MATLAB Fundamentals

Length: Three Days

Prerequisites: Familiarity with undergraduate level mathematics and experience with basic computer operations.

Summary: MATLAB Fundamentals is a three-day course that provides a comprehensive introduction to the MATLAB technical computing environment. This course is intended for beginning users and those looking for a review. No prior programming experience or knowledge of MATLAB is assumed, and the course is structured to allow thorough assimilation of ideas through hands-on examples and exercises. MATLAB competency is developed in a natural way, with an emphasis on practical application. Themes of data analysis, visualization, modeling, and programming are explored throughout the course. Topics Include:

- Working with the MATLAB user interface
- Entering commands and creating variables
- Performing analysis on vectors and matrices
- Visualizing vector and matrix data
- Working with data files
- Working with data types
- Automating commands with scripts
- Writing programs with logic and flow control
- Writing functions

_____

Course Content

Day 1
Introduction
- Obtain a quick overview of The MathWorks and MATLAB
- Discuss course set-up, materials, and logistics
- Provide a "big picture" view of the course ahead

This chapter introduces the main features of the MATLAB® integrated design environment and its user interfaces. Many themes for the course are established in this chapter, to be explored in detail in later chapters.

Working with the MATLAB User Interface

- Reading data from file

- Saving and loading variables
- Plotting data
- Customizing plots
- Calculating statistics and best-fit line
- Exporting graphics for use in other applications

This chapter introduces the concept of entering MATLAB® commands. Creating and accessing data in variables is emphasized.

Variables and Expressions

- Entering commands
- Creating variables
- Getting help
- Accessing and modifying values in variables
- Creating character variables

This chapter concentrates on performing mathematical and statistical calculations with vectors, and creating basic visualizations. The benefit of MATLAB® syntax is demonstrated as it allows for calculations
on whole data sets with a single command.

Analysis and Visualization with Vectors

- Calculations with vectors
- Plotting vectors
- Basic plot options
- Annotating plots

This chapter focuses on matrices. Matrices can be used as mathematical objects, or as collections of (vector) data. The appropriate use of MATLAB® syntax to distinguish between these applications is highlighted.

Analysis and Visualization with Matrices

- Size and dimensionality
- Calculations with matrices
- Statistics with matrix data
- Plotting multiple columns
- Reshaping and linear indexing
- Multidimensional arrays

As the complexity of your tasks increases, entering long sequences of commands in the Command Window becomes impractical. This chapter focuses on collecting MATLAB® commands into scripts for ease of
reproduction and experimentation.

Automating Commands with Scripts

- A modeling example
- The Command History
- Creating script files
- Running scripts
- Cells

Day 2

Working with Data Files This chapter considers the common important task of bringing data into MATLAB® from formatted files. Because imported data can be of a

wide variety of types and formats, emphasis is given to working with
cell arrays and date formats.

Working with Data Files

- Importing data
- Mixed data types
- Cell arrays
- Numbers, strings, and dates
- Exporting data

This chapter focuses on making more complex vector plots: making multiple plots, and using color and string manipulation techniques to produce eye-catching visual representations of data**.**

Multiple Vector Plots

- Graphics structure
- Multiple figures, axes, and plots
- Plotting equations
- Using color
- Customizing plots

This chapter highlights the use of logical operations, variables, and indexing techniques to create flexible code that can make decisions and adapt to different situations. Other programming constructs are considered as well, for repeating sections of code,
and allowing interaction with the user.

Logic and Flow Control

- Logical operations and variables
- Logical indexing
- Flow control
- Loops
- User input and output
- Publishing code

Matrix and Image Visualization This chapter investigates the capabilities of MATLAB® for visualizing images and also matrix data in two or three dimensions. The differences in displaying images and visualizing matrix data using images is
highlighted.

Matrix and Image Visualization

- Indexed images and colormaps
- Surface plots
- Contour plots
- True color images

This chapter focuses on performing typical data analysis tasks in MATLAB®. Emphasis is given to developing and fitting theoretical models to real-life data. This leads naturally to one of the most powerful features of MATLAB: solving linear systems of equations with a single command.

Data Analysis

- Correlation
- Fitting and interpolation
- Smoothing
- Regression models
- Solving linear systems of equations
- Spectral analysis and FFTs

Day 3

This chapter demonstrates how to increase automation by encapsulating modular tasks as user-defined functions. Understanding how MATLAB® resolves references to files and variables is highlighted.

Writing Functions

- Creating functions
- Calling functions
- Workspaces
- Subfunctions
- Path and precedence

Very few programs work perfectly at the first attempt. Tracking down all possible problems and unintended behaviors of a program takes time and effort. In this chapter we discuss the tools available in MATLAB to assist in the debugging process. We also highlight techniques for measuring code performance.

Troubleshooting Code and Improving Performance

- Debugging with the MATLAB® Editor
- Using breakpoints
- Measuring code performance

Data types differ from one another in the kind of data they may contain and the way the data is organized. This chapter introduces the types that have not been discussed elsewhere in the course, focusing on the syntax for creating variables and accessing array elements. The chapter also discusses methods for converting among data types.

Data Types

- MATLAB® data types
- Integers
- Cells
- Structures
- Function handles
- Converting types

File I/O As well as high-level data import and export functions, MATLAB® also provides low-level functions that allow precise control over text and binary file I/O. This chapter highlights the use of these functions, particularly textscan, which provides

precise control of reading text files.

File I/O

- Opening and closing files
- Reading and writing text files
- Reading and writing binary files