Homework 3

Jatin Abacousnac

October 25, 2019

Problem 1

Problem 7.2 (Newman)

The periodicity of sunspots is a well-known phenomenon, and is studied in this problem. A plot of the sunspot number (a measure of the number of sunspots) is plotted against time. From the figure, an estimate for the periodicity of sunspots, T, can be obtained:

$$T = \frac{\text{total time}}{\text{number of cycles}} \approx \frac{3142}{24}\,\text{months} = 131\,\text{months}. \tag{1}$$

This agrees with the "common knowledge" that the sun's activity has an 11-year cycle. In this problem, a more rigorous examination of the sunspot data, carried out in Fourier space, will enable a calculation for T.

Recall the Discrete Fourier Transform (DFT) that converts variables into the Fourier ('k') space. The Fourier coefficients are given as

$$C_k = \sum_{k=0}^{N-1} Y_n e^{\frac{-2\pi i k n}{N}}. \tag{2}$$

The power spectrum of a signal can be obtained by calculating the fourier coefficients, given by Equation 2. This was implemented using Python[1], where a DFT is coded up within a

[1] Yes, I did not use MATLAB this time!

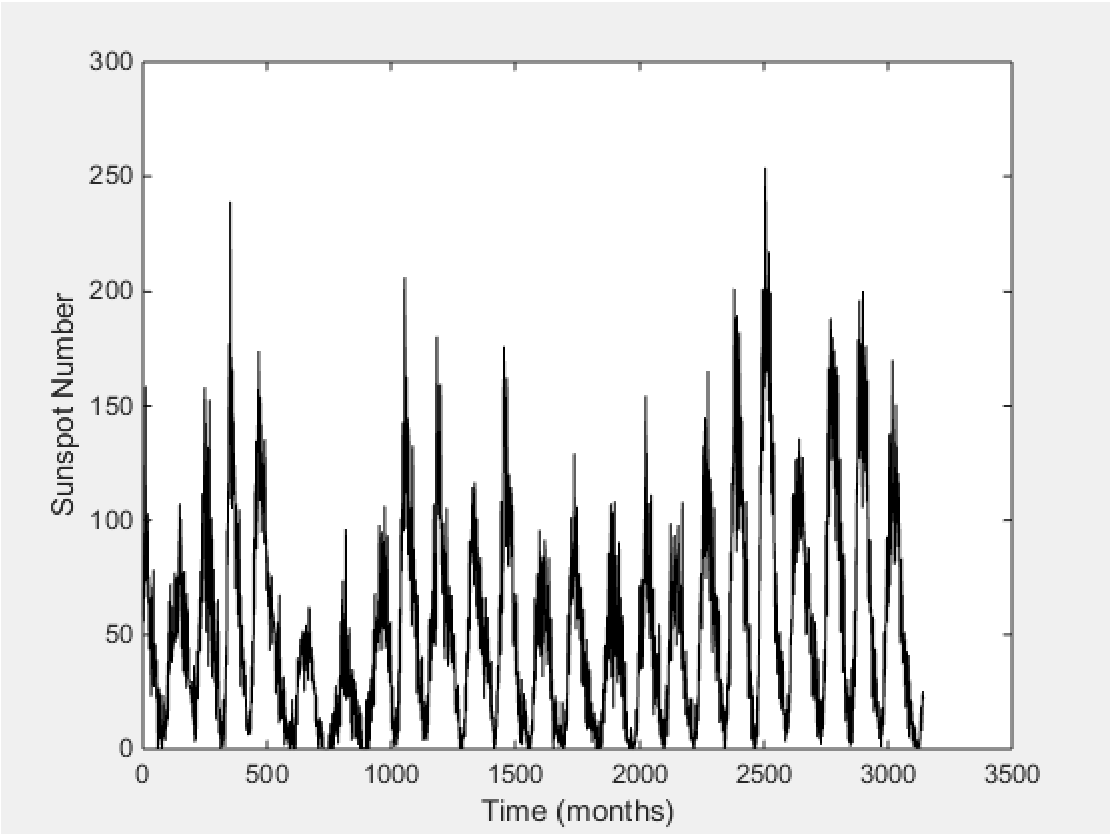

Figure 1: The periodicity of sunspots

nested for loop that finds for values of k, the corresponding value of $|C_k|^2$. The end result is a plot, known as the power spectrum. From the plot, we notice that at $k = 25$, a peak appears. This corresponds to a wavenumber of 25, which indicates that over the entire time, 25 cycles appears. And hence, the period is found to be

$$T = \frac{3142}{25}\,\text{months} = 126\,\text{months}, \tag{3}$$

which is similar to the earlier estimate.

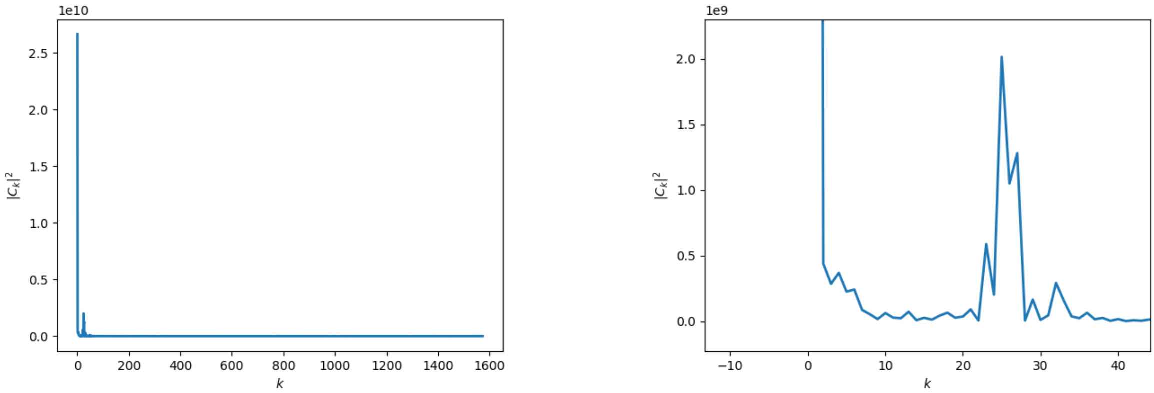

Figure 2: A plot of the Fourier coefficients^2 against k, with a zoomed in portion of the plot on the right hand side. Note the presence of a peak at a non-zero value of k.

Problem 2

Problem 7.9 (Newman)

An image may be blurred according to some function known as the point spread function. In 2D, this function can be denoted as $f(x, y)$. The blurred image is then a convolution of the brightness function $a(x, y)$ and this point spread function:

$$b(x, y) = a(x, y) * f(x, y) = \int_0^W \int_0^H a(x', y')f(x - x', y - y')dx'dy', \tag{4}$$

where the limits W and H are the width and height of the image. Naturally, to obtain the original image, a deconvolution is in order. Then, the inverse fourier transform of the function that is obtained gives the original image.
In this problem, we are provided with a blurred image in the form of values of the image brightness at a point (x, y). If the brightness values are plotted, the blurred image may be displayed. The point spread function used to blur the image (presumably that of a house) is a gaussian function of form:

$$f(x) = e^{-(\frac{x^2+y^2}{2\sigma^2})}, \tag{5}$$

where σ is set to be 25. Often, the psf may be the result of some phenomenon that affects measurements, and thus a deconvolution helps make the data sharper.

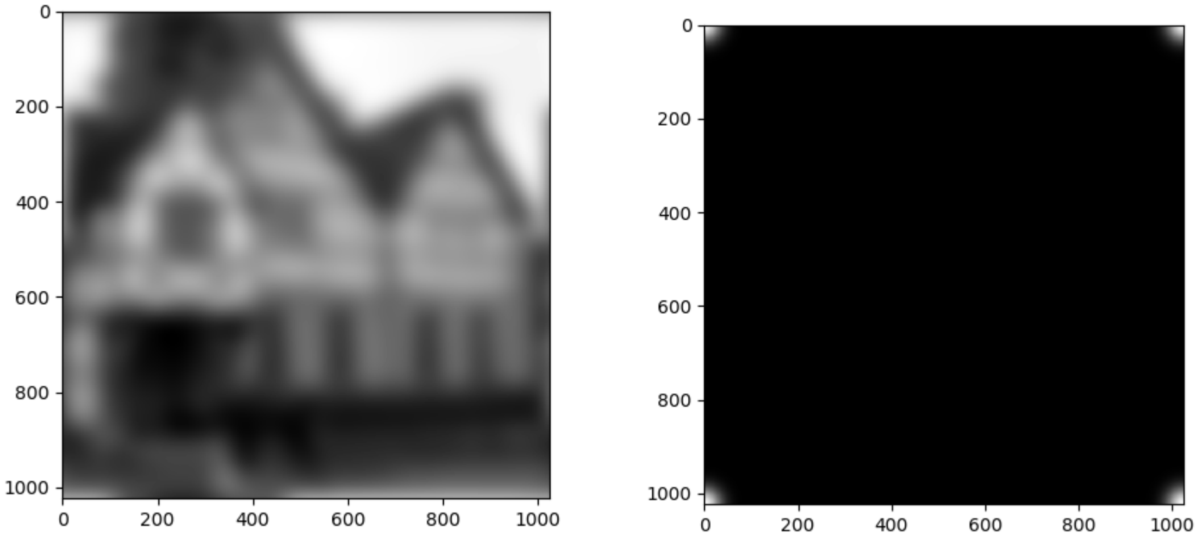

Figure 3: The left image shows the blurred picture that was generated using brightness values for every (x, y), and on the right hand side is the Gaussian point spread function that was used to blur the image.

The algorithms is as follows:

- Obtain an array for the convolved image (i.e. blurred image).
- Obtain an array for the point spread function. Note that both arrays have the same dimensions.
- Perform a deconvolution.
- Inverse Transform the deconvolved array and plot in physical space.

The deconvolution process of fourier transformed functions A=B/F. The fourier transformed functions A, F and B are the desired unblurred image array, the gaussian function and the convolved (blurred) array, respectively, all in fourier space. Finally (in 1D),

$$b(x) = \mathcal{F}^{-1}(B(k)) = \int_{-\infty}^{\infty} B(k) e^{2\pi i k x} dk. \tag{6}$$

When this algorithm is applied to the blurred image obtained previously, the image is unblurred, as shown below[2].

[2]Image deconvolution works! Is that someone's house? Oh wow, those (previously blurred individuals) must be the owners!

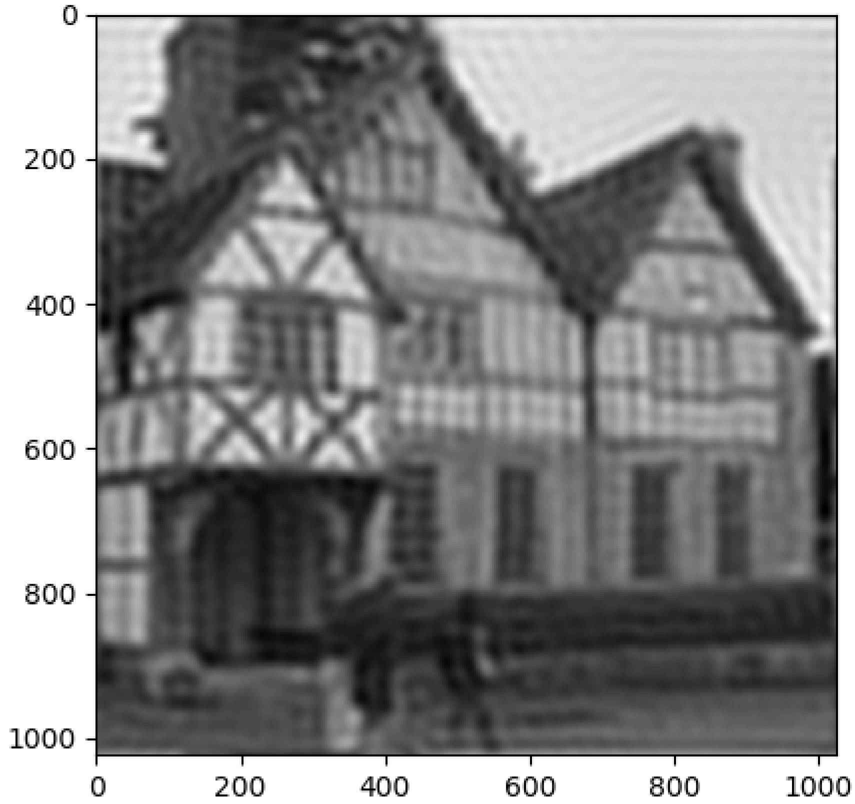

Figure 4: The image is unblurred, and several details previously blurred by the psf are now visible.

Why still blurry?

The final image is not perfectly sharp, but it is a drastic improvement upon the original one that was provided. The fourier transform of the point spread function has several zeros, which means that when performing the deconvolution (matrix division), we are dividing by zeros sometimes. A workaround is to define an ϵ value such that when the gaussian value is lower than ϵ, then we do not divide. Since the fourier transform of any function has large spikes at zero, it is likely for us to obtain zeros frequently in fourier space, and thus we are not necessarily deconvolving several times. Thus, the image is not perfectly sharp. For this purpose, I included a counter in my code to see how many times this condition (psf$(x, y) < \epsilon$), and found that it happened 523252 times, which is almost half of the times! Note that for a value of $\epsilon = 10^{-}5$, it occured 522640 times, showing that the algorithm is not sensitive to the value of ϵ. Another major reason is that the Gaussian function with $\sigma = 25$ is a guess for what the psf is. σ could in actuality be different, or the psf could be some other function or complicated combination of functions entirely.