ASTR 400B Homework 2

Due: Jan 23rd 2018 5 PM

In this assignment you will write a code to read in the data file you symbolically linked to your home directory in the first homework, MW_000.txt using Python. We are also going to get used to keeping track of units using AstroPy.

1 Organization of Data File

Each files is organized as follows:

- First Row is the time in units of 10 Myr (equivalent to SnapNumber/0.7)
- Second Row is the total number of particles
- Third Row describes the units of the columns that follows the fourth row
- Fourth Row describes the header name for each column that follows

The remaining rows contain the particle data:

1. First Column *type*: Particle Type. Type 1 = Dark Matter, Type 2 = Disk Stars, Type 3 = Bulge Stars
2. Second Column m : Mass of the particle in units of $10^{10} M_{\odot}$
3. Third-Fifth Columns *x, y, z* : Position (x,y,z) in kpc measured from the center of mass position of the Milky Way
4. Sixth-Eighth Columns *vx, vy, vz*: Velocity (vx,vy,vz) in km/s measured in a cartesian coordinate system centered on the location of the Milky Way

Each column is delimited by tabs (spaces)

2 Create a ReadFile Program

Create a new python script, or Jupyter notebook, entitled *ReadFile*. We will use this code throughout the course, so if you start with a Jupyter notebook, you will need to download the final product as a python script (open the File tab, select Download as Python) so that you can import it in later programs (like Part **??**).

In *ReadFile*, create a function called *Read* that will :

1. open and read the MW_000.txt data file.
2. return the time, and total number of particles as variables (first two lines of the file).
3. return the particle type, mass, x,y,z, vx,vy,vz columns as a data array

To do this, follow these steps in Python (watch your indentation!):

- the first lines of the code import relevant modules: NumPy and AstroPy
  import numpy as np
  import astropy.units as u
- Define the function *Read* that takes the name of the file as input.
  def Read(filename):
- open the file
  file = open(filename,'r')
- Read the first line and store the time in units of 10 Myr. For example:
  line1 = file.readline()
  label, value = line1.split()
  *time = float(value)*10.0*u.Myr*
  Do the same for the 2nd line for the *total* number of particles.
- close the file
  file.close()
- store the remainder of the file using the NumPy function np.genfromtxt. This allows you to use the column header information (4th line of the file, starting with #)

  data = np.genfromtxt(filename,dtype=None,names=True,skip_header=3)

  parameters: "dtype=None" means line is split using white spaces
  "skip_header=3" skipping the first 3 lines .
  the flag "names=True" creates arrays to store the data with the right labels

  Later you can retrieve the data array using the header information. E.g. if you wanted all the masses of all the particles you could call : *data['m'][:]* since the column for mass was called *'m'* as listed in the 4th line of the "MW_000.txt" file.
- return the time, total number of particles and the full data array

3 Create a new program, ParticleProperties, that uses ReadFile

Create a new python script or Jupyter notebook, entitled *ParticleProperties*, which takes as inputs: particle type and particle number. In that code, create a new function called *ParticleInfo*, which returns the following properties for **any given particle of any given type** (Disk, Halo, etc):

1) 3D distance in kpc; 2) 3D velocity in km/s; 3) Mass in units of $M_{\odot}$.
Round all values to 3 decimal places using *np.around(value,3)*.
Since you need to read in the file again, call *Read* from *ReadFile* by adding the following to the list of modules: *from ReadFile import Read*

Hints:

- To read in, say, the x position of *all* particles: *x = data['x']* . Don't forget to assign kpc as the unit using AstroPy (or km/s for velocity).
- To create an index for all particles with a given property, use the NumPy function *np.where*. For example, if I wanted to store all x components of particles that are more than 2 kpc away in the x component, I would write:

  index = np.where(data['x'] > 2)
  xnew = data['x'][index]

4 Prove Your Codes Work!

Use *ParticleProperties* to determine the properties (3D Distance, 3D Velocity and Mass) of the 100th disk particle of the Milky Way for SnapNumber 0. Convert the 3D Distance of the particle to LIGHTYEARS (3 decimal places) using AstroPy (hint: use the "to" function).

5 Homework Submission

- You must DOCUMENT your code . Explain each step.
- Create a directory called Homework2. Save your code and answers to question 4 in that directory. Your answer to question 4 should be saved EITHER:
  1. As part of your Jupyter notebook solution.
  2. Take a screen shot of your python output (with the relevant print statements for each quantity) from the command line.
- Upload your Homework2 directory to your public ASTR400B_yourlastname repository on GitHub