SAGE INTERACT GUIDE

DARYL DEFORD

Introductory Questions

- **What is Sage?**
  Sage is a great tool for doing mathematics on a computer. It is an open–source project that extends the Python programming language and incorporates almost all of the major computer algebra systems. You can download Sage here.

- **What is the Sage Cell server?**
  The Sage Cell is a collection of web–browser tools that allow you to run Sage code in any web browser without installing any software or configuring a server. The main webpage for this service is here.

- **What is the interact environment?**
  The interact environment lets you convert ordinary Sage functions into an interactive GUI that controls the function inputs directly. This is a particularly useful tool for teaching purposes as it allows you to specify which parameters your students can adjust (more on this below). In addition to the official documentation, there is also a nice "Quickstart" guide. The official Sage website also collects neat examples from across the internet here. My examples are linked here (teaching) and here (research).

- **How do they fit together?**
  The combination of the Sage Cell and the interact module allows instructors to create teaching tools and lecture examples that students can access directly from their (omnipresent) devices. These widgets can easily be tailored to specific lessons or assignments which avoids the overwhelming nature of many approaches to integrating technology into undergraduate courses.

- **How do they work in classes?**
  I use these tools both to create examples for class and for the students to use on assignments. In class demonstrations allow me to both show more complex visualizations than I could reasonably draw on the board and examine problems where the algebra would be prohibitive due to time constraints. This setup also makes it straightforward to integrate real data into lectures.

  Since the widgets can be run in any browser I also encourage my students to experiment with the programs. The goal is to help them develop their intuition by looking at many different examples without getting lost in the details of the algebra. I also tend to incorporate these into homework assignments. For daily problems, which are usually simple and computational the programs can be used to check their answers. On longer weekly problems I ask them to experiment with the programs to look for patterns and generate conjectures.

Implementation Details

Once you have a gauss account everything is pretty straightforward. If you just want an empty Sage cell to experiment with, you can just upload this file to your public html directory. Once you have finalized your interact function you can copy it into this template and upload it to your public html directory. That is all it takes.

Interact Inputs

The interact environment converts the standard Sage/Python function syntax `f(a,b,c)` to a customizable GUI where each input is assigned a separate interactive element. Each element can be assigned a label that is displayed to the user and a default value that is originally assigned. The four main types of inputs are discussed below. Working examples of each of these elements are linked from the workshop page and examples of the full syntax are shown on the next page of this document.

- **Input boxes:** The simplest GUI element is also the most general. Input boxes accept everything from numbers to strings to functions and can be used for any situation.
- **Slider bars:** Slider bars are a good way to restrict the inputs to discretized values. Sage supports both single–valued slider bars as well as range sliders that let you select two values at once, for example for choosing lower and upper bounds for an integral.
- **Selectors:** To offer discrete options that aren't numbers you can use selectors, either drop down lists or rows of buttons. Sage also offers a color selector tool that passes RGB values into your functions.
- **Checkboxes:** If you want to pass Boolean variables to your function, Sage supports yes/no checkboxes.

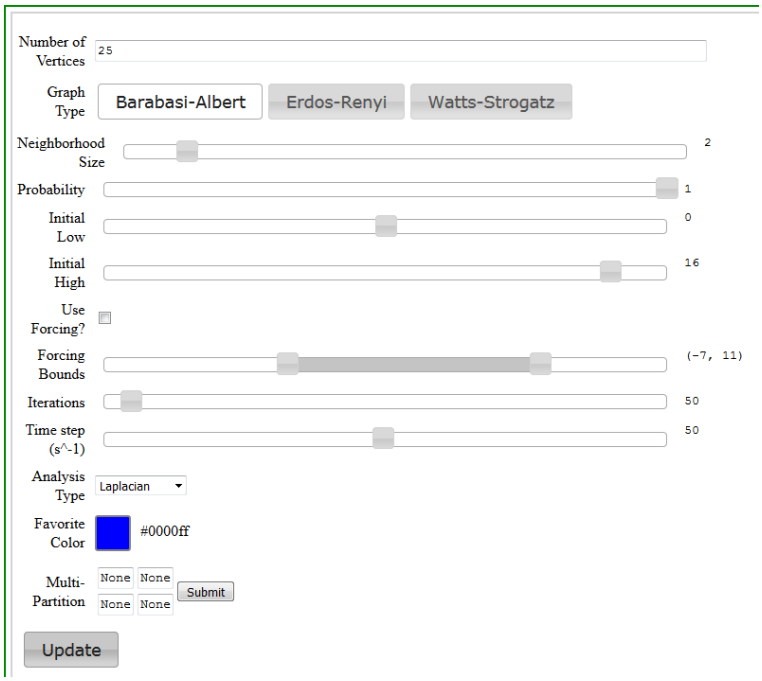

Figure 1. Example using all of the possible input types. Don't try this at home ☺

Input Examples

This page shows the syntax for each of the standard input environments. The examples can be downloaded (or copy/pasted) from here and run on the workshop cell.

```
######################################## Input Box ##############################
@interact
def _(f=input_box(default=x^2-2*x+3,label='Function: '),
s=input_box(default='Quadratic', label='Plot Label: ',type=str)):
      plot(f,-4,4,title=s).show()

######################################## Slider Bars ############################

@interact
def _(f=slider([x,sin(x),cos(x),e^x],default=sin(x),label='f :'),
n=slider(0,5,step_size=.5, default=3,label='Exponent: '),
rs=range_slider(-10,10, default=(-4,4),step_size=1, label='X Limits: ')):
    g=f^n
    plot(g,rs,title='f(x)^n').show()

######################################## Checkbox ###############################

@interact
def _(Q=checkbox(default=False,label='Show plot?' )):
      if Q==True:
          plot(x^2).show()
      if Q==False:
          print('Plot is turned off!')

######################################## Selectors ##############################

@interact
def _(f=selector([x^2,-x^3,tan(x),e^x],label='Function 1: '),
g=selector([x^2,-x^3,tan(x),e^x],buttons=True,label='Function 2: ') ):
    plot(f).show()
    plot(g,color='red').show()


######################################## Color Selector #########################

@interact
def _(c=color_selector(default=(0,1,0),label='Plot 1 Color: ',hide_box=True),
d=color_selector(default=(0,1,1),label='Plot 2 Color: ',hide_box=False)):
    plot(x^2,color=c).show()
    plot(x^3,color=d).show()
```

Interactive Plotting Activity

The purpose of this activity is for you to get a little bit of practice using the Sage interact tool to build a function plotting widget. You will work in groups of 2 or 3 to create a tool that plots 3 functions on the same set of axes using at least one of each of the Interact input types.

Instructions: Use the Sage cell at http://math.dartmouth.edu/~ddeford/sage_workshop.html and the interact module to create a GUI that accepts three different functions and plots them on the same set of axes using any number of inputs. You should begin by discussing in your group what types of information you would like to collect from the user and what type of interact option goes along with each input. You can record your choices in the table below and there are some suggestions at the bottom of the page.

Input	Interact Type

Once you have decided on the structure of your widget the syntax for function plotting is:
`plot(`x^2`,xmin=0,xmax=5,ymin=-1,ymax=30,color='red',thickness=2,title='Quadratic Plot',axes_labels=['`x` axis','`$f(x)$` axis']).show()`

More details about plotting in Sage can be found here. Once you have entered your function into the Sage Cell, switch laptops with a nearby group and test out each others widgets. What differences do you notice?

Possible Inputs: Here are some options that you might want to include in your function:

- X and Y bounds on the plot
- Plot title
- Colors for each function
- Pre-defined function options
- Multiple axes
- Axis scaling
- Animation controls
- ...