

Boltzmann Distribution and Probability

March 31, 2023

```
[1]: import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
import mpld3
mpld3.enable_notebook()

[2]: E = [12,122,12222] # these are the possible energy states that can be occupied
T = np.linspace(0.01,20000,1000) # This is temperature range and resolution

[3]: # Function of  $e^{-E/RT}$ 
def Boltzmann_Probability(E,T):
    R = 8.3144
    return np.exp(-E/(R*T))

[4]: # Canonical Partition Function
def Q(E,T):
    q = 0
    for i in range(len(E)):
        q = q + Boltzmann_Probability(E[i],T)
    return q

[5]: # Probability of Energy States
def probability_of_an_energy_state(S, E, T):
    return "The probability of the specified energy state is",\
    Boltzmann_Probability(S,T)/(Q(E,T))

[6]: # Probability of All Energy States outputed per energy state
def all_probabilities(E, T):
    for i in range(len(E)):
        print("The Probability for Energy State", i , "is",\
        probability_of_an_energy_state(E[i],E, T)[1])

[7]: def graph(E):
    for i in range(len(E)):
        plt.plot(T, (probability_of_an_energy_state(E[i],E, T))[1])
        plt.title("Probability of Each Energy State as a Function of\
        Temperature")
```

```
plt.xlabel("Temperature in Kelvin")
plt.ylabel("Probability")
```

0.0.1 Set Temperature Range and Energy States,

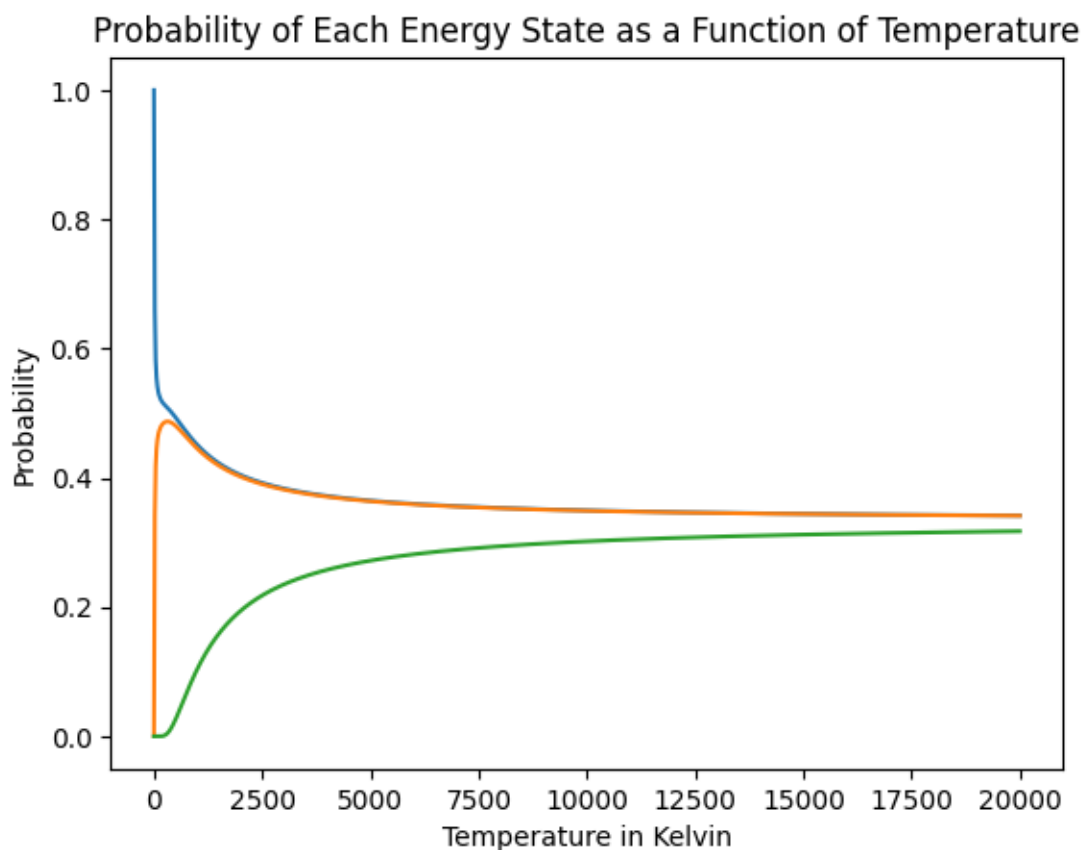
```
[10]: E = [12,122,12222] # these are the possible energy states that can be occupied
      T = np.linspace(0.01,20000,1000) # This is temperature range and resolution
```

1 Examples

1.1 Graphing

1.1.1 You can graph the function by just specifying the list of energy states the molecule can occupy. The interactive graph function of the matplotlib library is enable by default to allow scrolling.

```
[11]: graph(E)
```



1.2 Find out the specific probability of an energy state at a specified temperature

1.2.1 First type the specific energy state you desire, if you know its index in a list you can use that. Then the reference list of energy states, then the temperature. If you just desire the number then index it like the next example

```
[12]: probability_of_an_energy_state(E[1], E, 200)
```

```
[12]: ('The probability of the specified energy state is', 0.4833068575105357)
```

```
[13]: probability_of_an_energy_state(E[1], E, 200)[1]
```

```
[13]: 0.4833068575105357
```

1.2.2 If you want a list of all the probabilities, then and in the energy states then the temperature in which you desire the list of probabilities

```
[14]: all_probabilities(E, 300)
```

```
The Probability for Energy State 0 is 0.5090765606791284
```

```
The Probability for Energy State 1 is 0.4871140202502623
```

```
The Probability for Energy State 2 is 0.0038094190706092464
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```