contorno de la función:

-5 +5

$$z = y * \sin\left(\pi * \frac{x}{10}\right) + 5 * \cos((x^2 + y^2)/8) + \cos(x + y)\cos(3x - y).$$

- En la misma figura dibuje en la parte superior y centrada la gráfica de la superficie, en la parte inferior izquierda la gráfica de la superficie en forma de malla y en la parte inferior derecha la gráfica del contorno. Además, añada la barra de color al contorno.
- Deben añadirse etiquetas a los ejes, y un título a cada gráfica

Algunas funciones útiles: meshgrid, mesh, surf, contourf xlabel, ylabel, subplot.

Ejercicio 5. Transformadas de señales.

1. Obtenga la transformada z de la siguiente función: $f(k) = 2 + 5k + k^2$. Represente gráficamente las señales original y transformada.
2. Obtenga la transformada z de la siguiente función: $f(k) = \sin(k) \cdot e^{-ak}$. Represente gráficamente, de nuevo, la señales original y transformada.
3. Dada la siguiente función de transferencia discreta:

$$T(z) = \frac{0.4 \cdot z^2}{z^3 - z^2 + 0.1z + 0.02}$$

- Obtenga y represente la respuesta al impulso del sistema.
- Obtenga y represente la respuesta del sistema ante una entrada escalón.

Algunas funciones útiles: ztrans, step, impulse, tf (para la resolución de este ejercicio es necesario tener instalada las Toolboxes "Symbolic Math Toolbox" y "Control System Toolbox").

3. Entrega de la memoria de la práctica

Se realizará una única entrega de las actividades Intro Matlab (I) e Intro Matlab (II) que contenga 2 archivos:

- Un informe/memoria (en .pdf) que incluya las 2 prácticas con el nombre AX_PXX_Practica0.pdf (por ejemplo A2_P02_Practica0.pdf). Esta memoria debe incluir los datos / tablas solicitadas en los guiones, capturas de pantalla de los códigos programados, los resultados numéricos o gráficos obtenidos, los problemas encontrados y soluciones propuestas, así como las respuestas a aquellas preguntas / comparativas / razonamientos que se indican en los guiones. [diagramas visuales como el de flujo para explicar las líneas de código](#)

- Un archivo comprimido .zip con el nombre Grupo_Puesto_Practica0.zip (por ejemplo A2_P02_Practica0.zip) que contenga los archivos fuente de los apartados implementados tanto de Matlab (.m / .mlx) como Simulink (.slx).