**Rubric for Weekly Assignments**

Each weekly assignment is an individualized write-up of some or all of the Group Projects conducted in the previous week. Elements of the assignment.

**Title:** including the Assignment number and a descriptive phrase.

**Author:** Your name and Group number, and the names of the other Group members. All analysis must be in your own words, but graphs and code may be shared among Group members.

**Statement of the problem:** Include some background material that broadly describes the phenomenon that you are investigating and the specific type of analysis that you have been asked to make.

**The Group’s approach.** Describe the strategy that underpins your analysis.

**Code Outline.** If your code is based on class-provided code, give its name and explain how you modified it.Describe in broad the way the code works by breaking it into a few (say, less than ten) color-coded sections and explaining what each section does. But put the code in a separate **Code** section at the end.

**Techniques Used in the Code.** List a few (say, 3 to 5) of the most important techniques that are used in the code, and provide a brief explanation of what they do.

**Analysis.** Explain what you learned from the Group Project. Cite at least one figure.

**Figures.** All figures must be numbered and must include a figure caption that explains the figure. Graphs must be “science normative” in the sense of having axes that are properly labeled (including units when appropriate). Each figure must be cited in the text.

**Code.**  Color coded code that parallels the **Code Outline** section.

Group Projects are graded on an Acceptable / Unacceptable basis and contribute to your class participation grade. An exemplary assignment is shown below.

**Assignment 1: Sensitivity of a trigonometric formula to changes in a parameter**

Bill Menke

I participated in Group 1 consisting of X and Y and myself

**Statement of the problem.** The Group project is focused on the behavior of trigonometric functions. Such functions are important because they can describe many different kinds of phenomena in the Earth and environmental sciences, including ocean and seismic waves, daily and seasonal variations in climate, etc. We are presented with a formula for the trigonometric function

where , , , and asked to analyze its behavior as is varied from to .

**The Group’s approach.** We used the simple strategy of evaluating for a selection of s and overlaying the result on a single graph, with colors used to distinguish the cases.

**Code Outline.**

This code is exactly the same Cell 3 of Lec02\_sensitivity1D.ipynb.

Part 1: Define constants ,

Part 2: Define -axis.

Part 3. Create a short list of selected values of and associate with each one a color.

Part 4: More-or-less standard x-y graph, except that the plotting of is inside a FOR loop (over the different values of )

**Techniques Used in the Code.**

(1) arrays are used to organize the -axis and the list of s.

(2) Pyplot color specification character strings (e.g. ‘r-‘) are organized as a list that parallels the list of s.

(3) Numpy trigonometric functions are used to evaluate the formula.

**Analysis.** We used a Python script (see below) to create a plot that superimposes five versions of for a suite of five representative values of (Fig. 1). The function has a single peak whose position and amplitude depends on . The position of the peak moves from right to left as is increased, with its amplitude being largest when it is near either end of the interval. The shape of the curve for is a mirror image of the shape for (with the case being exactly symmetric around ). The value is constant, irrespective of the value of .

In order to insure that these results were not dependent on our particular choice of the five values of , we also created an animation (not shown) based on equally-spaced values. No discrepancies were detected.

**Code.**

A=2.0;

L=5.0;

x=gda\_cvec(np.linspace(0.0,L,20));

B=gda\_cvec( -1.0, -0.5, 0.0, 0.5, 1.0 );

sym = ['r-', 'y-', 'g-', 'c-', 'b-'];

NB, i = np.shape(B);

fig = plt.figure();

ax = plt.subplot(1,1,1);

plt.axis( [0.0, L, -2.0\*A, 2.0\*A] );

for i in range(NB):

y = A \* np.sin( pi\*x/L ) + B[i,0] \* np.sin( 2\*pi\*x/L );

plt.plot(x,y,sym[i]);

plt.xlabel('x (m)');

plt.ylabel('y (m)');

plt.show();

print('Fig. Graph of y(x) for B=-1 (red), -0.5 (orange), 0 (yellow), 0.5 (cyan), and 1 (blue)');

|  |
| --- |
|  |
| Fig. 1: Graph of for (red), (orange), (yellow), (cyan), and (blue), |