12.010 Computational Methods of Scientific Programming

Lecture 9: Graphics continued and GUIs

Overview

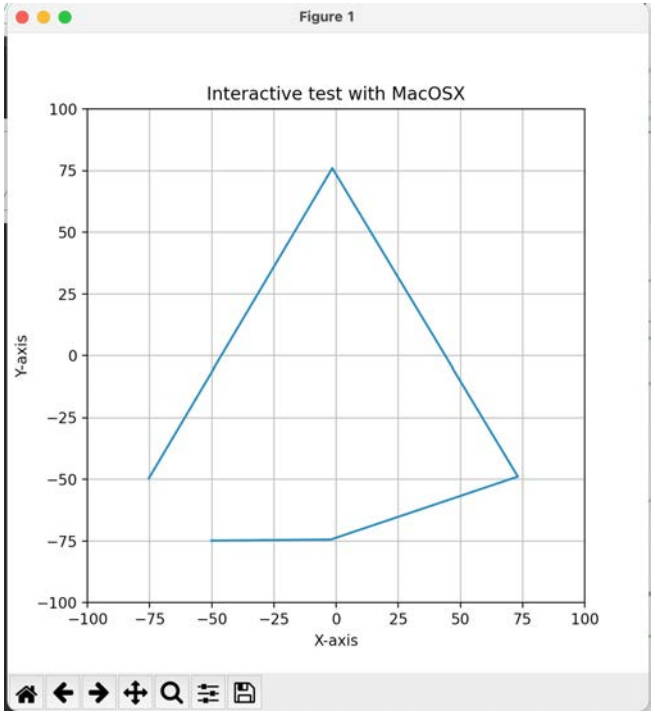
- Continue looking at different types of plots
- Interacting with graphics:
  - Backend specification
  - ginput
  - widgets

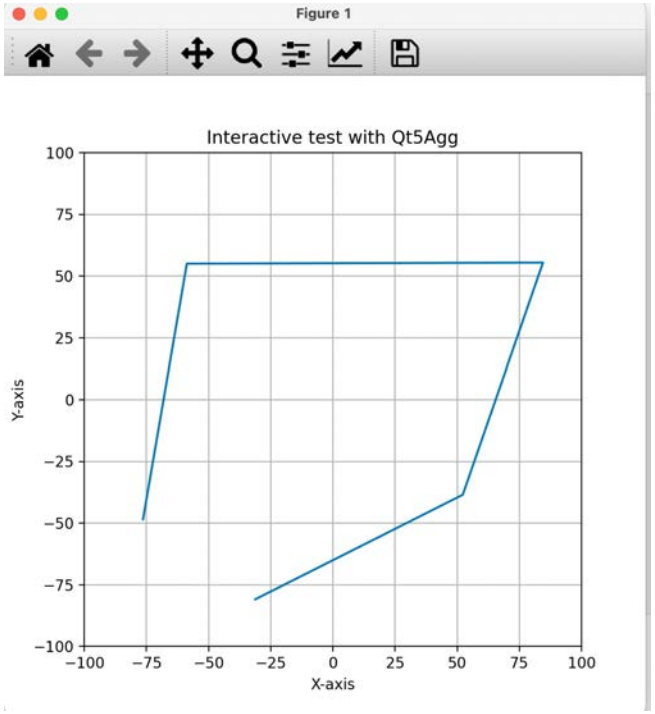
Interactive graphics and GUIs

- Graphic user interface (GUI) and interaction with graphics needs to change from inline graphics.
- Generally, virtual machines will not be able to display or use interactive capabilities.
- The “backend” used sets the behavior and can changed to allow interactive plots.
- We will look at matplotlib widgets here. Tkinter is a common GUI package, but we will not use it here (it needs additional conda installations).
- ipywidgets are Jupyter widgets.

Backends*

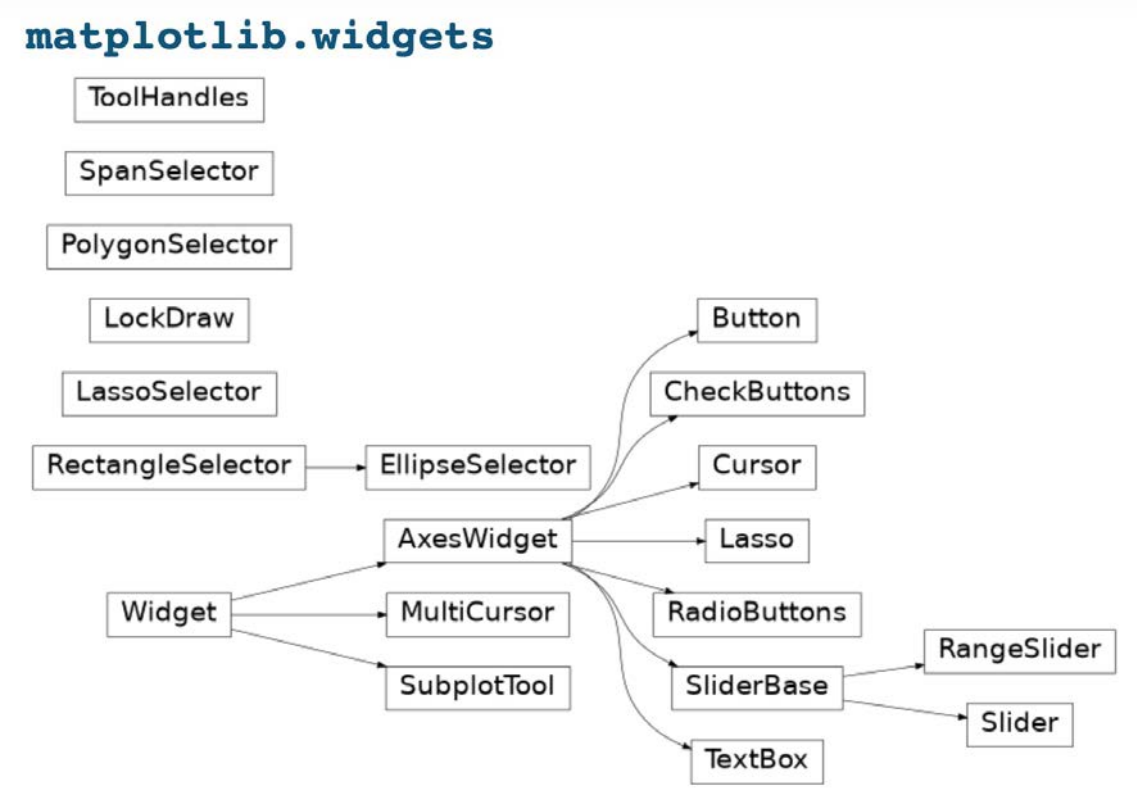
- macosx and qt5agg (screen shots): Uses ginput()





Matplotlib widgets

- Widgets available. The widgets require a set of axes be defined the widget to be displayed in and to test for event.



Example widgets and animations

- The notebook for this class includes examples of common types of GUI widgets.
- The basics concepts of using widgets are:
  - Create a fig,ax = plt.subplot()
  - For manipulating a line plot with the widget (e.g., change colors, change ydata) create initial line
    line, = ax.plot(…)
    Creates a 'line', the ',' is needed to get the line and not the line object
    Need to adjust position of axes to make room for widgets
  - Create axis region for button/slider
    rax = plt.axes([x,y,width,height],..) Units are proportions of plot
  - Create the widget object
    radio = RadioButtons(rax,…)
  - Action:
    radio.on_clicked(<function to call). Function has line.<line methods>)
  - Report button, slider codes for each operation needed.

Examples:*

- Notebook has examples of:
  - Slider and buttons
  - Radiobuttons
  - Textbox (and Python eval function)
  - Ellipse region selector
  - Simple animations: introduces new keyword yield and the concepts of iterable and generators (generators are useful for large lists that are only used once)

IPyWidgets

- These widgets work in a Jupyter (and vscode) Notebook.
- Graphics and widgets can be kept inside Notebook rather than external figure.
- However ginput does not work in these figures
- Examples: Lec10_ipywidgets.ipynb

Summary

- Finished up looking at plot types
- Looked at backends and the way matplotlib widgets can be used
- Enough tools now to build a GUI that can read data from a URL (comma separated) and then have buttons to work with the data.

MIT OpenCourseWare

https://ocw.mit.edu

12.010 Computational Methods of Scientific Programming, Fall 2024