

Exercises - 3

1. Produce the figure below for the parabola $y = -2x^2 + 3x + 5$.

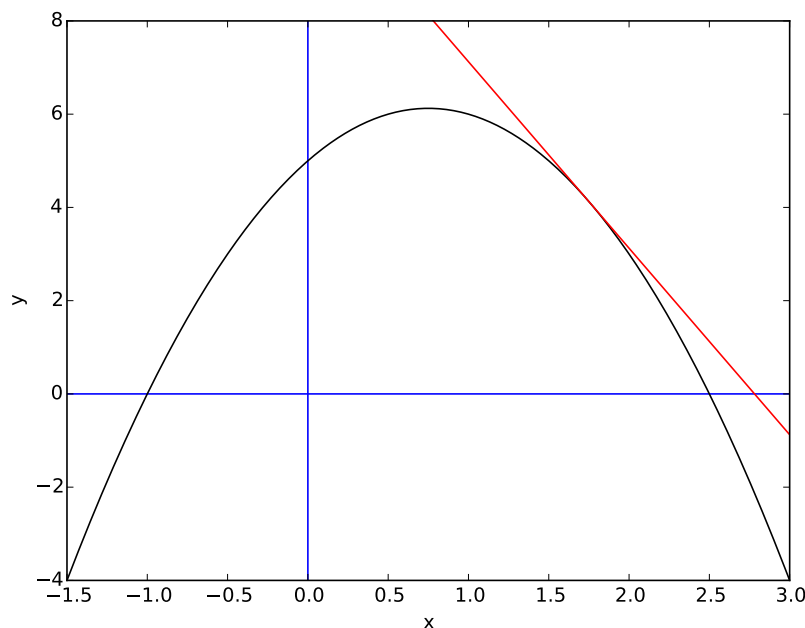


Figure 1: A nice figure

In addition to plotting the parabola, also plot $y = 0$ and $x = 0$ lines, as well as the tangent line to the parabola at $x = 1.75$.

2. Plank showed that the spectral radiance of a body at absolute temperature T is given by

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

Plot the black body curves of bodies with temperatures $T = (6000, 5000, 4000, 3000)$ K over the wavelength range $0.1 - 3 \mu\text{m}$. For each curve, also plot a vertical line at the location of λ_{max} .

3. Given a data set that is "nicely organised", the easiest way to read it in is by using `np.loadtxt`. Go to my ftp site and download the `HI_properties.txt`

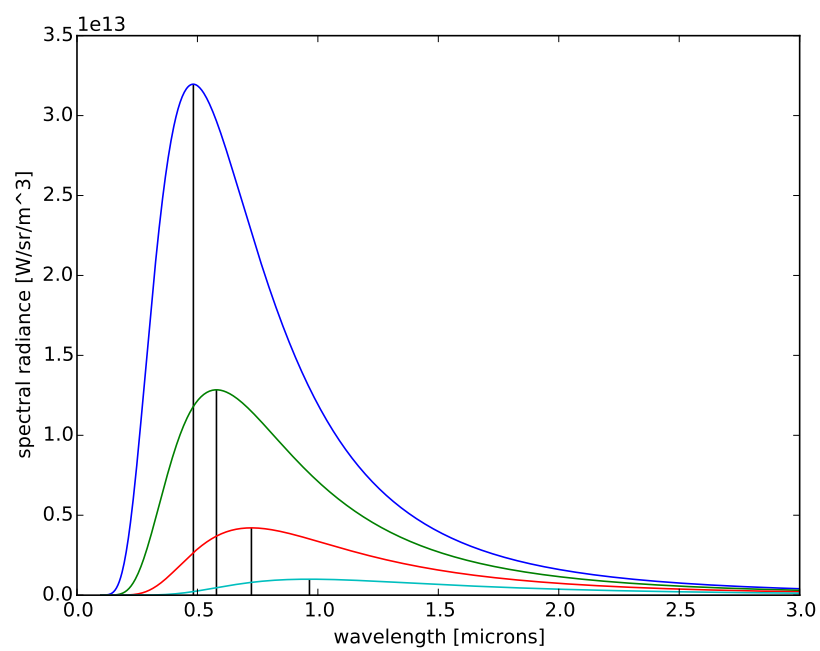


Figure 2: Black body curves

file. It's the data from one of the tables in the PhD thesis of Rob Swaters. Read it in and then plot HI mass against W_{50} . Produce a scatter plot. Also produce a histogram of the HI masses. Then, print to screen the entire data base, but arrange the rows according to increasing HI mass. In fact, write the sorted catalogue to file. There's a NumPy function that will allow you to do this in a single line.

4. Take a look the excerpt from Bram Stokers Dracula (on my ftp site). This particular excerpt constitutes part of the journal of Jonathan Harker, one of the main protagonists. Write some code to count up the number of vowels in the excerpt.
5. Simulate a game of Snakes and Ladders! Look at the board below. Write some code that will simulate the throws of a set of dice (sum = 12). Starting at square 1, start "moving" along the squares. If you land at the base of a ladder, then "move" to the top of it. If you land at the mouth of a snake, then "move" to the snake's tail. Count up the number of throws needed to past 100 (top left of the board).

Repeat the simulation 10 000 times and plot the distribution of the number of throws required to finish the games.

TIP 1: You can either use a for loop together with an if statement to check that you've not gone past 100, or you can use a while loop.

TIP 2: Each time you simulate a single game, you'll get the number of throws required to complete that game. Collect all of those numbers into an array of length 10 000. Then, at the end of your code simply histogram that array.

