SigSys II and NumMeth: MATLAB tips and resources

IfA & SAM
ETH Zurich
MATLAB

As a tool:

▶ Engineering tool, started with LA, now much more
▶ Extensive amount of research functionality
▶ Widely used at ETH, (engineering) academia in general

As a language:

▶ Dynamically-typed and interpreted
▶ Great for prototypes, less so for larger projects
▶ Powerful, very user-friendly debugger
Available resources

Large amount of tutorials available:

▶ MATLAB’s searchable official documentation
  ▶ Documentation much more extensive than what help somecommand
  ▶ Scroll down on these pages, often there are even usage examples

▶ MATLAB’s own getting started guide

▶ MATLAB’s online language fundamentals guide

⇒ Purpose of these slides: Help you better find help on your own

Important in all of programming: Experiment!
Important MATLAB concepts

▶ Command Window:
  ▶ Where you “run” your commands, formulas, functions, scripts.
Important MATLAB concepts

- **Command Window:**
  - Where you “run” your commands, formulas, functions, scripts.

- **Current folder:**
  - Folder “we are in”, always “on path”.
  - Changes if browsing into other folders.
Important MATLAB concepts

- **Command Window:**
  - Where you “run” your commands, formulas, functions, scripts.

- **Current folder:**
  - Folder “we are in”, always “on path”.
  - Changes if browsing into other folders.

- **Workspace:**
  - Similar to namespace in other languages.
  - Global workspace for scripts, each function has its own.
  - Visual “variable browser”.
Important MATLAB concepts

▶ Command Window:
  ▶ Where you “run” your commands, formulas, functions, scripts.

▶ Current folder:
  ▶ Folder “we are in”, always “on path”.
  ▶ Changes if browsing into other folders.

▶ Workspace:
  ▶ Similar to namespace in other languages.
  ▶ Global workspace for scripts, each function has its own.
  ▶ Visual “variable browser”.

▶ Command History:
  ▶ Collection of executed commands, with date and time tag.
Important MATLAB concepts

▶ Command Window:
  ▶ Where you “run” your commands, formulas, functions, scripts.
▶ Current folder:
  ▶ Folder “we are in”, always “on path”.
  ▶ Changes if browsing into other folders.
▶ Workspace:
  ▶ Similar to namespace in other languages.
  ▶ Global workspace for scripts, each function has its own.
  ▶ Visual “variable browser”.
▶ Command History:
  ▶ Collection of executed commands, with date and time tag.
▶ Path:
  ▶ List of folders in which MATLAB looks for functions and classes.
  ▶ User-settable, defines precedence of functions.
Short hands-on session

- As a simple calculator, using variables.
Short hands-on session

- As a simple calculator, using variables.
- Working with vectors and matrices:
  - Initialization.
  - Linear algebra operations.
  - Element-wise and vectorized operations.
Short hands-on session

- As a simple calculator, using variables.
- Working with vectors and matrices:
  - Initialization.
  - Linear algebra operations.
  - Element-wise and vectorized operations.
- Control statements:
  - for loop, while loop.
  - if else.
Short hands-on session

- As a simple calculator, using variables.
- Working with vectors and matrices:
  - Initialization.
  - Linear algebra operations.
  - Element-wise and vectorized operations.
- Control statements:
  - for loop, while loop.
  - if else.
- Scripts, functions, anonymous functions.
Debugging

Tutorial: Mathworks website

- Start debugger if error is found: `dbstop if error`
- Continue running the code normally: `dbcont`
- Exit debugger: `dbquit`
Alternatives

▶ Octave
  ▶ Free “clone” of MATLAB.
  ▶ Not all the same functionality, but code almost the same.
  ▶ Has a GUI.

▶ Python
  ▶ Free, also after you graduate.
  ▶ Widely used in different fields, Deep Learning - TensorFlow, PyTorch.
  ▶ Some useful packages: NumPy, SciPy, SymPy, control.
  ▶ Less unified, sometimes less well-documented and “ready-to-use”.

Recommendation: Start with MATLAB, keep an eye on Python.