

ASTR 1040 Problem Set 8

The first cell (below) contains lots of helpful constants you may need to use throughout the semester. Make sure to run it each time before you start working!

For reference, all of the `astropy` constants as well as examples can be found [here](#). Likewise all of the units and their names / how to access them can be found [here](#).

To create a variable with units you need to multiply by the corresponding unit class:

```
d = 1 * units.au
```

You can then convert unitful quantities to other units by calling the `to` method and passing the desired end unit class:

```
d_m = d.to(units.m)
```

If you have a ratio of quantities where all the units should cancel out, you can obtain the dimensionless number using the `dimensionless_unscaled` method. For example writing:

```
d_m/d
```

yields $1.4959787 \times 10^{11} \frac{\text{m}}{\text{AU}}$, but doing:

```
(d_m/d).to(units.dimensionless_unscaled)
```

returns `1.0` as expected.

```
In [ ]: #SETUP CELL (modify at your own peril)
from astropy import units #access units by doing units.<unit> (i.e. units.au)
from astropy import constants
import numpy as np #common math functions (i.e. np.sin(x)) and better arrays (i.e. np.array([1,
import matplotlib.pyplot as plt #plotting functions (i.e. plt.plot(x,y))
G = constants.G # gravitational constant
M_sun = constants.M_sun # mass of the sun
R_sun = constants.R_sun # radius of the sun
L_sun = constants.L_sun # luminosity of the sun
M_earth = constants.M_earth # mass of the earth
R_earth = constants.R_earth # radius of the earth
M_jup = constants.M_jup # mass of jupiter
R_jup = constants.R_jup # radius of jupiter
sigma_sb = constants.sigma_sb # Stefan-Boltzmann constant
b_wien = constants.b_wien # Wien's displacement constant
c = constants.c # speed of light
h = constants.h # Planck constant
k_B = constants.k_B # Boltzmann constant
m_e = constants.m_e # mass of electron
m_p = constants.m_p # mass of proton
m_n = constants.m_n # mass of neutron (basically just the mass of a proton but whatever)
g0 = constants.g0 # standard gravity, 9.8 m/s^2
e = constants.e # absolute value of electron/proton charge
```

Problem 1: Recent results from JWST [20 points]

Read this [astrobites article](#), and in a few sentences, summarize what the main result is of the paper described and why that result is surprising astronomers:

This article discusses how new JWST data from the SMACS 0723 galaxy cluster is being used to measure and analyze the morphology of its galaxies, and what this means for our current model of the Universe. What JWST found contradicts the previous results from Hubble; while Hubble predicted that the amount of disk shaped galaxies would only make up about 10% of all galaxies at redshifts between 3 and 6, JWST discovered that

actually, over 50% of galaxies at these redshift values are disks. This data suggests that disk-shaped galaxies have been living in a stable state for much longer than previously thought — over 10 billion years. These new findings challenge the previous model of the Universe, and are forcing scientists to rethink how galaxies are formed, and how galaxy mergers actually work. Understanding galaxy formation is a crucial building block for modeling the Universe, so these new findings are essential in reshaping what we thought we knew.

Problem 2: High- z redshifts and looking back in time [50 points]

Suppose a galaxy observed with a redshift $z = 8$.

a) Back of the envelope age of the universe. [10 points]

Current values for the local Hubble constant H_0 are roughly 70 km/s/Mpc. Use this to provide a rough estimate for the age of the universe, in billions of years.

Hint: Think about Hubble's law and/or the units of H_0

```
In [ ]: H_0 = 70 * (units.km / units.s) / units.Mpc
age = (1 / H_0).to(units.Gyr)
print(age)
```

13.968460309725558 Gyr

b) What is the recession velocity of a galaxy at $z = 8$? [10 points]

Provide your answer in terms of the speed of light, c .

```
In [ ]: # v / c ≈ z == (λ_measured - λ_rest) / λ_rest
# v / c = (1+z)^2 - 1 / (1+z)^2 + 1
# v = H_0*d = a galaxy's recession velocity is proportional to its distance away

z = 8
stretch = ((1+z)**2 - 1) / ((1+z)**2 + 1)
# d = (stretch) / H_0).si
# print(f"v = {v} * c")
# v = (H_0 * (stretch)).si
v = (stretch*c)
print(f"v = {v/c:.4f} * c")
```

$v = 0.9756 * c$

c) According to Hubble's law, how far away was the galaxy when it emitted its light? [10 points]

Provide your answer in billions of lightyears, using $H_0 \approx 70 \frac{\text{km/s}}{\text{Mpc}}$.

Hubble's Law:
$$\frac{v}{c} = \frac{(1+z)^2 - 1}{(1+z)^2 + 1}$$

```
In [ ]: # v is recession velocity from hubble's law equation
# v = H_0 * d
# => d = v / H_0

d = (v / H_0).to(units.Glyr)
print(d)
```

13.627766155829812 Glyr

d) How long ago was the light from that galaxy emitted? [10 points]

Provide your answer in billions of years.

```
In [ ]: # should be pretty close to age of universe (?)
t = (d / v).to(units.Gyr)
print(t)
```

13.968460309725558 Gyr

e) What was the age of the universe when that light was emitted? [10 points]

Provide your answer in millions of years.

Note: We are pushing the limits of relativity and linear calculations with this problem. Our answers are good for a back of the envelope calculation, but in general for $z > 1$ it is a best practice to use general relativity to properly account for the expansion of space over time to correctly answer the question (but that's beyond the scope of this class).

```
In [ ]: # think of distances as "lookback time"
# Hubble constant is a function of time: H(t)

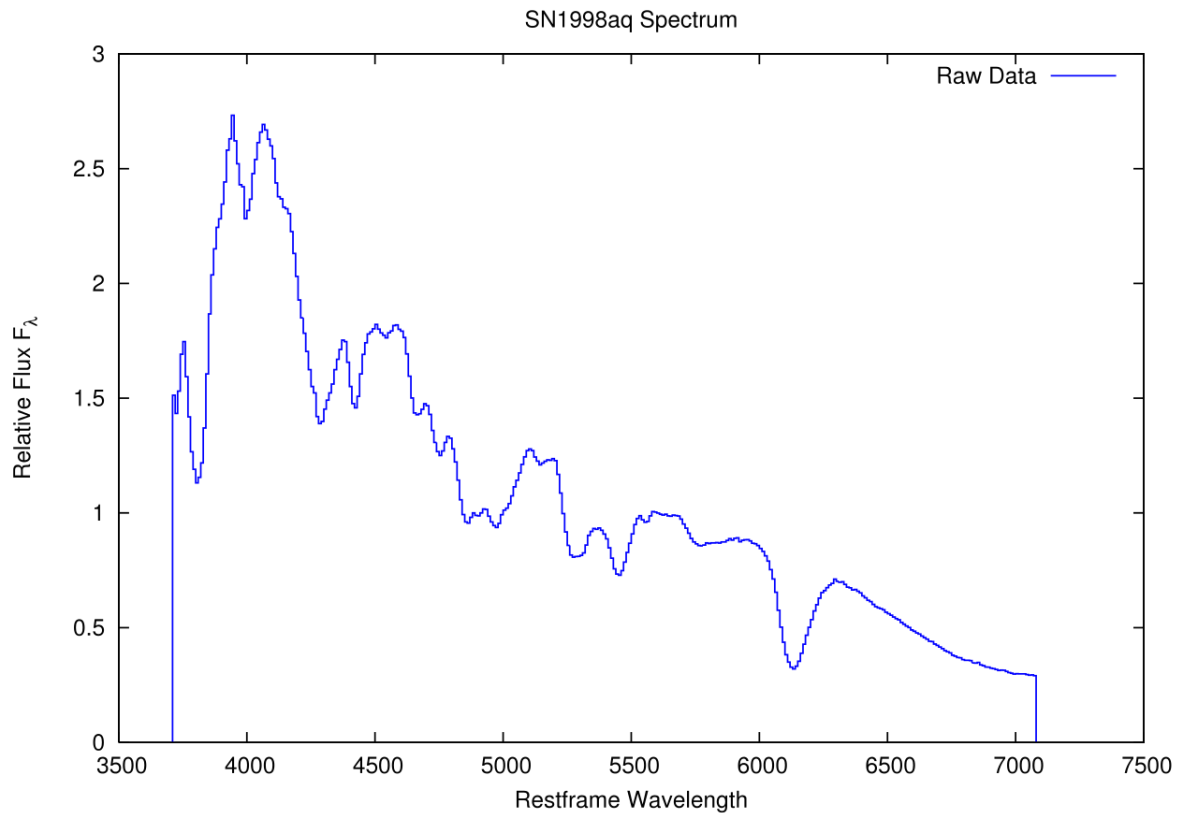
(age-t).to(units.yr)
```

Out[]: 0 yr

Problem 3: Type 1a Supernovae [30 points]

Type 1a supernovae are one of the most valuable "standard candles" in galactic astronomy, because they are bright enough to be detected from billions of light years away! Thus they are the most accurate method to determine the distance to deep-space galaxies.

Below is a plot of a type 1a supernova spectrum taken from real data!



[Link to plot in case it doesn't display in your Jupyter notebook](#)

Fun fact — this supernova had an absolute magnitude of almost -20, which is about the same as the estimated brightness of the entire Milky Way!

a) What type of light is being emitted? [10 points]

Hint: make a rough estimate. Where is the peak of the emission? Note the units of wavelength axis of the plot are in Å, where one Angstrom $10 \text{ Å} = 1 \text{ nm}$.

```
In [ ]: # peaks at about ~4000 Å
λ_rest = (4000 * units.angstrom).to(units.nm)
print(f"λ_rest peaks at about {λ_rest}, which is on the cusp of violet visible light and UV light")
```

λ_rest peaks at about 400.0 nm, which is on the cusp of violet visible light and UV light on the EM spectrum.

b) If we observed this supernova in a galaxy at $z = 2$, what type of light would we receive? [10 points]

What would you expect the peak wavelength of emission to be and what part of the spectrum is this?

```
In [ ]: # z = Δλ / λ_rest_peak
# Δλ = z * λ_rest_peak
z = 2
dλ = lambda z, λ: z * λ
dλ_2 = dλ(z, λ_rest)
print(f"Δλ = {dλ_2}, which is infrared light.")
```

Δλ = 800.0 nm, which is infrared light.

c) Can JWST see faint supernovae? [10 points]

Suppose we want to be able to see type 1a supernovae in at least $z = 10$ galaxies. What wavelength regime should we concentrate on? JWST has an observing range of $0.6\mu\text{m}$ to $28.5\mu\text{m}$ — is this appropriate for observing high- z type 1a supernovae?

```
In [ ]: z = 10  
        dλ_10 = dλ(z, λ_rest)  
        dλ_10.to(units.micrometer)
```

```
Out[ ]: 4 μm
```

At $4\mu\text{m}$, we should focus on infrared wavelengths if we want to observe 1a supernovae in $z = 10$ galaxies. This value falls within JWST's observing range, so it should be capable of observing high- z type 1a supernovae.