KU LEUVEN

MATLAB

IO : high level functions

input / output

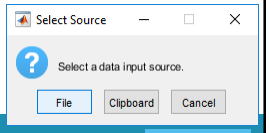
- File IO
  - Import tool
  - high level io: `save` and `load`
    - mat-files
    - text files
  - more info:
    - `doc fileformats`
    - `doc iofun`

Data Import interactively

Importing data interactively

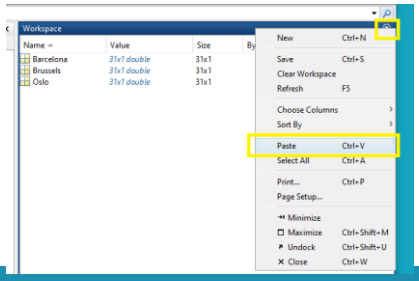
- You can import data into MATLAB from a disk file or the system clipboard interactively.
- To import data from the clipboard, do one of the following:
  - On the Workspace browser title bar, click , and then select Paste.
  - Call `uiimport (> clipboard)`
- To import data from a file:
  - On the Home tab, in the Variable section, select Import Data .
  - Double-click a file name in the Current Folder browser.
  - Call `uiimport (> file)`

https://nl.mathworks.com/help/matlab/import_export/recommended-methods-for-importing-data.html



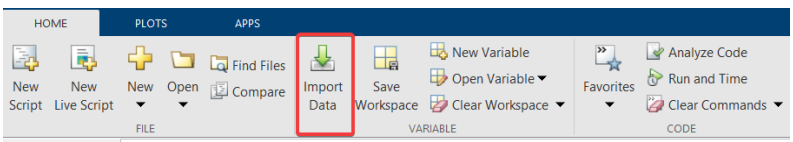
Import from clipboard

- To import data from the clipboard, do one of the following:
  - Call `uiimport (> clipboard)`
  - On the Workspace browser title bar, click , and then select Paste.
  - Click in Workspace browser and Paste



Import Data tool

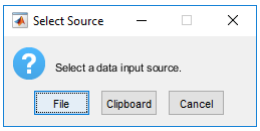
- https://www.mathworks.com/help/matlab/ref/importtool-app.html?searchHighlight=import%20data&s_tid=doc_srchtitle
- Click Import Data.
- Follow the steps of the import wizard
  - Select file
  - Select output type (how the variables will be stored)
    - Numeric matrix
    - Column vectors
  - Import selection

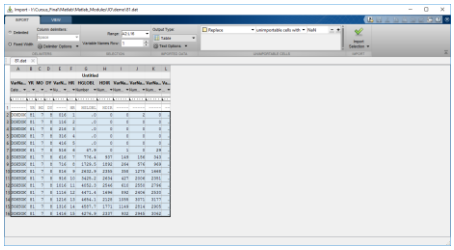


- *File: temp3city.dat*
- *File: temp3city_missing.dat*
- *File: temp3city_header.dat*

uiimport

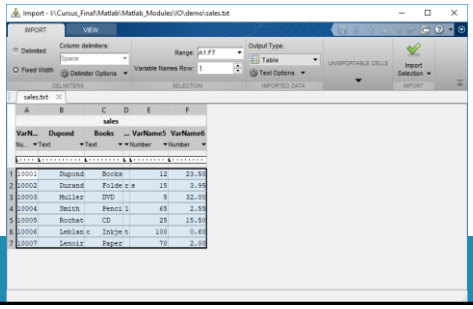
- MATLAB command prompt: `uiimport(filename)`
- `uiimport`
  - import data from a file or clipboard





Generate code

- Import wizard can generate a script or function from the selected options
- Useful to import similar files
- DIY:
  - Import temp3city.dat (sales.txt is more complex)
  - Align the columns (if necessary)
  - Select output type
  - Generate code



Double click a file

- Double-click a file name in the Current Folder browser
  - Extension .txt is opened in editor
  - Extension .dat is opened in import wizard

- *File: sales.txt*
- *File: 81.dat*
- *File: SampleDataUoR.xlsx*

High level IO

High level io functions

File Content	Extension	Description	Import Function	Export Function
MATLAB formatted data	MAT	Saved MATLAB workspace	load	save
Text	any	White-space delimited **numbers**	load	save -ascii
		Delimited **numbers**	readmatrix	writematrix
		Delimited numbers, or a **mix of strings and numbers**	textscan	
		Column-oriented delimited numbers or a **mix of strings and numbers**	readtable	writetable

High level io functions

File Content	Extension	Description	Import Function	Export Function
Spreadsheet	XLS XLSX XLSM XLSB (Systems withMicrosoft® Excel ®for Windows® only) XLTM (import only) XLTX (import only) ODS (Systems with COM interface)	Worksheet or range of spreadsheet	readmatrix	writematrix
		Column-oriented data in worksheet or range of spreadsheet	readtable	writetable

save/load mat-files

- To export workspace variables to a binary or ASCII file, use the `save` function. (easiest way)
- Save all variables from the workspace in a single operation (default file matlab.mat):
  `save(filename)`
- Save the variables that you specify:
  `save(filename, var1, var2, ... varN)`
- Use of wildcard character (*) in the variable name is allowed
  `save (filename, str*)`
- whos -file examines contents of the MAT-file:
  `whos -file filename`

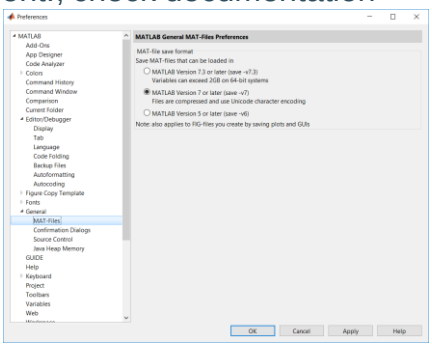
save/load mat-files - append

- add new variables to those already stored in an existing MAT-file with
  `save(filename, var1, var2, ... varN, '-append')`
  - For each variable that already exists in the MAT-file, MATLAB overwrites its saved value with the new value taken from the workspace.
  - For each variable not found in the MAT-file, MATLAB adds that variable to the file and stores its value from the workspace.

save/load mat-files - compression

- MATLAB compresses the data that you save to a MAT-file.
- can save a significant amount of storage space
- caution! version dependent!, check documentation

- **Preferences** dialog, select **General MAT-Files**



save/load mat-files - ascii

```
save(filename, var1, var2, ... varN, '-ascii')
```

Each variable to be saved must be either a two-dimensional double array or a two-dimensional character array.

- Saving a complex double array causes the imaginary part of the data to be lost
- Each MATLAB character in a character array is converted to a floating-point number equal to its internal ASCII code and written out as a floating-point number string. There is no information in the saved file that indicates whether the value was originally a number or a character.
- Advice: be careful with the –ascii option (check documentation)
- *File: io_load_save_1.m*

save/load mat-files

Using the load Function

- import variables from a binary or ASCII file on your disk to the workspace, use the `load` function. (inverse of save)
- load all variables from the workspace in a single operation (default filename: matlab.mat):
  `load(filename)`
- load specified:
  `load(filename, var1, var2, ..., varN)`
- wildcard character (*) in the variable name to load those variables that match a specific pattern. (This works for MAT-files only.)
  `load(filename, str*)`

save/load mat-files

Loading ASCII Data

- ASCII files must be organized as a rectangular table of numbers, with each number in a row separated by a blank or tab character, and with an equal number of elements in each row.
- In the workspace, MATLAB assigns the array to a variable named after the file being loaded
  `load mydata.dat`
  reads all of the data from mydata.dat into the workspace as a single array mydata

save/load mat-files

Advice:

- if data are to be exchanged between MATLAB and other programs, use the ASCII format.
  If data is to be exchanged within the MATLAB environment, use the MAT-file format
- use .dat extension for ASCII-files, .mat for MAT-files
- MAT-format contains more info, getting lost in the ascii-option