CONTACT

The year is 2024. The James Webb Space Telescope just observed the star system Knightro-2019. The system was already known to host a giant planet Knightro-2019a. The system turned out to have two more planets, which following standard planetary naming practice were named Knightro-2019b and Knightro-2019c. Knightro-2019b and Knightro-2019c are in the habitable zone of Knightro-2019, i.e., the zone where conditions are correct for these planets to support life. The first observations show hints that one of these planets may even have life on it! Can it be that we will soon be able to make first contact with another civilization?

Fig 1. The Arecibo Radio Telescope managed by UCF.

In a weird twist of events everyone that was able to characterize these observations vanishes and you are humanity's only hope to know the truth. Your mission, should you chose to accept it, is to create a code that can model the signal of a planet. Then you will fit the observations of Knightro-2019b and c, to retrieve the planetary properties (radius, distance to Knightro-2019, surface and cloud properties) and let scientists at the Arecibo Radio Telescope know if they should use the telescope to 'listen' for intelligent life on Knightro-2019b or c.

MISSION.

0. (**5 points**) Let's start by setting up our environment.

(a) Remember that good coding practices are taken into account in your grade. Write useful comments in each part of the code. Start the file by writing your name, NID, course number, assignment number, and the date should appear as comments at the top. Put the problems numbers in comments, as well as any remarks or written answers you may make. Print the problem number (as in "Problem 1:") on a line by itself before each problem's output. Use the `print()` function to print, don't just type the expression, as we will run these non-interactively.

(b) Import the necessary packages for what we will be doing: `numpy, scipy, matplotlib and pickle`

(c) Import the appropriate `scipy` function(s) for chi-square minimization.

(d) Download all files in the demos/final_project/ folder

(e) Make a directory called `<classdir>/handin/final_project_<username>/`. This is your "homework directory" for this assignment. Make this directory your current directory. Go into your homework directory `final_project_<username>`. Start a Git repository and keep under revision control the content of your `final_project_<username>/`. Every time you make a change to the folder content (including the content of your main homework file) commit it with a useful, informative comment. Make a Python file named `final_project_<username>.py` and save it here (in your `final_project_<username>/` folder).

1. **The star** (**20 points**): When we image a planet like Knightro-2019b and c, what we observe is a mix of the parent star signal and the planetary signal.

The spectrum of a star can be well approximated using the assumption that the star is a black body. A black body is an idealized object that emits thermal radiation in a continuum spectrum (i.e. at all wavelengths) according to its temperature. According to Planck's law a black body (here our star) of temperature T emits energy per unit wavelength λ of:

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

where h is Planck's constant, c the speed of light and k_B is the Boltzmann constant. Make a function called `star(T, resolution)`, that creates the stellar spectrum for a star of given temperature T (in K) between 0.5μm and 10μm. The resolution of the data shows how finely spaced your wavelength array is and is defined as `resolution=λ2/(λ2-λ1)`.

a) (**5 points**): Call the function `star` for temperatures of 2,000K, 5,000K, 6,000K, 7,000K , 8,0000K and 10,000K and a resolution of 100. Make a single plot that shows the energy emitted by the star at all these temperatures. Test that the plot makes sense using whichever method you prefer, and explain how you tested it. The plot should follow the standard rules we discussed in class to make it "publication ready". Write a few sentences about what you notice from this plot.

b) (**15 points**): Now that you know what an ideal star spectrum looks like, get the 'actual' spectrum of Knightro-2019 from Webcourses/homework/final_project/Knightro-2019.spec.

I) (7 points) Import the data in a variable which has an appropriate, descriptive name. Plot the spectrum of Knightro-2019. What differences do you see with the ideal spectrum?

II) (8 points) Fit the spectrum of Knightro-2019 with the spectra of black bodies at 3,000K, 4,000K 5,000K, 6,000K, 7,000K and 8,000K. Note that the observed spectrum is in [erg/s/cm**2]. What is the temperature of Knightro-2019? How did you come to this conclusion?

Tip: If you want to avoid unit conversions, you can do any $1/\lambda$ conversions needed because they will affect the shape of your spectrum, and then you can normalize the spectra (so, divide each spectrum by its maximum).

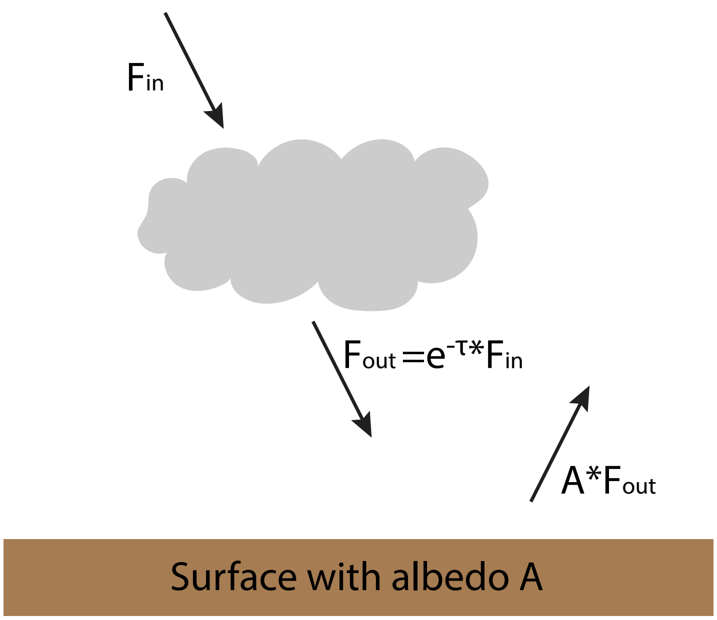

Fig 2. Clouds and the surface of a planet will interact with the incoming star light and reduce the light reaching the observer

2. **The planet** (**38 points total**): The light emitted by the star will travel toward the planet and it will interact with its atmosphere and surface before reaching us on Earth. Since we do not know what Knightro-2019b and c are made of we will need to model a planet that is as generic as possible. We can later fit the observations with different models and test which parameters best fit the observations to decide what Knightro-2019b and c look like.

In this section of the Project you will get a basic idea of all the information that a spectrum contains.

a) Surface (**10 points total**): A surface will reflect and absorb light that falls on it. How much light it reflects is defined by the surface albedo $A \in [0,1]$. If incoming flux F$_{in}$ falls on a surface of albedo A, the outgoing flux will be $F_{\text{out}}(\lambda) = A \times F_{\text{in}}(\lambda)$. The flux absorbed by the surface would be $F_{\text{abs}}(\lambda) = (1 - A)F_{\text{in}}(\lambda)$.

I) (**2 points**) Make an expression that calculates how much light is reflected by a surface of albedo A when it is hit by incoming flux $F_{\text{in}}(\lambda)$.

II) (**8 points**) Create an array `lambda` that contains values from 0.5 to 10 with a step of 0.1. Make a single plot showing the outgoing flux for albedos of 0.1, 0.3 and 0.6, when the incoming flux is the 'actual' spectrum of Knightro-2019 from the previous section. Overplot (so plot in the same plot) the 'actual' spectrum of Knightro-2019.

(**10 bonus points**) In reality albedo is a function of wavelength, not a constant. What happens if A changes as a function of wavelength? Read `forest_surface.dat` into an appropriately named array. This is the albedo of a forest surface at different wavelengths. Interpolate the spectrum of Knightro-2019 to the the forest albedo wavelength grid. In one line calculate the flux of Knightro-2019 that escapes to space after it gets reflected off a forest surface at all wavelengths.

Make a plot that shows the incoming flux from Knightro-2019, the flux reflected by a surface of albedo 0.3 and the flux reflected by the forest surface.

b) Atmosphere (**28 points total**): An atmosphere has gases and clouds that interact with the light. The way this happens in reality is pretty complex, and modeling it accurately is outside the scope of this class. Here will make some oversimplifications to get a toy model of the atmosphere's behavior.

I) Clouds (**4 points**). Clouds scatter and absorb light at all wavelengths. In the most simple treatment you can model a cloud as a 'gray' absorber, i.e., a material that absorbs light at all wavelengths in the same way. The cloud property that defines how much light will be absorbed, is the optical thickness $\tau(\lambda)$. If an amount of light F_0 enters a cloud of optical thickness τ then $F_0 \times e^{-\tau(\lambda)}$ will come out on the other side. When we treat a cloud as a gray absorber, we ignore the wavelength dependence of its absorption so we get: $F_0 \times e^{-\tau}$.

Create an expression that will give you the amount of light F_{out} coming through the other side of a cloud with optical thickness τ, for incoming amount of light F_{in}. Create a tuple (not a list and not an array!) `tau` that contains the optical thicknesses of a possible cloud. Make tau contain all numbers from 0 to 100 with a step of 0.5. Use the expression you created to print the outgoing flux at 0.7μm and 1.1μm, after the light of

Knightro-2019 passes once through a cloud of optical thickness `tau` (for all values of `tau`).

II) (**6 points**). Make a function `base_model` that calculates the outgoing light (to space) from Knightro-2019 b, after the light of Knightro-2019 passes through a cloud of optical thickness τ, reflects on the surface of albedo A and then either passes through another cloud of optical thickness τ_2 or reflects off the cloud (assume perfect reflection), then reflects on the surface with albedo A and then again either passes through another cloud of optical thickness τ_2 or reflects off the cloud….This process can repeat up to 5 times. On the 5th time the light *will* pass through the cloud of optical thickness τ_2 and leave to space.

III) (**6 points**) Use the function `base_model` to calculate the outgoing light (to space) from Knightro-2019 b at 0.5μm, after the light of Knightro-2019 passes through a cloud of optical thickness 12, reflects twice on the surface of Knightro-2019b which has an albedo A = 0.3, and then passes through a cloud of optical thickness of 10 before it escapes to space.

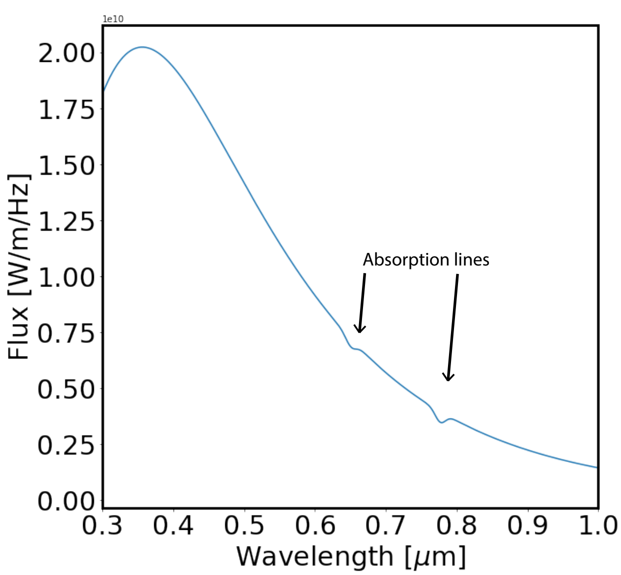

Fig. 3 Reflected light off a planet with O_2 in its atmosphere.

IV) (**12 points**). Gases in an atmosphere absorb light. To see how it works let's create a simple Gaussian absorption profile for the O_2 A and B bands in the atmosphere. Note that the O_2 A band is centered around 0.765μm and B band around 0.688μm.

Create two Gaussian distributions: one that is centered around the center of the A and one around the center of the B band. Use a small sigma (for example, 4) that gives you a narrow distribution. Scale the Gaussians to peak at 0.12 and 0.05.
Since these will be your absorption lines, now you will have to invert them (so do: 1-x).

Create the 'response' function F_{res} of a clear atmosphere to light passing through it, assuming that at all wavelengths outside the two O2 lines light is not affected (F_{res} = 1) and in the O2 lines F_{res} = inverted_Gaussians. Now use F_{res} and the black body model at 6,000K to create a figure like the one shown in Fig. 3.

3. **Properties of the planets** (**60 points total**) By now you should have understood the basic features of planetary spectra. Let's go on to characterize the observations of Knightro-2019b and Knightro-2019c. We will start with retrieving the radius and distance of the planets from their parent star.

a) **Radius of our planets** (**10 points**) From Webcourses/homework/final_project/ use the file planet_lightcurves.dat This file contains the observations of transits of Knightro-2019 b and c. It may be a good idea to first plot the data to see the difference between b and c. We observed two transits of each planet. Knightro 2019 b has the shallower dips and c the deeper dips.

Knowing that the depth of a transit `d` (so the difference between the continuum and the minimum of light you get from the system) is related to the radius of the planet with: $d = \frac{R^2_{\rm planet}}{R^2_{\rm star}}$ and that the star Knightro-2019 has a radius of 0.95 solar radii, find the radius of Knightro-2019 b and c in Earth radii.

b) **Orbital properties of the planets** (**10 points**) Knowing that the minimum of a transit happens every time a planet crosses from the center of Knightro-2019's stellar disk use the light curves to find the orbital period of the Knightro-2019 b and c in days. Use Kepler's 3rd law (as in Homework 7) to find the orbital distance of b and c from their parent star.

We now know that Knightro-2019 b and c are terrestrial planets and orbit their parent star in the right distance to be habitable! It finally makes sense to test if their spectra suggest that they have life. If so, we should let the Arecibo know that they should listen for signs of civilization from Knightro-2019b or c!

c) **The characterization of the planets** (**40 points total**) Use the pickle `model_runs.pickle`. This pickle contains a dictionary with all models that you run with different surface properties and cloud properties. The keys are the model names (they include surface name, species that absorbs if applicable (h2o, o2) and cloud optical thickness if applicable (t10 `==` $\tau = 10$ etc)).

I) (**5 points**) Read the pickle in a dictionary with an appropriate name.

II) (**5 points**) File `knightro_2019_bc_photometry.dat` are our spectral observations of the two planets. Read the observations in a numpy array with an appropriate name.

III) (**20 points**) Knowing that the observations can be fitted by a mix of up to 5 of your `model_runs.pickle` models in the shape: $F_{total} = \Sigma_i(a_i * F_i)$ with $\Sigma_i(a_i) = 1$, use the chi-square minimization criterion to find the best-fit mix of models for the observations of Knightro-2019 b and c.
Note that the observations are in a lower resolution than the model. To compare the observations with the model mix you will need to use the appropriate model wavelengths!

IV) (**10 points**) Using the best-fit mix of models you found for Knightro-2019 b and c which planet, if any, should Arecibo listen to for signs of a technologically advanced civilization (assume that signs of water and/or forestation on the planet equals the existence of life on it).

4. (**10 points**) Print the Git message log. Copy into your log.

5. (**10 points**) Prepare and submit your homework. Write the commands to make and submit the zip file into your log and test them. When satisfied, close the log, copy it to your homework directory one last time, and run the commands to make and submit the zip file. **Make sure you don't submit an empty zip file by using the appropriate Unix commands to check the content of your zip**. Turn the file in on WebCourses.