Crash Course in Python II

Devansh Agarwal
da0017@mix.wvu.edu
devanshkv.github.io

In today's class

1. Plot the black body curve
    a. Using what we have learnt
    b. Using astropy
2. Planning observations with Astropy

Plotting in python: `Blackbody and Rayleigh Jeans Curve`

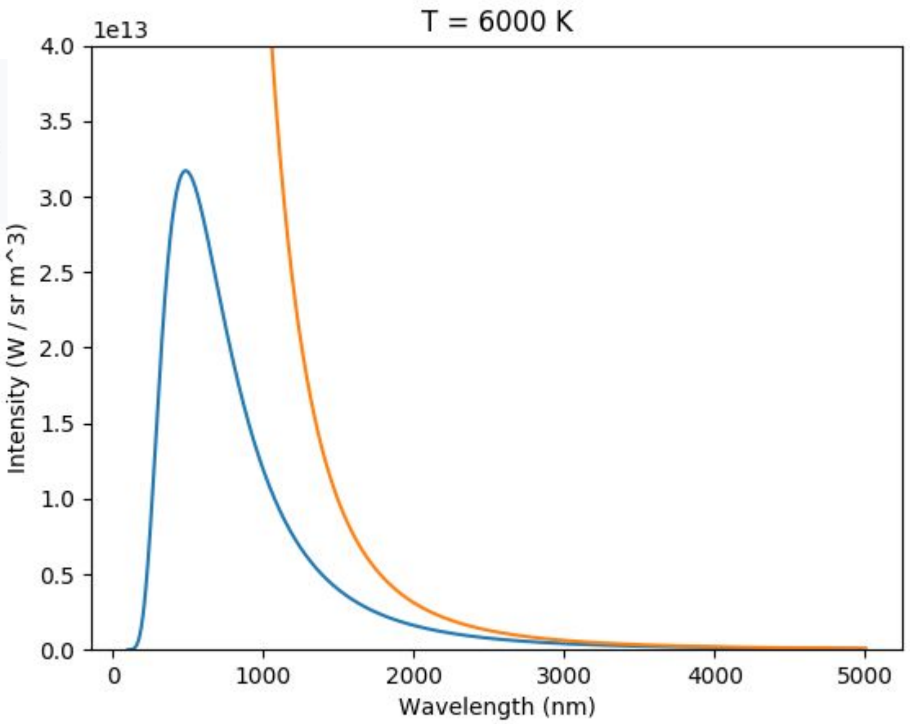
Planck's Law

$$B_\lambda(\lambda, T) = \frac{2hc^2}{\lambda^5} \frac{1}{e^{hc/(\lambda k_B T)} - 1}$$

Rayleigh Jeans Law

$$B_\lambda(T) = \frac{2ck_B T}{\lambda^4}$$

Use: λ = 100--5000 nm



Plotting in python: `Blackbody and Rayleigh Jeans Curve`

```
import numpy as np
import pylab as plt

wavelength_nm = np.linspace(100,5000,1000)

wavelength_m = wavelength_nm*1e-9

#planck constant
h = 6.626e-34 #Js

#speed of light
c = 3.0e+8 # m/s

# boltzmann constant
k = 1.38e-23 # J/K

# Temperature

T = 6000 # K

a = 2.0*h*c**2
b = h*c/(wavelength_m*k*T)
black_body_intensity = a/ ((wavelength_m**5) * (np.exp(b) - 1.0))

RJ_intensity = 2*c*k*T/wavelength_m**4

plt.plot(wavelength_nm,black_body_intensity)
plt.plot(wavelength_nm,RJ_intensity)
plt.ylim(1e8,0.4e14)
plt.xlabel('Wavelength (nm)')
plt.ylabel('Intensity (W / sr m^3)')
plt.title('T = 6000 K')
plt.show()
```

Planck's Law

$$B_\lambda(\lambda, T) = \frac{2hc^2}{\lambda^5} \frac{1}{e^{hc/(\lambda k_B T)} - 1}$$

Rayleigh Jeans Law

$$B_\lambda(T) = \frac{2ck_B T}{\lambda^4}$$

Use: λ = 100--5000 nm

Using: `astropy`

- Blackbody Plotter:
  - `import numpy as np`
  - `import matplotlib.pyplot as plt`
  - `from astropy.modeling.blackbody import blackbody_lambda`
  - `from astropy import units as u`
  - `T = 6000*u.K`
  - `wavelength = np.linspace(100,5000,1000)*u.nm`
  - `black_body_intensity=blackbody_lambda(wavelength,T)`
  - `plt.plot(wavelength,black_body_intensity)`
  - `plt.show()`

http://docs.astropy.org/en/stable/units/

Planning observations: `astropy`

- `import astropy.units as u`
- `from astropy.time import Time`
- `from astropy.coordinates import SkyCoord, EarthLocation, AltAz`
- `crab = SkyCoord.from_name('Crab')`
- `morgantown = EarthLocation(lat=39.629524*u.deg, lon=-79.955894*u.deg, height=271*u.m)`
- `utcoffset = -4*u.hour  # Eastern Daylight Time`
- `time = Time('2019-3-20 17:00:00') - utcoffset`
- `crab_alt_az = crab.transform_to(AltAz(obstime=time,location=morgantown))`
- `print(crab_alt_az.alt, crab_alt_az.az)`
- `print(crab_alt_az.alt.degree, crab_alt_az.az.degree)`

http://docs.astropy.org/en/stable/generated/examples/coordinates/plot_obs-planning.html