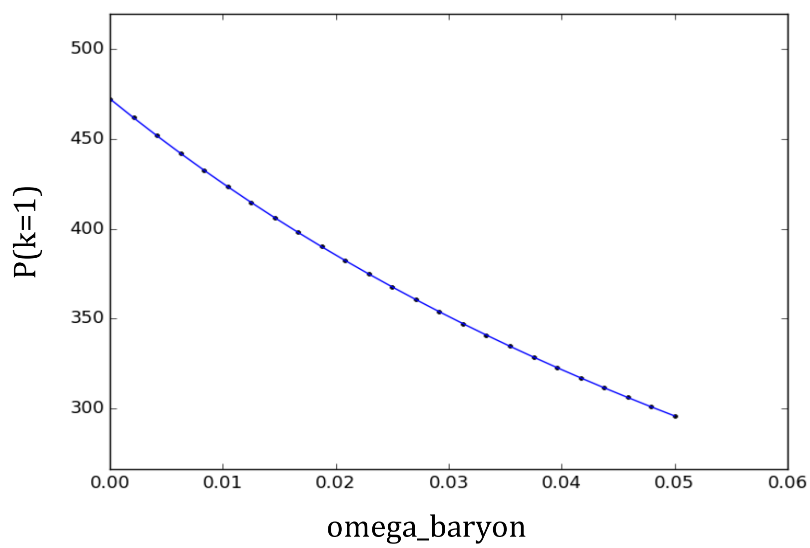
Computing Numerical Derivatives

1. Run axionCAMB for many values of a parameter (~25) around the central value.

2. Plot P(k) vs. parameter values for every k.



3. Fit a polynomial to every k using

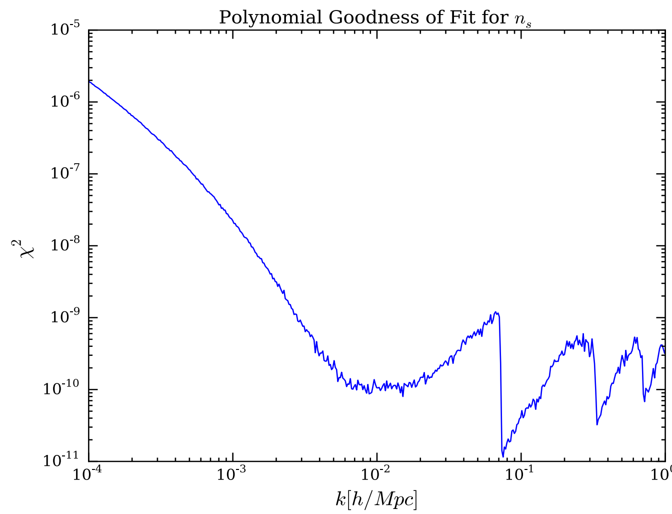
   ```
   np.polyfit(x,y,deg)
   ```

   which outputs the coefficients of the fit (highest order is index 0).

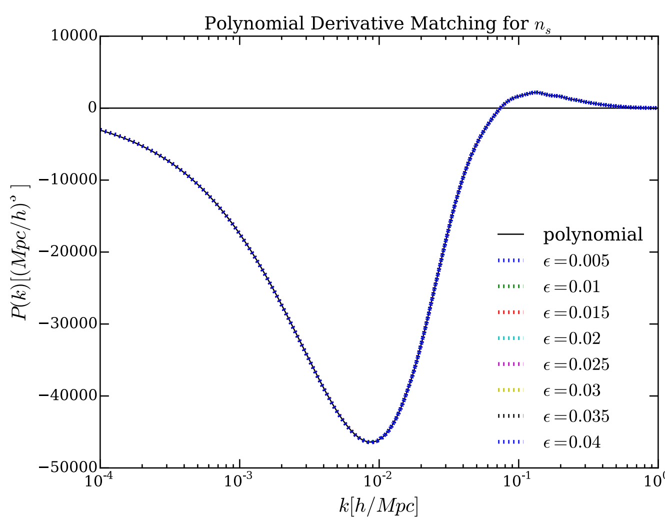
4. Show that the polynomial fits the axionCAMB data well by computing a χ^2 test

$$\chi^2 = \frac{\sum_i (y_i - f_i)^2}{\sum_i (y_i - \bar{y})^2}$$

   where y_i are the polynomial fit points, f_i are the axionCAMB data points, and $\bar{y}$ is the average of all y_i .



5. Plot the coefficient of the first order term from each polynomial vs. the corresponding k-values to construct a polynomial derivative of the power spectrum.



6. Run axionCAMB varying every parameter by multiple step sizes and compute a finite difference derivative. The definite epsilon version is shown below.

$$\left.\frac{\partial P(k;p_n)}{\partial p_i}\right|_{p_n=p_{n,0}} = \frac{P(k;p_{i,0}+\epsilon) - P(k;p_{i,0}-\epsilon)}{2\epsilon}$$

7. Compute a χ^2 test between the polynomial derivative and the finite difference derivative for all step sizes.