Assignment 3:
Introduction to *Mathematica*
Introduction to Data Analysis for Physics

Evan Ott and Will Beason

Spring 2014

Submission Requirements

Submit the assignment to data.analysis.physics@gmail.com by Wednesday at 5pm. Just submit the *Mathematica* document you create (typically a .nb file).

Problem 1

For this problem, we'll look at how to inject intuition into *Mathematica* to make it be able to solve problems more easily. You will benefit from some basic calculus knowledge.

a

Let's try to calculate

$$\int_0^1 x^a dx$$

where a is an arbitrary number. If we try this in *Mathematica*, we get a conditional expression. Why? What would happen if we set $a \leq -1$? (for this one, yes, I want English)

b

How about if we wanted to calculate

$$\lim_{x \to \infty} x^r$$

for some r. What can we tell *Mathematica* so that we can evaluate the limit? (you can just show me this time)

c

In the first homework, when you calculated $f(g(x))$, this did not evaluate immediately to just x, but instead $1 + \frac{1}{2}(-2 + 2x)$. The `Simplify` and `FullSimplify` functions can simplify expressions, and are particularly helpful when you can apply some assumptions (look in the *Mathematica* documentation to see some examples). Create an example where these simplification functions cannot fully simplify the expression without applying some outside information (`x > 2` or `Element[y,Reals]` or similar), then show what it simplifies to when you do apply the background information. "Awesome points" for finding such an expression that changes the end value based on the assumption made, but with multiple simple solutions.

Problem 2

For this problem, you'll need the data at http://www.cs.utexas.edu/~evanott/ PHY110C_Textbook/static/data_analysis/_downloads/assignment3data.csv (9 MB). This is data for a simple (read: bad) gravity simulation for the Sun, Halley's Comet, the 8 planets, and Pluto (the 9th planet and everyone knows it). The columns are in groups of 3, as the x, y, and z coordinates of each of the objects (not necessarily in the order above), and each row represents an advance of 1800 seconds.

a

Find a way to determine which object is which. In my case, I made a manipulable plot that showed the position of each particle at the time I wanted. I also fixed the plot range so that I was always looking at the same range of coordinates as I manipulated time. For reference, see Figure 1. Big hint: all 9 planets are listed consecutively from Mercury to Pluto.

b

The masses of the objects (in kg) are (in order of Halley's Comet, Mercury, Venus, ... Pluto, Sun):

{2.2*10^14, 3.3022*10^23, 4.869*10^24, 5.9722*10^24, 6.39*10^23, 1.8988*10^27, 5.685*10^26, 8.6625*10^25, 1.0278*10^26, 1.324*10^22, 1.989*10^30}
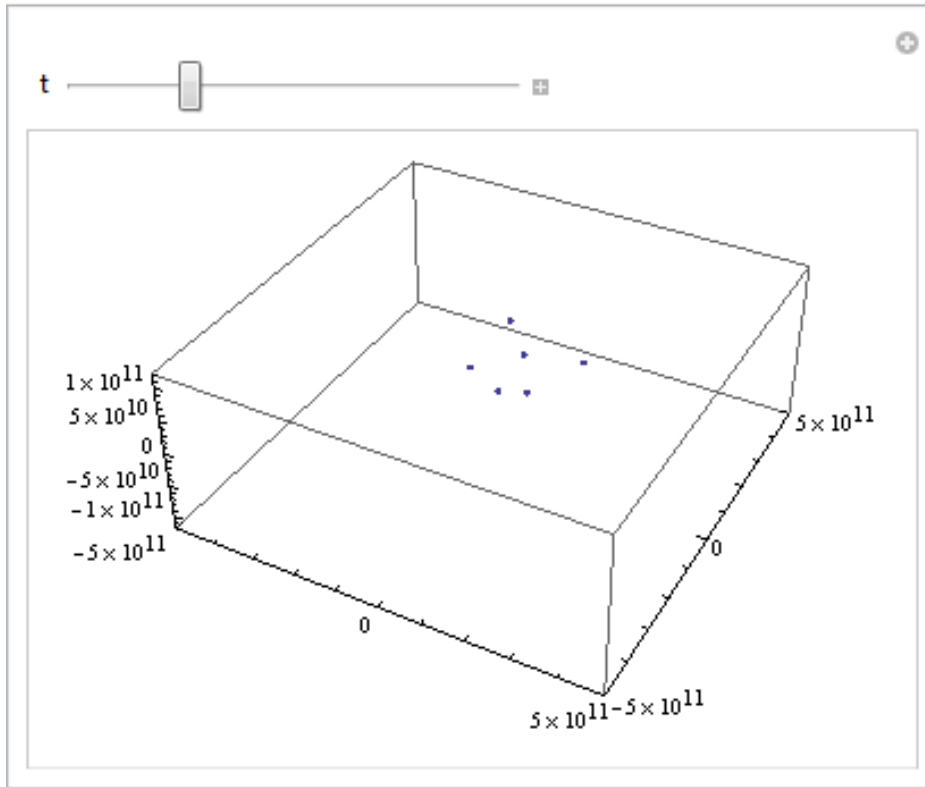
Figure 1: Objects in orbit. (Don't worry, we'll make pretty graph extractions. This one is just low-resolution for space)

Using this, can you calculate the net momentum at the start (first two timesteps which are separated by 1800 seconds) of the simulation (x, y, z are in meters)? Furthermore, test other times to see if total momentum has been conserved (2-3 additional points throughout the simulation is sufficient).

To figure out the instantaneous momentum of one object relative to the reference frame we're in (arbitrary selection of origin with Cartesian coordinates), we want to look at the difference in position between two timesteps $\Delta \vec{x} = (x_2 - x_1, y_2 - y_1, z_2 - z_1)$ then find the velocity by $\vec{v} = \frac{\Delta \vec{x}}{\Delta t}$ and non-relativistic momentum as $\vec{p} = m\vec{v}$. We'll total the momentum of the system by adding the vectors for each object together. You may compare the direction or magnitude (or both) of the momentum to make your judgment about the simulation.