Exercise 1. What is the universe made of?

WMAP's website published an interactive simulation that allows you to investigate the effects of the different cosmological parameters on the primordial power spectrum curve.

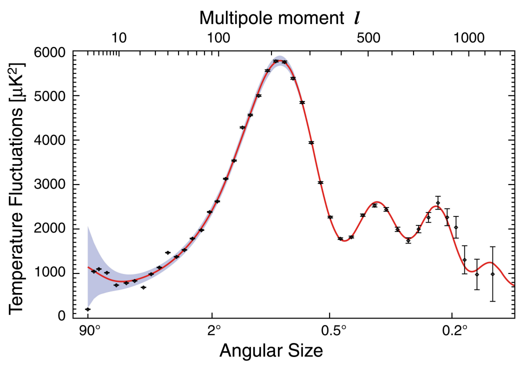

Figure 1: The y-axis represents temperature fluctuations and the x-axis is the multipole moment (in spherical harmonics), or inversely the angular scale (small x corresponds to large scales which are subject to cosmic variance). The red curve represents the best fit curve to the experimental points.

The key question today is, "How do we estimate the correct parameters of a simulation if we only get to observe the results (output) of the simulation?" You do not necessarily have to understand the physics to do this exercise. You are essentially doing a "manual" Monte Carlo parameter estimation. Refer to the study guide on the meaning of the different parameters contemplated in the simulation, if you would like a refresher.

Task: Adjust the 6 parameters of the WMAP simulation to obtain an overlap between the simulation curve (in blue) and the observational curve (in red). What are the final values of each parameter for a 'perfect' fit? Record your parameter values.

Exercise 2. What is the protactinium half-decay?

Proctatinium is a radioactive isotope that decays exponentially with time, whose available quantity at a given instant in time follows

$$N(t) = N_0 \exp(-\lambda t) \ , \tag{1}$$

where N_0 is the initial amount of material and λ is the decay constant. In this exercise we will focus on a **non-MCMC** approach for determining the half-life of proctatinium. The theoretical value is 1.17 minutes. Do the following:

1. Download experimental data for proctatinium decay. You may wish to read and store the data in a pandas data frame. Here is some code to get you started. The `url` variable should be the data URL provided above.

```
import pandas as pd
data = pd.read_csv(url)
t_values = np.array(data["time"], dtype=float)
count_values = np.array(data["count_rate"], dtype=float)
```

2. Load the data into Python and plot it. You should get something that looks like this.

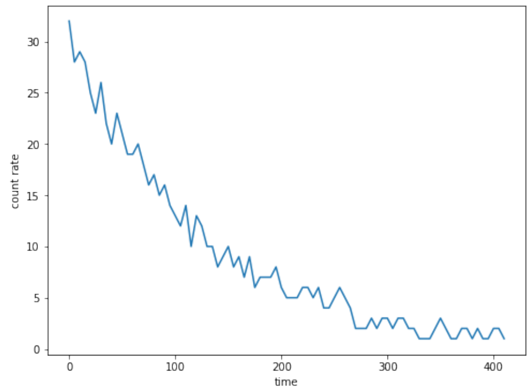

Figure 2: Radioactive decay data for the pre-class work.

3. Again, the key question today is, "How do we estimate the correct parameters of a simulation if we only get to observe the output of the simulation?" To find the half-life of proctatinium, you will run 10,000 trials, and start with an initial guess $\lambda = 1$. Since experimental results are subject to measurement error, add Gaussian random noise with mean 0 and standard deviation 1 to all the $N(t)$ readings.

4. For each trial, fit the noisy data to the function

$$N(t) = 32 \exp(-\lambda t) \ , \tag{2}$$

and find the best value of λ by using the `scipy.optimize.curve_fit()` function. Notice that this equation means $N_0 = 32$. The `curvefit` function takes 4 inputs, namely the function to fit $(N(t))$, the x-values of the data, the y-values of the data, and the initial guess for the parameter (λ). Record the best-fit value for λ at the end of each trial.

5. Plot a histogram of best fit values for λ.

6. Report on the error in the experimental estimate for the half-life of proctatinium by using the mean best fit for λ and the theoretical value of 1.17 minutes.

7. Why is this **not** an MCMC method?