ENGN 2020:

Homework #9

Brown University School of Engineering

Assigned: April 22, 2019, Due: April 29, 2019

Problem 1

Here, you will implement the simple Metropolis algorithm, as presented in class, to gather statistics on the average bond length of an H_2 molecule. The data for this problem[1] are available on Canvas and are shown in Figure 1. The script below loads the data and creates an interpolation function (`get_energy`) that you can use in your implementation, plus creates Figure 1. Note that this function returns $+\infty$ for values less than the minimum of the data set and 0 for values greater than the maximum.

File attached here.

```
import numpy as np
from scipy.interpolate import interp1d
from matplotlib import pyplot

# The below creates an interpolation function, get_energy(r).
data = np.load('h2-data.npz')
get_energy = interp1d(data['bondlengths'], data['energies'],
                      fill_value=(np.inf, 0.), bounds_error=False)

fig, ax = pyplot.subplots()
ax.plot(data['bondlengths'], data['energies'], 'o')
ls = np.linspace(0., data['bondlengths'][-1] * 1.1, num=1000)
es = [get_energy(_) for _ in ls]
ax.plot(ls, es, 'r-')
ax.set_xlabel('bond length, $\AA$')
ax.set_ylabel('energy, eV')
fig.savefig('interp.pdf')
```

The probability of the H_2 molecule having any bondlength l is given by the Boltzmann distribution, and is proportional to:

$$P(l) \propto \exp\left\{\frac{-E(l)}{k_B T}\right\}$$

[1] As calculated with quantum mechanically in density functional theory with the package NWChem.

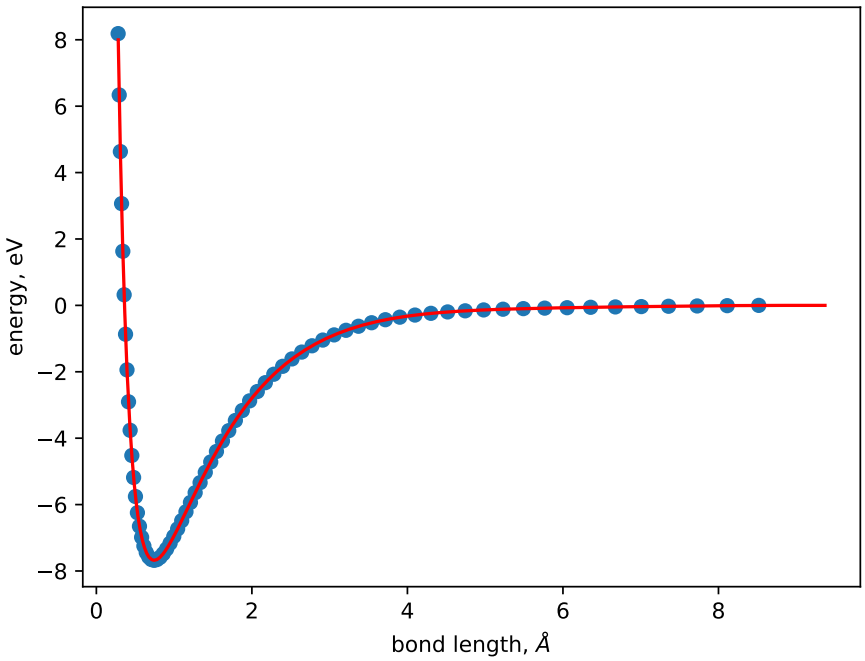

Figure 1: Bond length versus (potential) energy for H_2.

where k_B is the Boltzmann constant, T is the temperature, and $E(r)$ is the energy (given by `get_energy`).

(a) Write an algorithm that implements the Metropolis algorithm as discussed in class. That is, for each random proposed step, accept it if $r = P_{\text{new}}/P_{\text{old}}$ is greater than 1, otherwise accept it with a probability r.

Hint: Within your code, it is better to work in terms of $\log P$ than P, because P can take on very extreme values.

Initially, use a temperature of 300 K and a maximum step size of 0.1 Å. For a run of 500 steps, create the following plots:

- Average l (for the ensemble) versus step number.
- Standard deviation of l (for the ensemble) versus step number.
- Fraction of steps accepted versus step number.
- Final histogram of l after 500 steps.

(b) The maximum step size parameter controls how your algorithm performs. However, for long runs, your algorithm should always converge to the same statistics (that is, the shape of the final histogram). Show examples of how quickly your algorithm converges when you change the maximum step size. In words, describe what happens if this parameter is too large or too small. Roughly, what is the optimum value?

(c) We will say that the bond has broken, and H_2 has broken apart into two H atoms, if the bond length exceeds 2 Å. Use your algorithm and make a plot of the fraction of your ensemble that has $l > 2$ Å across the temperature range from 300 to 1500 K.

Problem 2

Beers 8.A.4. However, use *sampyl* to perform the Bayesian analysis. (Ignore Beers' remark about a quadratic expansion.) It is recommended that you first work through the *sampyl* tutorial at the below link.

`http://matatat.org/sampyl/tutorial.html`