Q1) Based on the chi-square value, are the parameters dialed into my test script an acceptable fit? What do you get for χ^2 for parameters equal to [69,0.022,0.12,0.06,2.1e-9,0.95]? Would you consider these an acceptable fit?

The χ^2 value from the test-script parameters is 15267.937968222595. Given the standard deviation of χ^2 is $\sqrt{2n} = \sqrt{2(n_d - n_m)}$ for n_m the number of parameters in our model (6) and n_d the number of data points (2507). Therefore, we can compute the standard deviation σ and the corresponding number of σ the test-script χ^2 is from the mean as,

$$\sqrt{2n} = \sqrt{2(2507-6)} \approx 70.725 \implies \text{for } \; n = 2501 \implies \sigma \text{ from mean} \approx \frac{15268 - 2501}{70.725} \approx 180.52\sigma. \tag{1}$$

Therefore, given this 180.52σ deviation from the mean (much larger than the commonly used 5σ rule-out), the initial parameters are a very poor fit. For parameters equal to [69,0.022,0.12,0.06,2.1e-9,0.95], χ^2 is 3272.2033778089576, which implies a deviation from the mean of,

$$\frac{3272.203377808957 - 2501}{70.725} \approx 10.90\sigma \tag{2}$$

which is still quite large, and therefore not an acceptable fit either.

Q2) Use Newton's method or LM to find the best-fit parameters, using numerical derivatives. Your code should report your best-fit parameters and their errors `planck_fit_params.txt`. Please write your own fitter/numerical-derivative-taker rather than stealing one. Keep track of curvature matrix at the best-fit values for the next problem. What are best-fit parameters with the DM density set to zero - how does this χ^2 compare to the standard value? If you get this to work, print the parameters/errors in `planck_fit_params_nodm.txt`.

We will write a LM non-linear least squares method using the double-sided derivative approximation to compute the numerical derivatives w.r.t. each of the model parameters. These derivatives will take the form,

$$\frac{\partial f}{\partial x_i} = \frac{f(x_i + \delta_i) - f(x_i - \delta_i)}{2\delta_i} \tag{3}$$

for x_i the parameter the derivative is taken with respect to, and δ_i the corresponding displacement over which the derivative is computed. We will take the δ values to be approximately one hundredth of the parameter value to ensure our step sizes are neither too large nor too small. Using our LM function, the best-fit parameters were found after 5 iterations for a χ^2-threshold of 0.01, at which point the convergence criteria of $\lambda = 0$ and $|\Delta\chi^2| < 0.01$ were met. The best fit parameters are reported in `planck_fit_params.txt`. The parameter uncertainties are also included, and are calculated from the parameter covariance matrix which is taken to be the inverse of the left-hand-side of the LM equation,

$$Cov = (J^T N^{-1} J + \lambda \text{diag}((J^T N^{-1} J))^{-1} \tag{4}$$

for J and N being the Jacobian matrix and noise matrix respectively.

The χ^2 evolution with step number is displayed below, which as expected, decreased significantly from its initial value before stabilizing around 2600, which is indicative of a decent fit (within 2σ) given the mean of χ^2 is 2501 (and $\sigma \approx 70$).

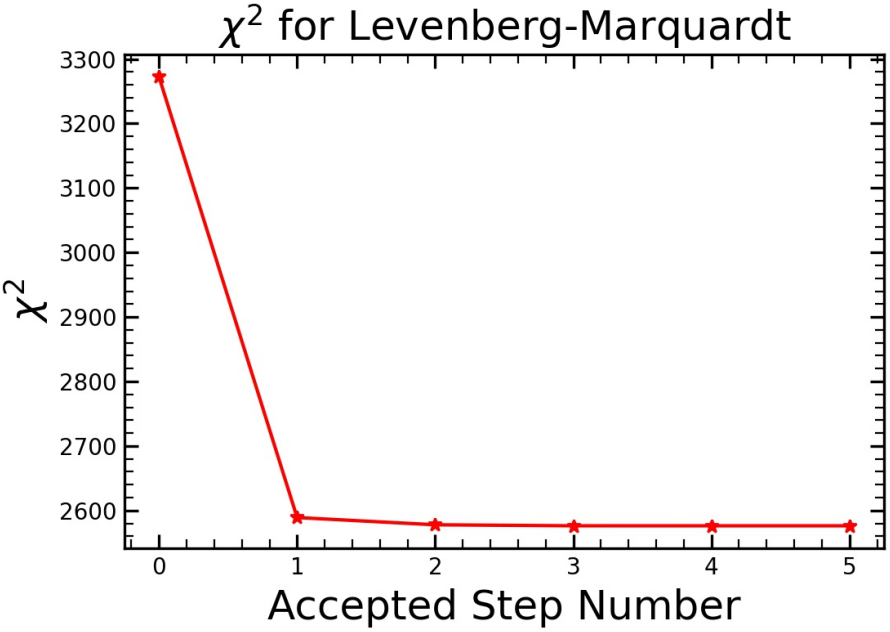
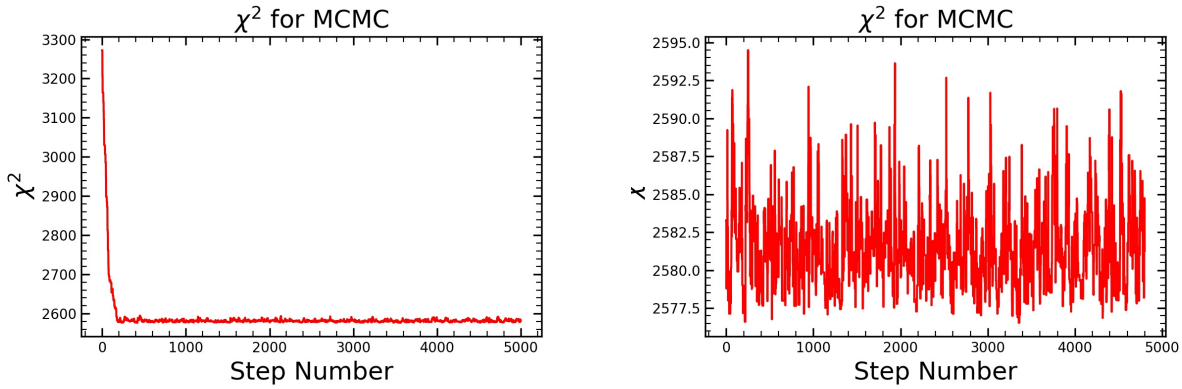


Figure 1: χ^2 evolution with step number for Levenberg-Marquardt.

Q3) Write an MCMC sampler to get parameter values and uncertainties. Draw trial steps from the curvature matrix from Q2. Save your chain, including the corresponding χ^2 values in `planck_chain.txt`. Explain why you think your chains have converged. What is your estimate on the mean value of dark energy Ω_Λ and its uncertainty (given $\Omega_b+\Omega_C+\Omega_\Lambda=1$).

We can use the covariance matrix for the best fit parameters in Q2 to draw trial steps for our MCMC. This is done by Cholesky decomposition of the covariance matrix and then left multiplying this matrix by a column matrix of m-rows (for m parameters in our model) of randomly sampled numbers from a normal distribution. Therefore, a given MCMC step will be the product of this matrix multiplication times a scale factor s, which I have set to be 0.85 to ensure $\approx 25\%$ acceptance rate of steps. This s was determined by running the MCMC for a relatively small number of iterations (100) for different s-values and computing the acceptance rate of steps. After running the MCMC for 5000 steps, the chains (including χ^2 for each step) were recorded in `planck_chain.txt`. The χ^2 evolution with step number is displayed below in Figure 2 and the corresponding parameter chains are presented in Figure 3.

To verify chain the has converged, we can look at the χ^2 evolution with time (steps). If χ^2 has stabilized by the end of the chain, remaining relatively constant over a significant number of steps this can provide some indication that our chain has converged. Further, we can also look at the Fourier transform of the chains and plot the result, which will display the power spectrum of the Markov chain for each parameter (Figure 4). If the power spectrum for a given parameter flattens off in the left half of the spectrum, this is indicative of convergence.



(a) χ^2 evolution over 5000 steps from MCMC.

(b) χ^2 from MCMC with 200 steps removed as burn-in.

Figure 2: χ^2 evolution with and without burn-in. χ^2 quickly stabilizes around 2580 after the first couple hundred steps.

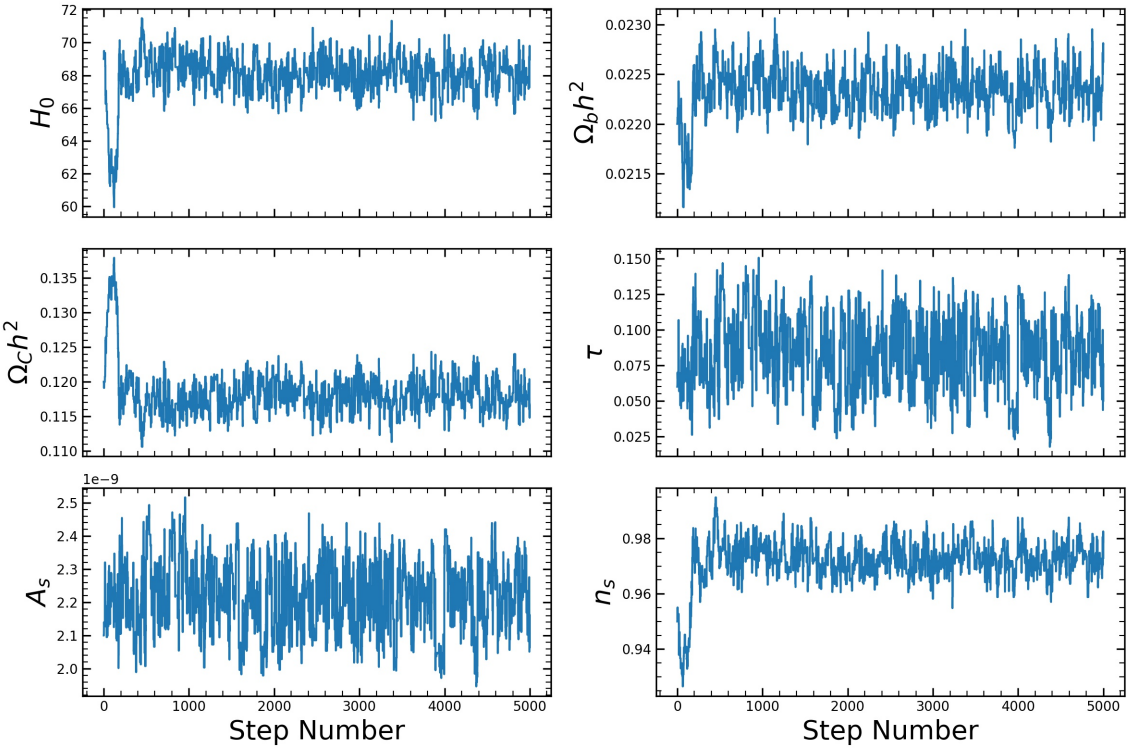


Figure 3: Parameter chains from MCMC, without burn-in. Because burn-in was not removed, there is still some memory of the initial input parameters on the left hand side of each plot, although after a few hundred steps, the chains exhibit the expected white noise behaviour as they explore each parameter space.

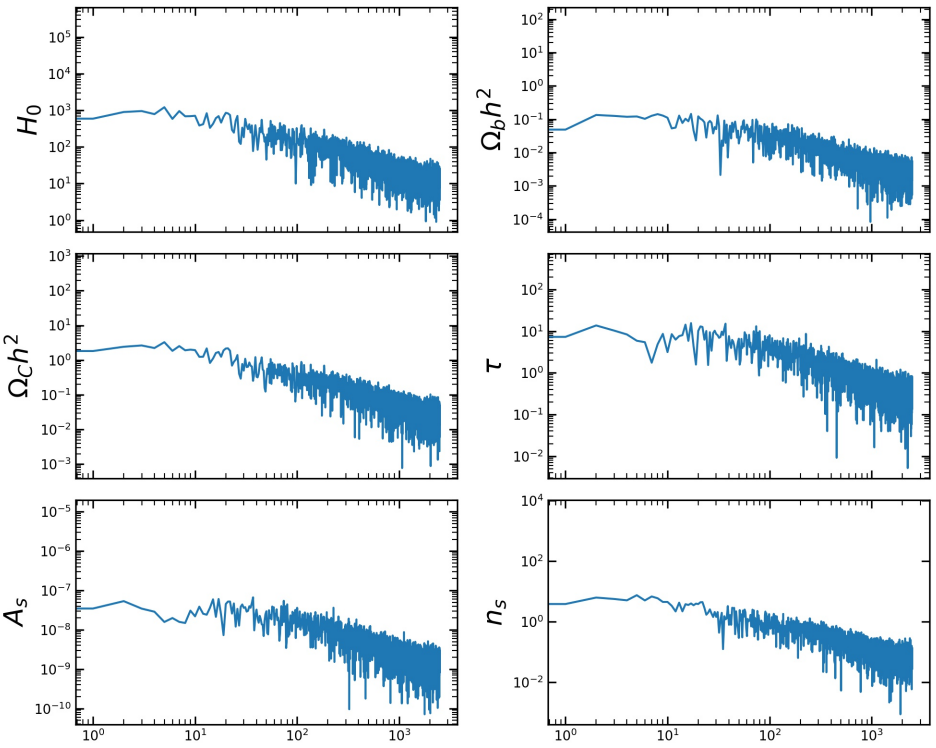


Figure 4: Power spectra for each parameter (without the burn-in steps removed). There is a slight flattening on the left-hand side of each plot, followed by the characteristic 'knee' in the curve. It is expected that chains with $> 10^4$ steps would likely produce a more noticeable flattening and 'knee'.

Therefore, given we have plotted the power spectra (Figure 4) of the chains and observe a flattening of the power on the left side of the plot for each parameter, we can consider the chains to have converged. It is likely that with more iterations (i.e. $> 10^4$) we would observe a more extreme flattening, and a more noticeable 'knee' in the power spectrum, however due to computational time limitations this is sufficient to tentatively say that the chains have converged.

Now, to estimate the mean value of the dark energy, we can consider the formula,

$$\Omega_b + \Omega_C + \Omega_\Lambda = 1 \implies \Omega_\Lambda = 1 - \Omega_b - \Omega_C \tag{5}$$

where we have solved for $\Omega_b h^2$ and $\Omega_C h^2$, for $h = H_0/100$, where we have also estimated H_0. Therefore, we can estimate Ω_Λ as,

$$\Omega_\Lambda = 1 - \frac{100^2 \Omega_b h^2}{H_0^2} - \frac{100^2 \Omega_C h^2}{H_0^2} \approx 0.697920 \tag{6}$$

and its uncertainty as,

$$\delta\Omega_\Lambda = \sqrt{\left(\frac{\partial \Omega_\Lambda}{\partial \Omega_b h^2}\delta\Omega_b h^2\right)^2 + \left(\frac{\partial \Omega_\Lambda}{\partial \Omega_C h^2}\delta\Omega_C h^2\right)^2 + \left(\frac{\partial \Omega_\Lambda}{\partial H_0}\delta H_0\right)^2}. \tag{7}$$

Evaluating these derivatives analytically we can plug them back into the above formula for $\delta\Omega_\Lambda$. Taking the parameter errors as the standard deviation of the chains after omitting the first 200 steps as burn-in we get,

$$\delta\Omega_\Lambda = 10^4 \sqrt{\left(\frac{-2.06 \times 10^{-4}}{68.14^2}\right)^2 + \left(\frac{-0.00227}{68.14^2}\right)^2 + \left(\frac{2}{68.14^3}\left(0.022355 + 0.11791\right) 1.01\right)^2} \approx 0.010207. \tag{8}$$

Therefore, our estimate on Ω_Λ is 0.698 ± 0.010.

Q4) Polarization data give a much better constraint on reionization, with $\tau = 0.0540 \pm 0.0074$. Run a new chain where you include this constraint (saved to `planck_chain_tau_prior.txt`) and compare those results to what you get from importance sampling you chain from Problem 3. Re-estimate the parameter covariance matrix (via importance sampling) before running the new chain.

Importance sampling our chain from Q3 simply amounts to weighting by the prior on τ, which we can construct to be of Gaussian form with mean 0.0540 and standard deviation 0.0074,

$$\pi(\tau) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\tau-\mu)}{2\sigma^2}} = \frac{1}{\sqrt{2\pi(0.0074)^2}} e^{-\frac{(\tau-0.0540)}{2(0.0074)^2}}. \tag{9}$$

Therefore, we can weight the parameter chains from Q3, such that iterations with τ closer to the constrained value are more significantly considered in the calculation of the mean. This is done by way of simple weighted arithmetic mean,

$$\bar{x}_i = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i} \tag{10}$$

for x_i the parameter of interest, and w_i the Gaussian weights computed from a particular iteration's τ-value. The standard error of the weighted mean may then be calculated as,

$$\bar{\sigma}_{\bar{x}_i} = \sqrt{\frac{\sum_{i=1}^{n} w_i (x_i - \bar{x}_i)^2}{\sum_{i=1}^{n} w_i}}. \tag{11}$$

Applying this procedure to each parameter chain from Q3, and discarding the first 200 steps as burn-in, we get weighted means and standard errors of,

$$\begin{aligned} &H_0 = 67.658 \pm 0.910 \\ &\Omega_b h^2 = 0.022280 \pm 0.000178 \\ &\Omega_c h^2 = 0.11889 \pm 0.00214 \\ &\tau = 0.05515 \pm 0.00744 \\ &A_s = 2.0962 \times 10^{-9} \pm 3.21 \times 10^{-11} \\ &n_s = 0.96991 \pm 0.00511 \end{aligned} \tag{12}$$

which are not substantially different from those obtained via the MCMC in Q3, except for the τ value, which as expected, is now much closer to the value from polarization data.

Running a new chain with the constraint on τ included may be done by sampling a value of τ from a Gaussian distribution (of mean = 0.054, standard deviation = 0.0074) at each step. As suggested, the covariance matrix may be recomputed as well, this time from the resulting parameter chains in Q3 after importance sampling. To importance sample the chain from Q3, we need to compute the weighted means for each parameter in the chain using the Gaussian prior on τ (which is done above, ignoring the first 200 steps in chains). Then the new covariance matrix can be written as,

$$Cov = \bar{w}_i (x_i - \bar{x}_i)(x_i - \bar{x}_i)^T \tag{13}$$

where $\bar{w}_i$ are the normalized Gaussian weights from the prior on τ, and $\bar{x}_i$ are the weighted arithmetic means of the each of the parameters. The result of this matrix multiplication yields a 6-by-6 importance sampled parameter covariance matrix.

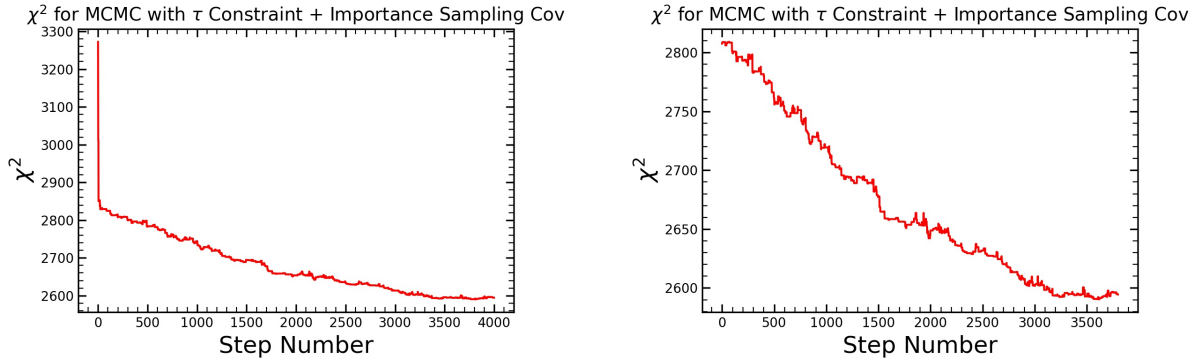
Therefore, passing this as the covariance matrix to our MCMC algorithm, the steps for all other parameters other than τ (after Cholesky decomposition and multiplication by a random column matrix), may be

obtained. The mean and standard deviation of these new chains are reported below for each parameter after discarding the first 200 steps as burn-in;

$$\begin{aligned} H_0 &= 68.845 \pm 0.657 \\ \Omega_b h^2 &= 0.022213 \pm 0.000153 \\ \Omega_c h^2 &= 0.11780 \pm 0.00155 \\ \tau &= 0.052181 \pm 0.00378 \\ A_s &= 2.0994 \times 10^{-9} \pm 1.65 \times 10^{-11} \\ n_s &= 0.95885 \pm 0.00460. \end{aligned} \tag{14}$$

The χ^2 evolution for the importance sampled covariance matrix MCMC is presented in Figure 5, the parameter chains in Figure 6, and the power spectra in Figure 7.

Comparing these results to those from importance sampling the chain from Q3, they are roughly similar, for the most part within 1σ of their respective parameters. It may be noted however that the magnitude of the 1σ errors from the re-run MCMC in Q4 are smaller than those from the importance sampled chain from Q3, with some errors reduced by up to a factor of 2. For example, the error on τ for the re-run MCMC is roughly one half the size of that from the importance sampled chain in Q3.



(a) χ^2 evolution over 4000 steps from MCMC.

(b) χ^2 from MCMC with 200 steps removed as burn-in.

Figure 5: χ^2 evolution with and without burn-in for an importance sampled covariance matrix. χ^2 evolves differently in this chain, steadily declining for a few thousand iterations, before stabilizing around $\chi^2 = 2590$.

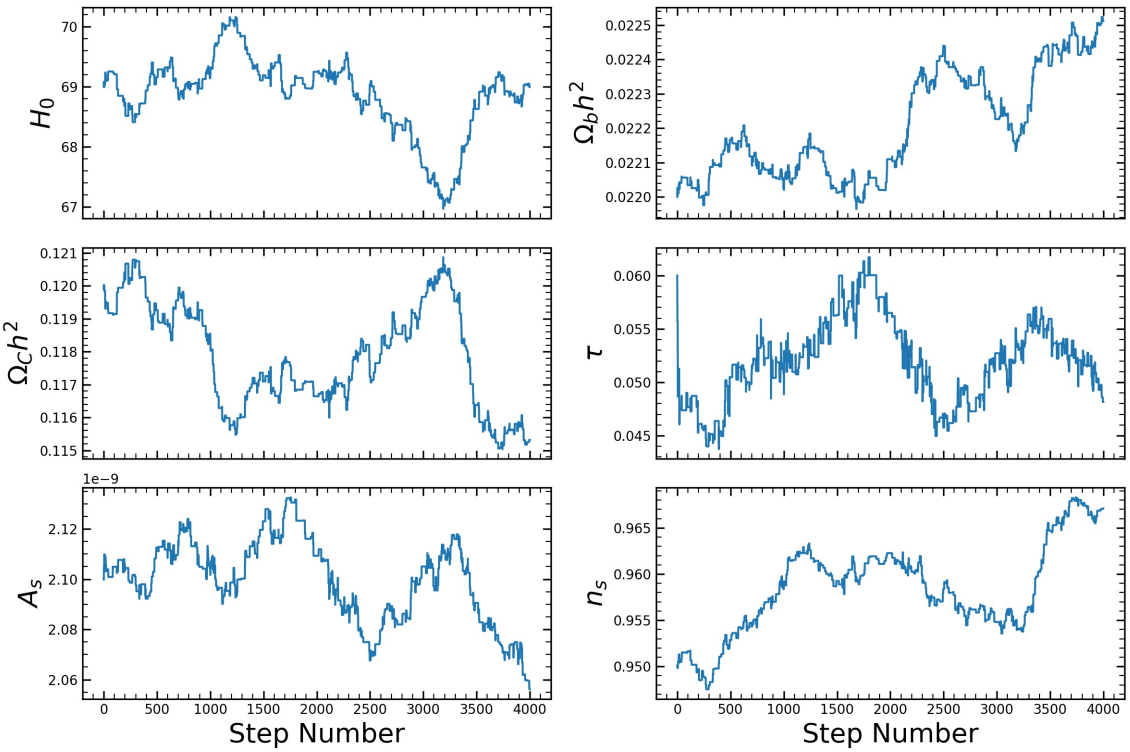


Figure 6: Parameter chains from the MCMC, without burn-in. The chains do not appear to exhibit the same white noise as those from Q3, although it may be noted that the scaling of the y-axis is substantially smaller than that from Figure 3. Further, the acceptance rate was much smaller (around 10%) compared to the MCMC in Q3, which results in less movement around the respective parameter spaces, and hence more defined structure in the parameter chains.

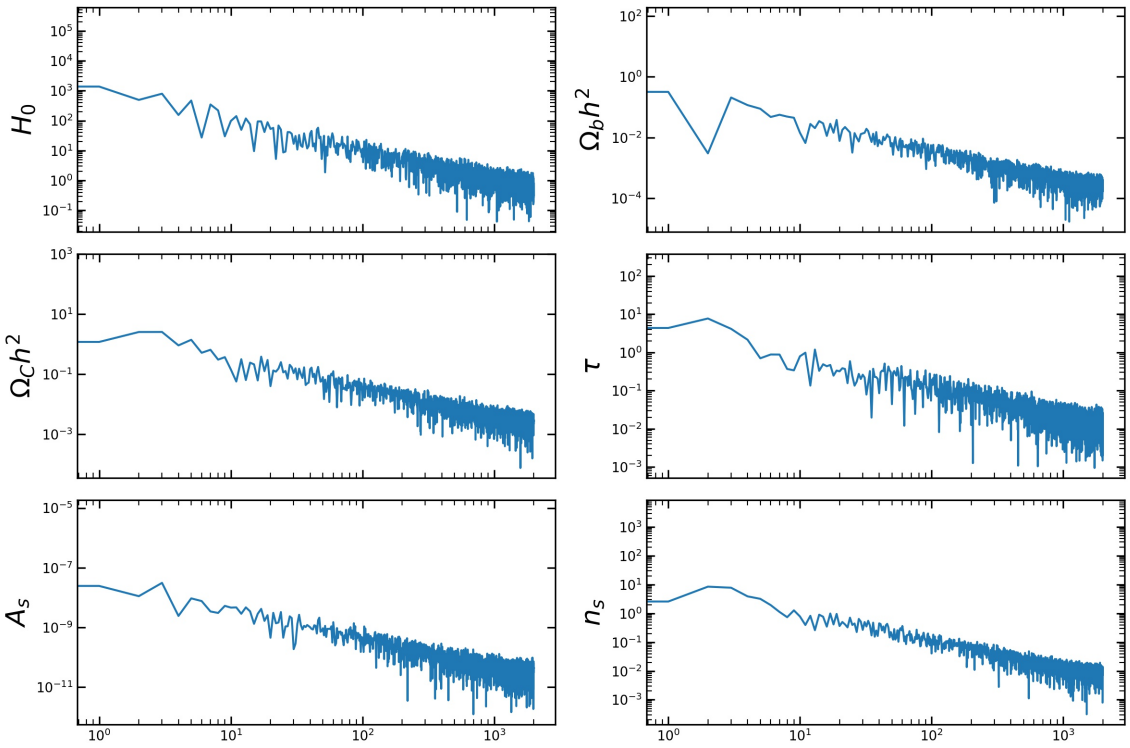


Figure 7: Power spectra for each parameter (without the burn-in steps removed). There is a slight flattening on the left-hand side of each plot for most of the parameters, although much less of a characteristic knee in the curves compare to Figure 4. It is expected that chains with $> 10^4$ steps would likely produce a more noticeable flattening and 'knee'.