Macro Analysis II: Recitation 1

January 20, 2020

Roadmap for today

- Setting up MATLAB
- Intro to MATLAB programming
- General recommendations

MATLAB

Intro

- Excelent scientific computing software

MATLAB

Intro

- Excelent scientific computing software
- Pros:
  1. Intuitive (no complicated 'types', declaring, pointers, etc)
  2. Very fast linear algebra
  3. Good optimization tools
  4. "Industry standard" for computational economics

MATLAB

Intro

- Excelent scientific computing software
- Pros:
  1. Intuitive (no complicated 'types', declaring, pointers, etc)
  2. Very fast linear algebra
  3. Good optimization tools
  4. "Industry standard" for computational economics
- Cons:
  1. Closed source
  2. Expensive
     - However: you can use Columba Econ license (walk-through next slide)
     - Also, open source alternatives: Scilab, Octave, Julia
  3. Not popular outside academia

MATLAB

Obtaining it

- Click here to active Columbia license: `https://www.mathworks.com/licensecenter/total_headcount/19313-53004-70961-65618-81123?s_tid=tah_po_start_columbia`

Intro to MATLAB programming: Basics

- Comes with a "point-and-click" interface where:

Intro to MATLAB programming: Basics

- Comes with a "point-and-click" interface where:
  1. You can run instructions directly to the interpreter, such as

     ```
     M = [1 4 7;2 5 8;3 6 9]
     ```

Intro to MATLAB programming: Basics

- Comes with a "point-and-click" interface where:
  1. You can run instructions directly to the interpreter, such as

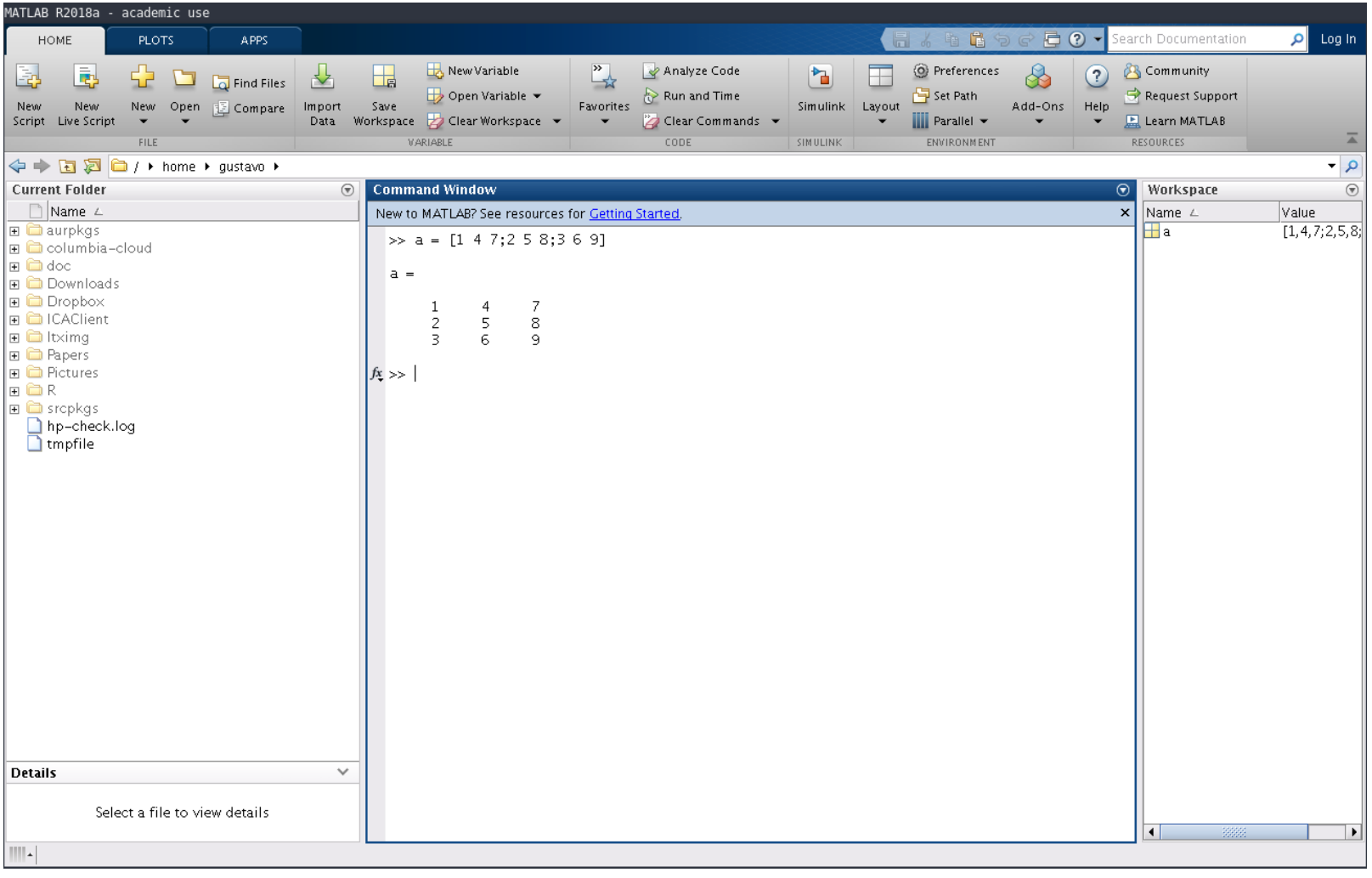
     `M = [1 4 7;2 5 8;3 6 9]`

     - Meaning: store the matrix

$$\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$

     in the memory as a variable `M`

Interface: main



Intro to MATLAB programming: Basics

- Comes with a "point-and-click" interface where:
  1. You can run instructions directly to the interpreter, such as

     `M = [1 4 7;2 5 8;3 6 9]`

     - Meaning: store the matrix

$$\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$

     in the memory as a variable `M`

Intro to MATLAB programming: Basics

- Comes with a "point-and-click" interface where:
  1. You can run instructions directly to the interpreter, such as

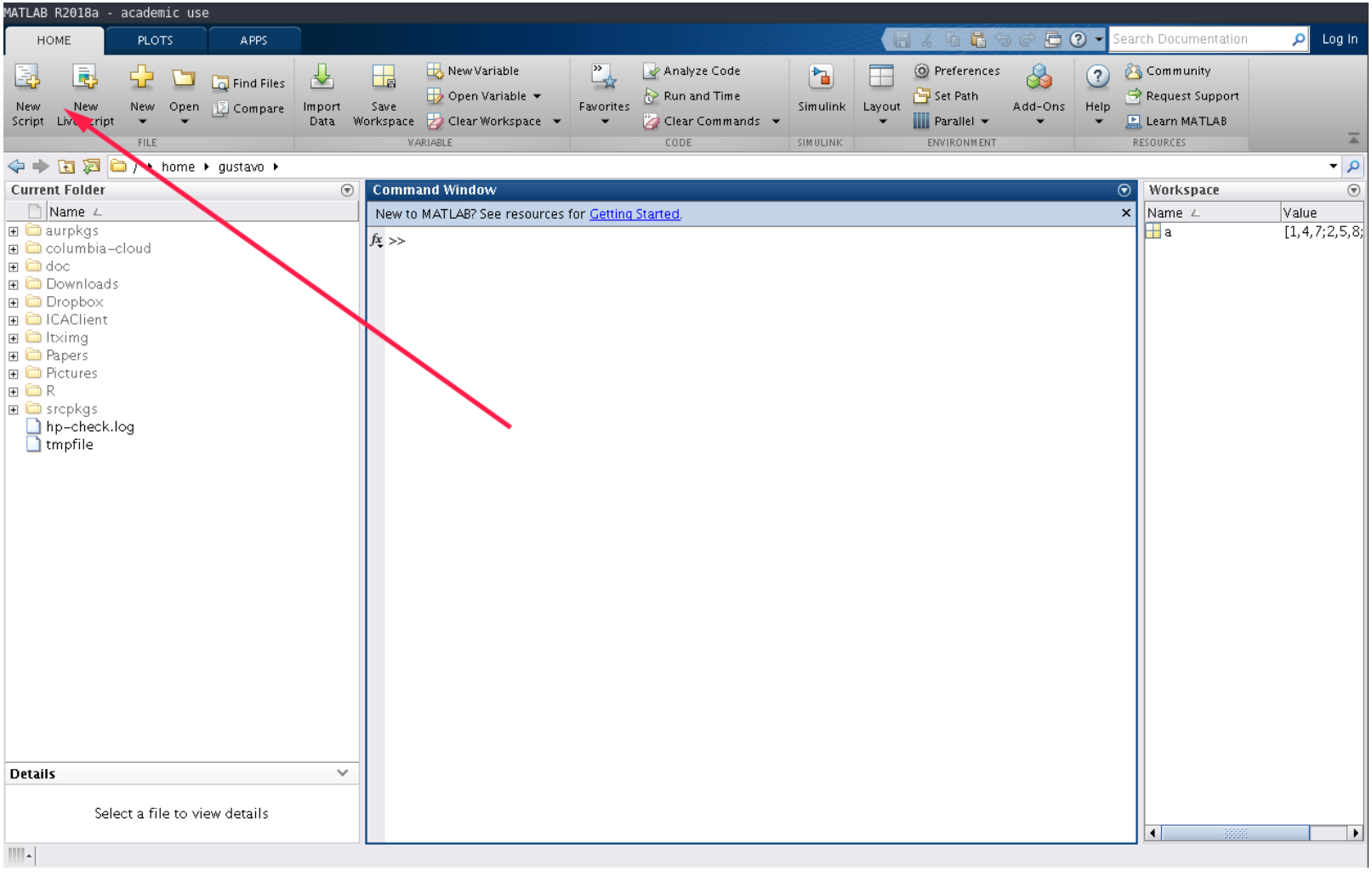
     `M = [1 4 7;2 5 8;3 6 9]`

     - Meaning: store the matrix

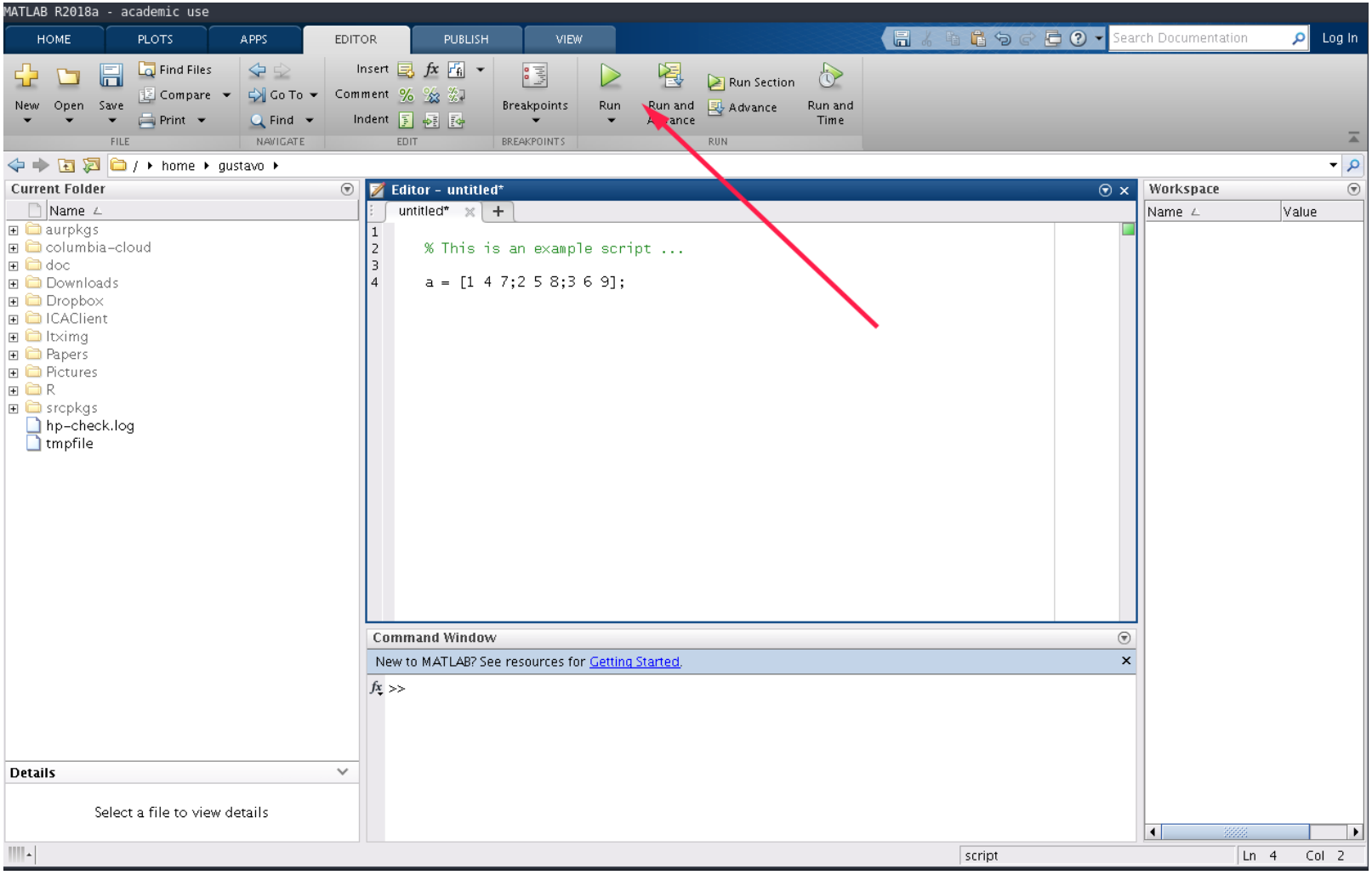
     $$\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$

     in the memory as a variable `M`
  2. You can (and *should*) write "scripts" (called *m files*)
     - Essentially, a list of commands to be run by the interpreter

Interface: script



Interface: script



Intro to MATLAB programming: example

Suppose you're programming something and you run into the following problem.

Problem

You have some matrix A and need two numbers:

1. The mean of all elements in the matrix
2. The mean of all elements except for the diagonal ones

Intro to MATLAB programming: example

- First attempt: `$repo_path/Codes/Recitation1/badcode.m`
- Results are stored in variables `meanA` and `meanB`

Intro to MATLAB programming: example

- First attempt: `$repo_path/Codes/Recitation1/badcode.m`
- Results are stored in variables `meanA` and `meanB`
- Why is it bad code?

Intro to MATLAB programming: example

- First attempt: `$repo_path/Codes/Recitation1/badcode.m`
- Results are stored in variables `meanA` and `meanB`
- Why is it bad code?
  1. Using a loop instead of built-in functions

Intro to MATLAB programming: example

- First attempt: `$repo_path/Codes/Recitation1/badcode.m`
- Results are stored in variables `meanA` and `meanB`
- Why is it bad code?
  1. Using a loop instead of built-in functions
  2. Variable names are meaningless

Intro to MATLAB programming: example

- First attempt: `$repo_path/Codes/Recitation1/badcode.m`
- Results are stored in variables `meanA` and `meanB`
- Why is it bad code?
  1. Using a loop instead of built-in functions
  2. Variable names are meaningless
  3. Code isn't commented

Intro to MATLAB programming: example

- First attempt: `$repo_path/Codes/Recitation1/badcode.m`
- Results are stored in variables `meanA` and `meanB`
- Why is it bad code?
  1. Using a loop instead of built-in functions
  2. Variable names are meaningless
  3. Code isn't commented
- See now `$repo_path/Codes/Recitation1/goodcode.m`

Intro to MATLAB programming: recommendations

- *In general:*

Intro to MATLAB programming: recommendations

- *In general:*
  - Have a coherent, well organized set of *scripts*, *input files* and *output files*

Intro to MATLAB programming: recommendations

- *In general:*
  - Have a coherent, well organized set of *scripts*, *input files* and *output files*
  - Don't repeat yourself. Write functions!

Intro to MATLAB programming: recommendations

- *In general:*
  - Have a coherent, well organized set of *scripts*, *input files* and *output files*
  - Don't repeat yourself. Write functions!
- *For MATLAB in particular:*

Intro to MATLAB programming: recommendations

- *In general:*
  - Have a coherent, well organized set of *scripts*, *input files* and *output files*
  - Don't repeat yourself. Write functions!
- *For MATLAB in particular:*
  - Keep each function in a separate file (be careful with paths though)

Intro to MATLAB programming: recommendations

- *In general:*
  - Have a coherent, well organized set of *scripts*, *input files* and *output files*
  - Don't repeat yourself. Write functions!
- *For MATLAB in particular:*
  - Keep each function in a separate file (be careful with paths though)
  - Avoid loops: use built-in functions whenever possible

Intro to MATLAB programming: recommendations

- *In general:*
  - Have a coherent, well organized set of *scripts*, *input files* and *output files*
  - Don't repeat yourself. Write functions!
- *For MATLAB in particular:*
  - Keep each function in a separate file (be careful with paths though)
  - Avoid loops: use built-in functions whenever possible
  - Avoid using global variables: pass parameters as arguments to functions

Other software advice

- Have a decent text editor/IDE that integrates well with your favorite programming languages
  - Idea: efficient text editing leaves you more time to think
  - Problem: possibly steep learning curve
  - Suggestions: Atom, Sublime Text, VScode, Vim, Emacs, etc..

Other software advice

- Have a decent text editor/IDE that integrates well with your favorite programming languages
  - Idea: efficient text editing leaves you more time to think
  - Problem: possibly steep learning curve
  - Suggestions: Atom, Sublime Text, VScode, Vim, Emacs, etc..

- If you do lots of coding (or Stata) adopt version control ASAP (esp. if co-authoring)
  - Recommended: git