Homework 2

Part 1: Using the Saha and Boltzmann Equations

Get started on your program by setting up the initial statements. You will be running the Saha equation and calculating the log10 of the number of ionized versus neutral hydrogen atoms. This uses the two provided model atmospheres, needing only the temperature and log10 of the electron pressure for a given layer.

The Saha equation to be solved, which has been rearranged a bit from what is in the slides, is:

$$\log_{10}\left(\frac{N_{II}}{N_I}\right) = \log_{10}\left(\frac{Z_{II}}{Z_I}\right) + \log_{10}(2) + \frac{5}{2}\log_{10}(T) - \chi_{ion}\Theta - \log_{10}\left(P_e\right) - 0.4772$$

A couple useful clarifications:

$\Theta = 5040/T$ [K]: this is the "inverse temperature", a classical spectroscopic shorthand that folds in the Boltzmann constant in units of [eV/K].

χ_{ion} = 13.595 [eV]: this is the ionisation energy of neutral hydrogen.

The 2 data files you need are: `sun-atmosphere-kamp.txt`, and `vega-atmosphere-kamp.txt`, both available on Canvas. Each file represents a model stellar atmosphere and contains the electron pressure and inverse temperature at different depths/layers within the star. The first row of each file is a list of Θ values, the second row is a corresponding list of $\log_{10}(P_e)$ [erg cm^{-3}]. These are what is needed for the Saha equation above.

You'll also need Chapter 8.1 from *An Introduction to Modern Astrophysics* (or "BOB") by Carroll & Ostlie (2nd ed); the excerpt is also available on Canvas if you don't have a copy. Please read this excerpt carefully before continuing.

(5pts)
You must determine the partition functions for ionised (Z_{II}) and neutral (Z_I) H. Since we're working with hydrogen, we can take a couple shortcuts. Remember from the BOB excerpt that there's no degeneracy in a hydrogen ion (put another way, there's only one way to arrange a hydrogen atom's electrons). We can also note that the energy of the first excited state of hydrogen is E_2-E_1 = 10.2 eV above the ground state energy, and 10.2 eV >> kT for the temperature regime of stars. With these shortcuts in mind, state clearly what your partition function ratio Z_{II}/Z_I is and why.

(20pts)
Your program should:
Read in rows of data from both files (together or separately)
Solve the Saha equation for each layer (i.e. each column of the provided files)
Create two graphs (or one overplay) with clear labeling:

- One showing the $\log_{10}(N_{II}/N_I)$ vs. T for a Sun-like star
- One showing the $\log_{10}(N_{II}/N_I)$ vs T for a Vega-like star

Part 2: Hα Absorption vs. Temperature

Recall, stellar spectral type letters were originally determined using the strengths of Hα absorption lines. However, Annie Jump Cannon famously determined the "correct" order, arranging the letters by descending surface temperature (OBAFGKM), that is still used today.

Your goal here is to recreate several figures explaining why A-type main sequence stars have the strongest Hα absorption lines based on their T_{eff} (remember that A stars have a typical T_{eff} range of ~8000-10000 K). The figures you're recreating (shown below) are the same as figures 8.7, 8.8, and 8.9 from BOB Ch 8.1.

You can reuse the simplified Saha equation from Part 1 and assume a fixed electron pressure of P_e = 20 N/m^2 (though mind the units). Evaluate each plot using a range of T_{eff} values from at least 5000K to 25000K in 1K increments.

(15pts) Recreate Figure 8.7, showing $N_2/(N_1 + N_2)$ as a function of T_{eff}.

(15pts) Recreate Figure 8.8, showing N_{II}/N_{total} as a function of T_{eff} (note the simplifying assumptions that the book makes!)

(15pts) Recreate Figure 8.9, showing N_2/N_{total} as a function of T_{eff}. Since electrons must be in the first excited state to undergo Balmer absorption, this ratio gives the fraction of atoms that are available to produce Hα absorption (n=2 to n=3) in the atmosphere.

Turn in your write-up, including the labeled plots, as a PDF. Remember to include an attribution for any group work! Also turn in your code or Jupyter notebook used to solve the assignment. Note: we'd like to be able to run your code to check that it actually works, so be sure (if using Jupyter notebooks) to check that it runs "top down"!