13.1 Frequency domain analysis and FFT

Frequency domain analysis, in particular the so-called Fast Fourier Transforms (FFT), is a standard computational tool in many branches of science. The original approach was developed by Gauss and over a number of years it has been used as a very efficient method of improving the computational economy. We will open this chapter by considering a few examples to illustrate how to use spectral analysis and how to interpret the results. Once we’ve done this, we’ll dive deeper into how to do it.

Fig. 13.1.1 shows a typical seasonal time series graph:

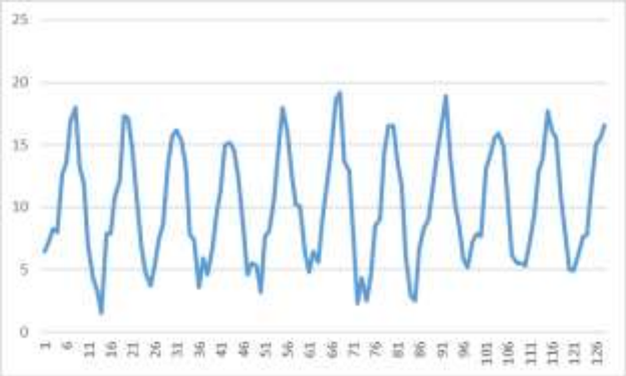


Fig. 13.1.1 Monthly mean temperatures in Central England Jan 90 –Aug 04

The above series is presented in the time domain, which means that every observation is recorded chronologically on the ‘x’ axis. If the series, for example, constitutes monthly data, then an observation for one month represents 1/12 of the year, although not necessarily a proportional part i.e. not all the values for all the months are the same. If we translate the series from a time domain into a frequency domain, the same series will look like Fig. 13.1.2.

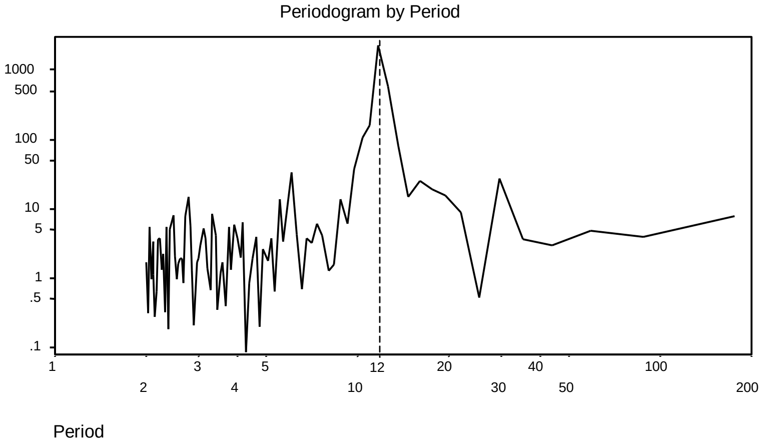


Fig. 13.1.2 Periodogram by period for Monthly mean temperatures in Central England

The ‘x’ axis consists of periods. Clearly the periods in Fig. 13.1.2 are not equally spaced. The scale is chosen to be logarithmic (as is the y-axis), otherwise it would be impossible to squeeze everything on one page. We can take one more step and change periods into frequencies (as in Fig. 13.1.3), which is yet another way of representing the same series.

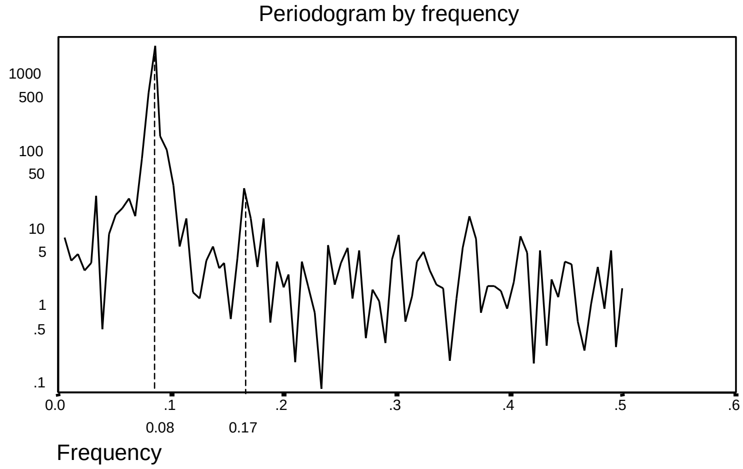


Fig. 13.1.3 Periodogram by frequency for Monthly mean temperatures in Central England

Let us explain the difference between these two concepts and what exactly do we see in these two graphs.

The word cycle usually implies that something will go up and down and will repeat itself after a period of time. If our time series of length ‘n’ consists of the shortest possible cycle (one up, one down observation interchangeably), then the maximum number of cycles for this frequency is n/2. This means that the number of cycles can reach maximum for 50% of the number of observations. If, on the other hand, the whole series consists of just one cycle, then the frequency is only 1/n. The lowest frequency possible has zero cycles, which means there is no cycle at all. Frequencies are, therefore, calculated as the number of times the cycle repeats itself in a series over the total number of observations.

An alternative way of looking at this is to say that if we have monthly data, then one cycle constitutes of 12 months. The frequency of this series is 1/12 cycles per year. This effectively tells us what we have already stated, that the frequencies and periods are reciprocals of one another. Because the peak in the graph in Fig. 12.1.2 shows number 12, this indicates that the periodicity of the series is 12 months. On the other hand, the graph in Fig.12.1.3 shows the highest frequency to be 0.08 (which is if we divide 1/12), confirming that these two graphs show one and the same thing, just from a different point of view. By the way, looking at other frequencies on the graph in Fig. 13.1.2, we can see that the second highest frequency is period six, which is equivalent to 0.17 (i.e. 1/6) in Fig. 13.1.3).

So, if frequencies are reciprocals to periods, then the above periodogram as in Fig. 13.1.2 can be presented as frequency periodogram (Fig. 13.1.3). The graph in Fig. 13.1.3 is also called the power spectrum, sometimes referred to as S(f), or as P(ω) if expressed as a function of the angular frequency (more about this further down). The power spectrum is the sum of the squares of individual sine and cosine terms that characterises the series at every frequency, and we will explain this shortly.

At each frequency, a time series can be presented as a weighted sum of a sine and cosine component having that frequency. In other words, if we square and add up all the sine and cosine weighting at a particular frequency, we will get the original series. This is a very important point, emphasising that spectral analysis does not change or skew data. It merely looks at the same data from a different, a frequency point of view. By having the spectral analysis data, we can actually reconstruct the original series without knowing how this series looked in the first place, as we also will show later.

This also implies that different time series will have different characteristic periodograms, which can be perceived as signature periodograms. A white noise process is typically characterised by broad-band power spectra. Without getting into details, a general rule is, the rougher the series, the more high frequencies it has. The smoother the series, the more low frequencies its spectrogram will have. As we already said, if a series is very ‘jumpy’, then it will have close to n/2 cycles. If for example, on the other hand, the series has n/2 cycles, we need to know that additionally it will have a constant with frequency zero. As every cycle is either a sine or cosine wave, the series could, in this case, be described by using (n/2)+1 coefficients. Generally, the number of cycles, plus the zero frequency is the number of terms that will define every series.

As the spikes in periodograms are sometimes difficult to position, or could be misleading due leakage and aliasing (engineering terms that go beyond the scope of this book), various people have applied a number of smoothing transformations to make the data more presentable. These transformations are called the ‘windows’ and they represent spectral density estimates. The most famous windows, used in many software packages are: the Tukey-Hamming window, the Parzen window and the Bartlett window. Fundamentally, they all take the actual observations and/or multiply them with some constants and/or other function values to make the peaks easier to interpret and “decipher” visually. The graphs showing such transformed frequencies are often called spectrograms, power spectra or spectral density estimates. Fig. 13.1.4 is an example of the Tukey-Hamming window showing spectral density estimates for the above series:

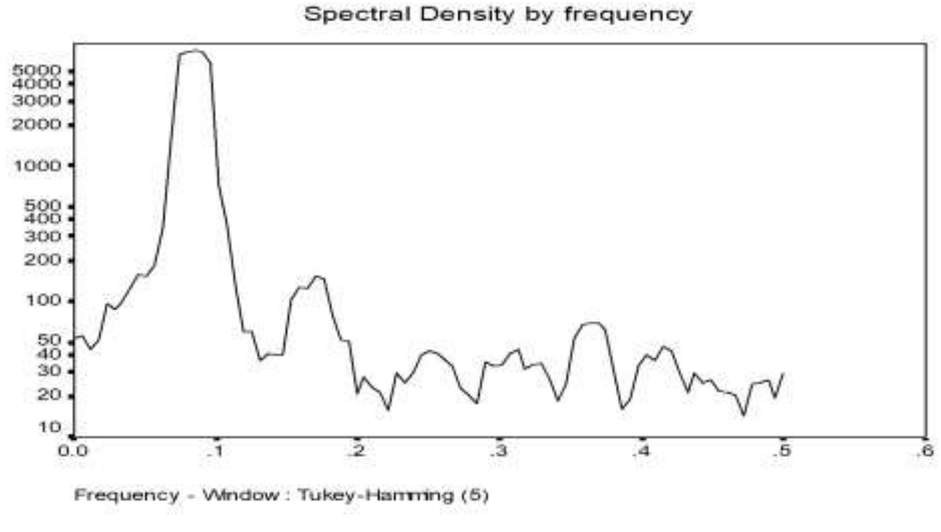


Fig. 13.1.4 Spectral density by frequency for Monthly mean temperatures in Central England

To relate the power spectrum to some of the terminology from previously described methods, we need to state that the **power spectrum** or **power spectral density**, as it is often called, is the inverse Fourier transform of the **autocovariance** or **autocorrelation** function (remember that autocorrelation is just normalized autocovariance function). Effectively, the power spectrum and the autocovariance are Fourier transforms of each other. This makes these two concepts interchangeable and it is, almost, a matter of preference in which space to operate and conduct an analysis. Some scientists talk in terms of power spectra or spectral density, whilst others use the terms autocovariance or autocorrelations. They are talking of the same thing, and it is their background that makes them more comfortable to think in one paradigm or the other.

We will now dive into some of the details of this approach to analysis and illustrate how to use it on a practical level. We'll start with a modest amount of theory.

You will recall from earlier chapters that the most common way of approximating a function is to use a standard polynomial equation. We have used, for example, a formula $y=a+bx+cx^2$ to describe parabola, which we tried to fit to a data set, i.e. a time series. However, some data sets defy all the rules and it is almost impossible to find a polynomial equation that fits them. Imagine a function pulsating at regular intervals, i.e. showing some periodic movements. Mathematicians would say that if 'T' is the period length, then a general function describing such a series is $f(t)=f(t+T)$. In order to deal with periodic functions our best option is to use some sort of a sinusoidal function. One such formula used to approximate periodic functions is:

$$f(t) = A \sin\theta \qquad (13.1.1)$$

Where, A defines the amplitude, i.e. the height of the oscillation and θ is the angle. Fig. 13.1.5 visualises these concepts.

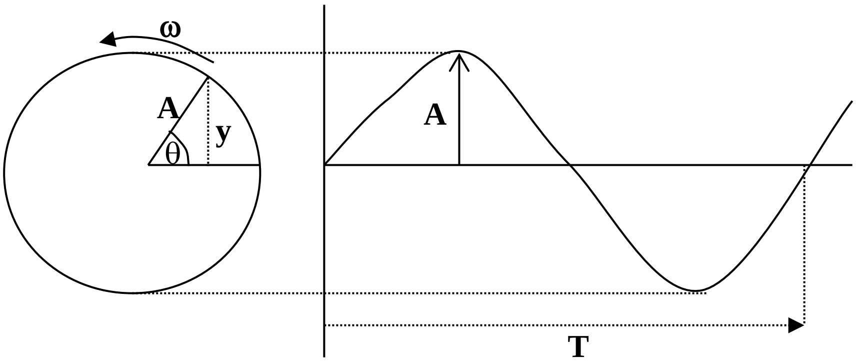


Fig. 13.1.5 Connection between periodicity and circular movement

From the above picture we can see that as the angle θ changes, so does the value of the function $y=f(t)$. The function will start at $A=0$, reach its maximum at $+A$, then it will cross the x axis (again $A=0$), reach its minimum at $-A$ and finally return to $A=0$. This will all happen inside a periodic interval T. As sinusoid is a representation of the circular movement of A, this means that T can be also be interpreted as $T=360^0$, or 2π radians, which is the full periodicity of a sine wave.

From the Fig. 13.1.5 we can also observe that A rotates at some regular (constant) angular velocity ω, which means that in time *t*, the angle θ is:

$$\theta = \omega t \qquad (13.1.2)$$

We can, therefore write equation (13.1.1) as:

$$f(t) = A \sin \omega t \qquad (13.1.3)$$

We have said that a periodic function repeats itself after a time 'T'. This is applicable if we present the function in the so-called time space. However, the same function can be presented in a frequency space. To do so, we need to define the frequency as:

$$f = \frac{1}{T} \tag{13.1.4}$$

If we do this, the complex relationship between the time and the frequency space can be visualised in a three-dimensional graph as in Fig. 13.1.6.

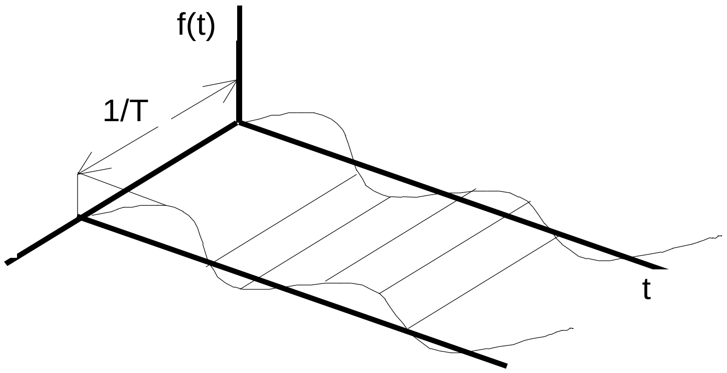


Fig. 13.1.6 Connection between the time and frequency space

A time needed to complete one full cycle, at an angular velocity of ω is $360^0/\omega$, or $2\pi/\omega$. In this case a periodic time T is:

$$T = \frac{2\pi}{\omega} \tag{13.1.5}$$

Combining (13.1.4) and (13.1.5) we get:

$$f = \frac{\omega}{2\pi} \tag{13.1.6}$$

This means that $\omega=2\pi f$, which also implies that (13.1.3) can be written as:

$$f(t) = A \sin 2\pi f t \tag{13.1.7}$$

The above explanations depended on two assumptions, the first one was that we start with A=0 and the second one that we express circular movements as a sinusoid. Neither of the two assumptions are really necessary. We could have used cosine function to achieve exactly the same effect, or we could have started not from $\theta=0$, but from some other angle ϕ. If this was the case, ϕ is the phase angle.

This also implies that the phase space can be presented in two forms, one where the frequency depends on amplitude and the other where it is dependent on the phase shift (see Figs. 13.1.7-8). These two plots represent the frequency spectrum.

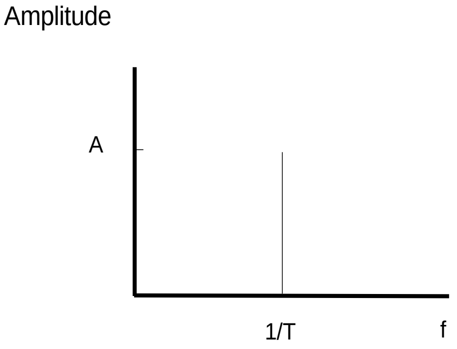


Fig. 13.1.7

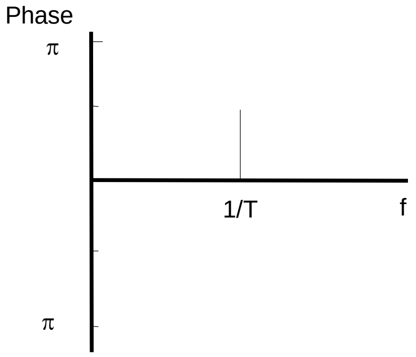


Fig. 13.1.8

The phase shift defines by how much the sinusoid is shifted horizontally. If the phase shift is positive, then it is said to be a leading phase angle (or shift). If it is negative, it is referred to as a lagging phase shift. We decided that the rest of this chapter will not focus on phase spectrum, but on amplitude spectrum, which is more intuitive to non-engineers.

We said that $\omega = 2\pi f$, or $\omega = 2\pi/T$, which is often called the fundamental frequency. The multiples of this fundamental frequency (2ω ,3ω ,4ω, ...) are called harmonics. This is the point we enter the world of Fourier analysis. Fourier ingeniously discovered that infinite series of harmonically related frequencies can fit virtually any periodic function. This is called a Fourier series. If we take a look at a discrete version of this equation, it is written concisely as:

$$f(t) = \frac{1}{2}a_0 + \sum_{k=1}^{N-1}(a_k cos 2\pi kt + b_k sin 2\pi kt) \qquad (13.1.8)$$

Or

$$f(n) = \frac{1}{2}a_0 + \sum_{k=1}^{N-1}\left(a_k \cos\,(k\frac{2\pi}{N}n) + b_k sin(k\frac{2\pi}{N}n)\right) \qquad (13.1.9)$$

The two alternative expressions above are called a Fourier series. To calculate the coefficients for the either of the two expressions above, we need to use the observation values x_n:

$$a_0 = \frac{1}{N}\sum_{n=0}^{N-1} x_n \qquad (13.1.10)$$

$$a_k = \frac{2}{N}\sum_{n=0}^{N-1} x_n \cos\left(\frac{2\pi kn}{N}\right) \qquad (13.1.11)$$

$$b_k = \frac{2}{N}\sum_{n=0}^{N-1} x_n \sin\left(\frac{2\pi kn}{N}\right) \qquad (13.1.12)$$

The line spectra of the function is a phase space (see Fig 13.1.8), i.e. a graphical representation of a complicated function f(t*)* such as the one described by equation (13.1.9), but more about that in a minute.

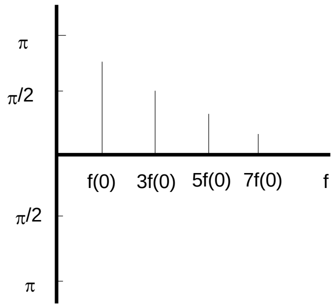


Fig. 13.1.9

However, many functions are quasi periodic, or not periodic at all. Although so far, we only referred to fitting periodic series, the same approach can be used for non-periodic time series. The trick is to assume that a non-periodic wave is just like a periodic one, but the one for which the periodicity approaches infinity. However, we will here concentrate on discrete functions only as majority of time series in business, economics and finance fall in this category.

Before we do this, in order to define Fourier transforms, we will use Euler's formula:

$$e^{ix} = \cos x + i \sin x \qquad (13.1.13)$$

We'll turn this other way around and say that the cosines and sines can also be expressed in terms of complex numbers:

$$\cos\theta = \frac{e^{i\theta}+e^{-i\theta}}{2} \text{ and } \sin\theta = \frac{e^{i\theta}-e^{-i\theta}}{2i} \qquad (13.1.14)$$

In this case, if we substitute all the coefficients a_k and b_k in the equations (13.1.9) and express the cosine and sine terms as complex numbers, then the discrete Fourier transforms (DFT) are defined as:

$$F_k = \sum_{n=0}^{N-1} x_n \, e^{\frac{-i2\pi nk}{N}} \qquad (13.1.15)$$

Where:

F_k = Fourier transform coefficients expressed as complex numbers
x_n = observations (i.e. the function)
k= number of complex coefficients
N = number of observations
n = number of samples n = 0 to N-1 (same as k)
i = imaginary number

It is important to remember that the Fourier transform applies to non-periodic functions, as opposed to the Fourier series, which applies to periodic functions.

So, if a Fourier series is a linear combination of oscillations, then every frequency in it oscillates at an integer multiple of the basic frequency. This is effectively a spectrum of the periodic function and it is often written as S(f). If the function is not periodic, the spectrum does not consist of multiples of the basic frequency but is a continuum. In this case the spectrum is represented by Fourier transforms.

From the equation (13.1.15) it is obvious that the Fourier transform is a complex function (containing 'i' numbers). Effectively, the Fourier transform is an extension of the Fourier series because the basic periodicity is allowed to become infinitely large, which means the function is no longer periodic!

Because we restricted the range of the function given by equation (13.1.15) from n=0 to N-1, this will lead us directly towards Fast Fourier Transforms (FFT). FFT is a well know algorithm used to calculate Fourier transforms. Excel also uses this method to calculate discrete Fourier transforms (DFT or F_k as in the above equation). However, there are two restrictions in Excel, the first one is that the maximum number of observations (or maximum number of Fourier transforms) is 4096 and the number of transforms is restricted to 2^n, which means that we can only calculate the number of transforms that correspond to 4, 8, 16, 32, 64, …, 4096 of them.

OK, let's use a hypothetical discrete time series and calculate the discrete Fourier transforms for this time series. We will use equation (13.1.15) to achieve this.

If we take the exponent to be $\frac{i2\pi kn}{N} = b_n$, then this DFT can be expressed as:

$$F_k = x_0 e^{-b_0 i} + x_1 e^{-b_1 i} + \cdots + x_n e^{-b_{N-1} i} \qquad (13.1.16)$$

As per Euler's formula: $e^{xi} = \cos x + i \sin x$, which means that DFT can be expanded as:

$$F_k = x_0[\cos(-b_0) + i\, sin(-b_0)] + x_1[\cos(-b_1) + i\, sin(-b_1)] + \cdots + x_n[\cos(-b_{N-1}) + i\, sin(-b_{N-1})] \qquad (13.1.17)$$

This implies that the final result of a DFT is a mixture of real and imaginary components. In other words, DFT is a complex number:

$$F_k = A_k + B_k i \qquad (13.1.18)$$

Because F_k is a complex number, it resides in a two-dimensional space, where the x axis is the real part and the y axis is the imaginary part of this complex number. The complex number is the intersection of these two parts (the real and imaginary part).

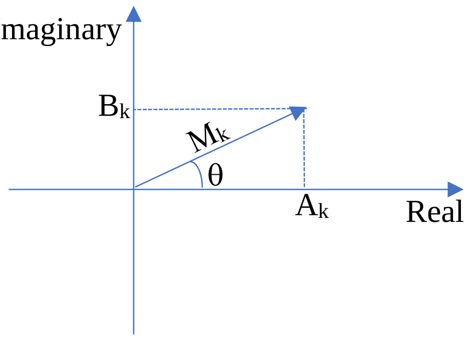


Fig. 13.1.10

To calculate the distance from the origin, we can use the Pythagorean theorem which says that $c = \sqrt{a^2 + b^2}$. In our case it is calculated as:

$$M_k = \sqrt{A_k^2 + B_k^2} \tag{13.1.19}$$

Another word that is used for M_k is that it is a *modulus* of the complex number, or the magnitude. In the picture above we can also see the angle θ. This angle can also be calculated from A_k and B_k as an arc tangent:

$$\theta = tan^{-1}\frac{B_k}{A_k} \tag{13.1.20}$$

If we present this in a time space, and let's assume that our signal is a sinusoid, then these values represent the following:

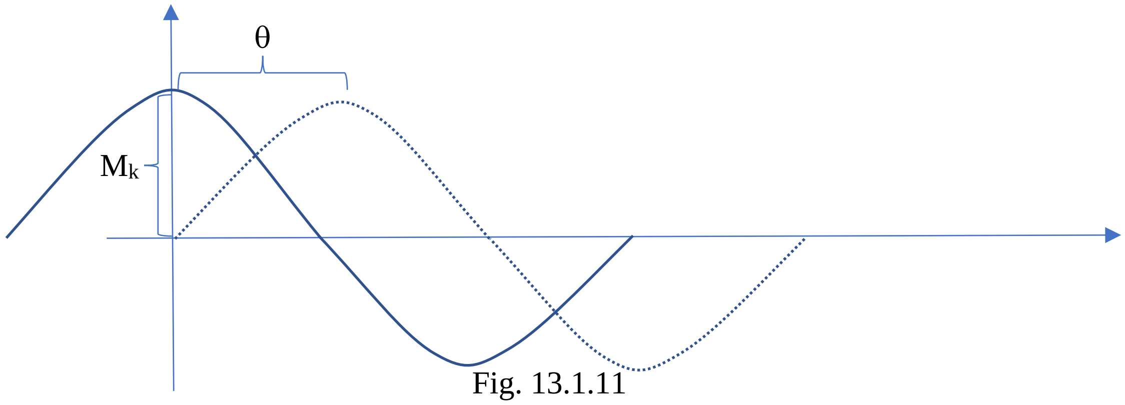


Fig. 13.1.11

We can see that the magnitude M_k (which was calculated as a modulus) is in fact the amplitude of the wave and that θ is the phase shift (which we already know). To calculate discrete Fourier Transforms (F_k) we can use either equation (13.1.15), or the one with the sinus and cosine version (13.1.17) based on Euler transformation. Although (13.1.17) might appear to be a bit easier to handle manually, because we will rely on Excel, equation (13.1.15) is in fact easier to use.

Let's assume that we have the following, very short, time series consisting of only 8 observations:

n	**x_n**
0	20
1	40
2	30
3	50
4	40
5	30
6	50
7	40

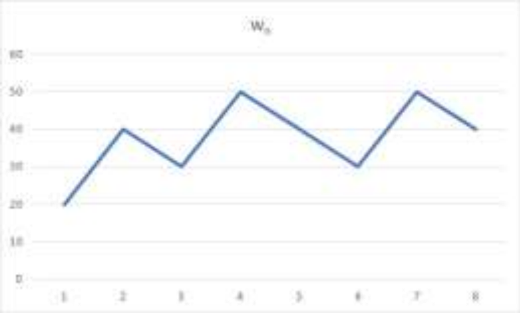


Fig. 13.1.12

For every time observation at the point *n*, we need a frequency value at the point *k*. As we have 8 observations, we will need to calculate 8 DFTs. They will be calculated, using equation (13.1.17) as:

$$F_0 = x_0 e^{\frac{-i \times 2 \times 3.14 \times 0 \times 0}{8}} + x_1 e^{\frac{-i \times 2 \times 3.14 \times 0 \times 1}{8}} + x_2 e^{\frac{-i \times 2 \times 3.14 \times 0 \times 2}{8}} + \cdots + x_7 e^{\frac{-i \times 2 \times 3.14 \times 0 \times 7}{8}}$$

$$F_1 = x_0 e^{\frac{-i \times 2 \times 3.14 \times 1 \times 0}{8}} + x_1 e^{\frac{-i \times 2 \times 3.14 \times 1 \times 1}{8}} + x_2 e^{\frac{-i \times 2 \times 3.14 \times 1 \times 2}{8}} + \cdots + x_7 e^{\frac{-i \times 2 \times 3.14 \times 1 \times 7}{8}}$$

.

.

$$F_7 = x_0 e^{\frac{-i \times 2 \times 3.14 \times 7 \times 0}{8}} + x_1 e^{\frac{-i \times 2 \times 3.14 \times 7 \times 1}{8}} + x_2 e^{\frac{-i \times 2 \times 3.14 \times 7 \times 2}{8}} + \cdots + x_7 e^{\frac{-i \times 2 \times 3.14 \times 7 \times 7}{8}}$$

Because the coefficients that were calculated using equation (13.1.15) are complex numbers, as shown in equation (13.1.18), from there we can calculate the modulus for every coefficient using the equation (13.1.19). This means that for every value of k, we get a series of magnitudes for every complex number:

$$M_0 = \sqrt{A_0^2 + B_0^2},\ M_1 = \sqrt{A_1^2 + B_1^2},\ \ldots,\ M_7 = \sqrt{A_7^2 + B_7^2}$$

And finally, to calculate the amplitude at every frequency, we double the magnitude and average it over the number of observations:

$$A_k = M_k \frac{2}{N} \qquad (13.1.21)$$

The factor $\frac{2}{N}$ that we used to multiply the modulus, provides just the normalization of the coefficients for the length of the series (in fact, the normalization is achieved by dividing M_k by N/2, which is the same as multiplying by 2/N). The next step is to calculate the power spectrum.

Let’s use a short example and take only 8 amplitudes and present them as a bar chart:

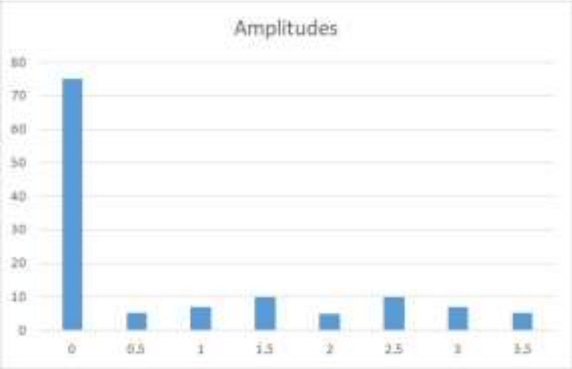


Fig. 13.1.13

To finish the job, we need to calculate the power coefficients for the power spectrum, which means that:

$$P_k = A_k^2 \qquad (13.1.22)$$

The only exception is that the first coefficient P_1 for the 0 frequency has to be divided by 2. In summary, for our artificially short example, we have:

k	**F_k**	**A_k**	**P_k**
0	300	75	2812. 5
1	-20 +5.85i	5.21	27.145
2	-20 +20i	7.07	50
3	-20 -34.14i	9.89	97.855
4	-20	5	25
5	-20 +34.14i	9.89	
6	-20 -20i	7.07	
7	-20 -5.85i	5.21	

A series of these power coefficients P_k represents the power spectrum, also called spectral density function.

These power spectrum coefficients as presented as vertical bars against every frequency. The number of frequencies is the same as the number of observations, so this power spectrum for our mini series looks as this:

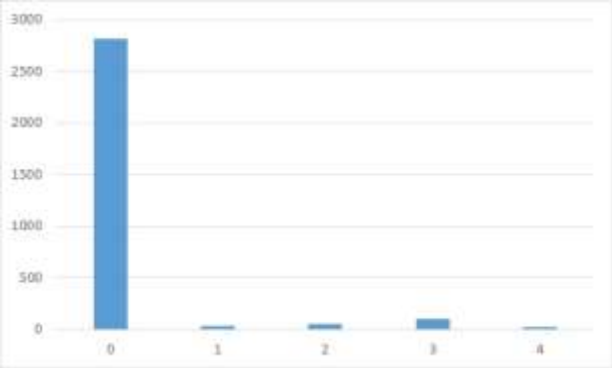


Fig. 13.1.14

As we can see, the graph is very much like the amplitudes graph, but because the first value is very large, all the subsequent values look very small. In practise, you would omit showing the amplitude for zero frequency, as it distorts the picture. The squaring of the amplitudes will help us see larger amplitudes even better, if we drop the zero-frequency power coefficient.

However, we can make some improvements. In our case the frequencies were not defined, so we took them as just sequential numbers. However, if there is a time stamp for every observation, then we can calculate the frequencies. For example, if data represent monthly value, and we know that one month is one twelfth of the year, then the frequency for every DFT that we calculate will be 1/12, which is 0.083. In this case, 0.083 will be the first frequency, the next one will be 2 x 0.083 (0.166), the third one 3 x 0.083 (0.25), etc.

What happens if we are taking specific measurements, like microphone recording, for example? Let's say that we captured 256 points (N=256), in 3 seconds (T=3). The time between every point is $\Delta t = T/N = 3/256 = 0.0117$ seconds. From this value, we can calculate the sampling frequency, which is $f_s = 1/\Delta t = 1/0.0117 = 85.3$. The sampling frequency is given in Hz, which means that we have captured 85.3 signals in one second. And finally, we can also calculate the frequency increments, which are $\Delta f = 1/T = 1/3 = 0.33$. When we calculate our power coefficients, they will be presented as bars against these frequencies, in other words, the first frequency will be 0.33, the second one 2 x 0.33 (0.66), 3 x 0.33 (0.99), 4 x 0.33 (1.33), etc.

You noticed that in the above simple example with only 8 observations we calculated only five power coefficients. There is a reason why it is not necessary to go beyond certain number of power coefficients.

We calculated DFTs for the full range of frequencies, which in our simple case was 8. However, if we look at the graph below, which is a typical power spectrum graph of a much larger time series, we can see that the left hand side of the picture is repeated on the right hand side of the picture.

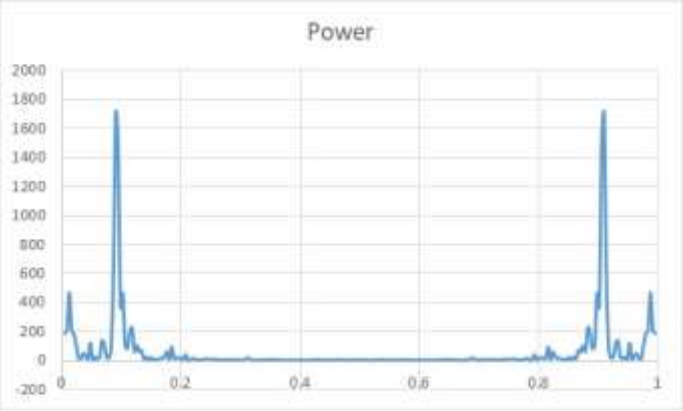


Fig. 13.1.15

This means that we only need half of the coefficients. The frequency at which the data begin to repeat themselves is called a folding frequency and it has been suggested by Nyquist, so some people call it the Nyquist frequency. Effectively, folding frequency is $f_f=f_s/2$ (if continue with an example for 256 points, then $f_f = 85.33/2 = 42.6$). This frequency is also expressed in Hz, so it means that there is no point calculating, or showing, the magnitudes and power coefficients beyond this frequency. As we can see, it will always be half way through the range.

If we take our simple 8-point case, the specific values are:

N = 8 observations
T = 2 sec
$\Delta t = T/N = 2/8 = 0.25$ sec
$f_s = 1/\Delta t = 1/0.25 = 4$ Hz
$\Delta f = 1/T = 1/2 = 0.5$ a frequency increment in Hz
$f_f = f_s/2 = 4/2 = 2$ Hz maximum frequency value for calculating power coefficients

So, the frequency increments will be 0.5 Hz, starting from zero, and will go to a max value of 2 Hz. In this case, our final power spectrum looks as follows:

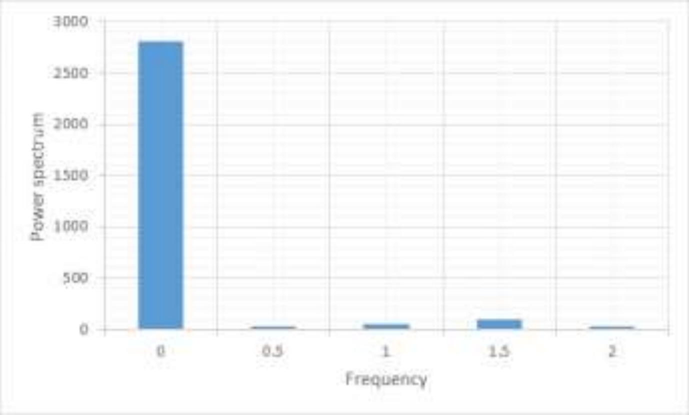
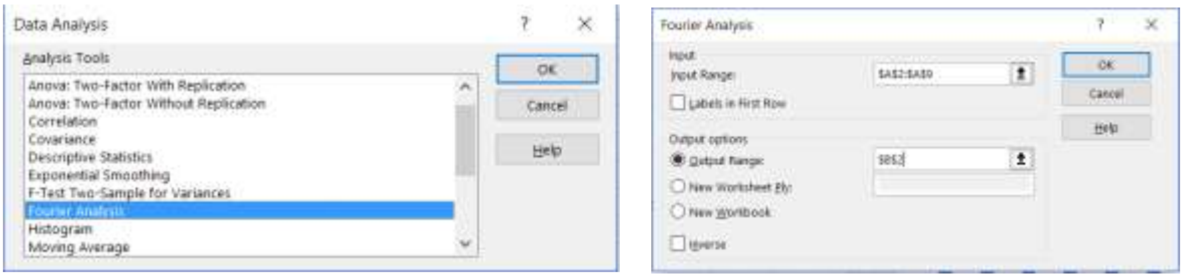


Fig. 13.1.16

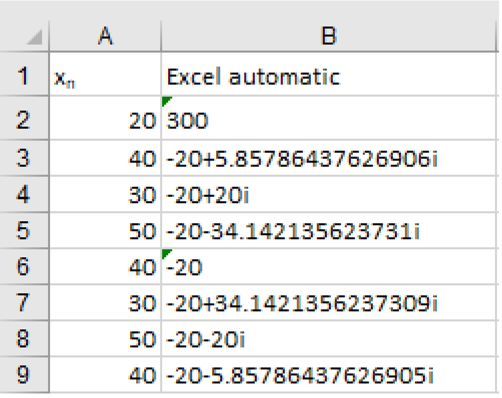
That's all there is to it. If we used Excel, the calculations would have been even easier. Excel has a built in FFT function and you can find it under Data tab, and then under Data Analysis option. Once you clicked on it, it will bring a dialogue box and you need to select Fourier Analysis option. This option brings another dialogue box:



Input Range: $A$2:$A$9
□ Labels in First Row
● Output Range: $B$2
○ New Worksheet Ply:
○ New Workbook
□ Inverse

Fig. 13.1.17

Our simple case of eight values is in A2:A9 and the result we would like to be placed starting from cell B2. What we get is as follows:

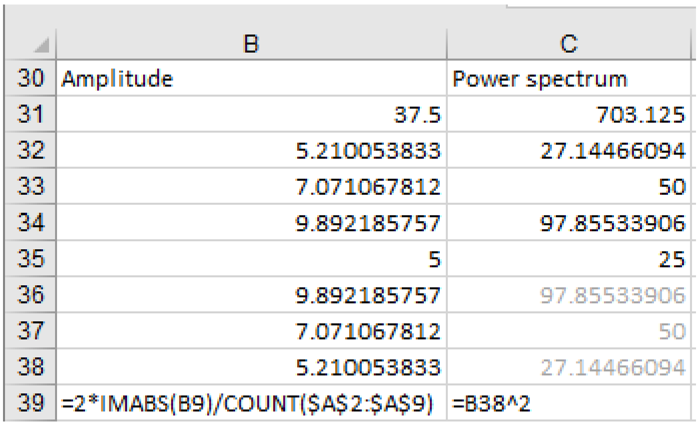


	A	B
1	x_n	Excel automatic
2	20	300
3	40	-20+5.85786437626906i
4	30	-20+20i
5	50	-20-34.142135623731i
6	40	-20
7	30	-20+34.1421356237309i
8	50	-20-20i
9	40	-20-5.85786437626905i

Fig. 13.1.18

Column B shows the values obtained by Excel by applying FFT (Fast Fourier Transforms), which in our case, because we took discrete values are, called DFT (Discrete Fourier Transforms). From these values, we need to calculate the amplitudes, and finally the power spectrum coefficients. To calculate the magnitudes, we use a simple Excel function for calculating the modulus, which is called =IMABS(). To convert the magnitudes into amplitudes, we multiply them by 2 and divide by the number of observations. If our Fourier coefficients were in cells B2:B9 and let’s say we want to put the amplitudes in B31:B38, then in every cell we have a formula =2*IMABS(B2)/COUNT($A$2:$A$9), =2*IMABS(B3)/COUNT($A$2:$A$9), etc. The only exception is cell B31, which is just =IMABS(B3)/COUNT($A$2:$A$9) as per equation (13.1.23), described below.

And finally, to get the power spectrum coefficients, we just multiply A_k × A_k, except that the first one has to be divided by 2. We calculated them in cells C30:C38. The values for both A_k and P_k are given below. Again cell C31 is an exception, as per equation (13.1.27) below.



	B	C
30	Amplitude	Power spectrum
31	37.5	703.125
32	5.210053833	27.14466094
33	7.071067812	50
34	9.892185757	97.85533906
35	5	25
36	9.892185757	97.85533906
37	7.071067812	50
38	5.210053833	27.14466094
39	=2*IMABS(B9)/COUNT($A$2:$A$9)	=B38^2

Fig. 13.1.19

Let’s just recap what we explained here and emphasise that the first amplitudes and powers are calculated differently from the rest..

To calculate properly normalized amplitudes A_k, we differentiate the case for k=0 and all other for k=1, 2, ..., ($\frac{N}{2}$ -1).

For k=0, the amplitude is calculated as:

$$A_k = \frac{1}{N}\sum_{n=0}^{N-1} x_n \qquad (13.1.23)$$

For k=1, 2, ..., ($\frac{N}{2}$-1), the amplitudes are:

$$A_k = \frac{2}{N}\sum_{n=0}^{N-1} x_n\, e^{\frac{-i2\pi nk}{N}} \qquad 13.1.24)$$

Effectively, the equations (13.1.23) and (13.1.24) can be rewritten as:

For k=0 $$A_k = \frac{1}{N}|F_k| \qquad (13.1.25)$$

For k>0 $$A_k = \frac{2}{N}|F_k| \qquad (13.1.26)$$

The Fourier power is calculated as:

For k=0 $$PSD_0(F_k) = \frac{A_k^2}{2} \qquad (13.1.27)$$

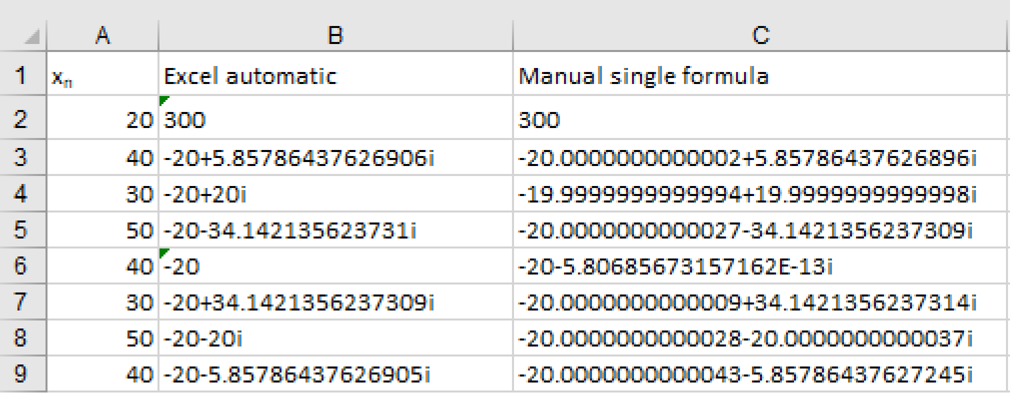
For k>0 $$PSD_x(F_k) = A_k^2 \qquad (13.1.28)$$

Sometimes, the power spectrum is specified as:

$$PSD_x(f_k) = \frac{F_k F_k^*}{N^2} \qquad (13.1.29)$$

Where, F_k^* is a complex conjugate of F_k, which means that the sign of the imaginary part of F_k is reversed. Remember that if you multiply a complex number with its conjugate, you always get the same outcome, e.g. (a+bi) × (a-bi) = a^2 + b^2, which takes us back to our equation (13.1.19).

What if you wanted to check that Excel correctly calculated the Fourier transforms? Well, this can be done easily. In C2:C9 below we are showing the same values calculated using Excel functions as opposed to cells B2:B9 that were calculate using Excel Ad in for Fourier Analysis.



	A	B	C
1	x_n	Excel automatic	Manual single formula
2	20	300	300
3	40	-20+5.85786437626906i	-20.0000000000002+5.85786437626896i
4	30	-20+20i	-19.9999999999994+19.9999999999998i
5	50	-20-34.142135623731i	-20.0000000000027-34.1421356237309i
6	40	-20	-20-5.80685673157162E-13i
7	30	-20+34.1421356237309i	-20.0000000000009+34.1421356237314i
8	50	-20-20i	-20.0000000000028-20.0000000000037i
9	40	-20-5.85786437626905i	-20.0000000000043-5.85786437627245i

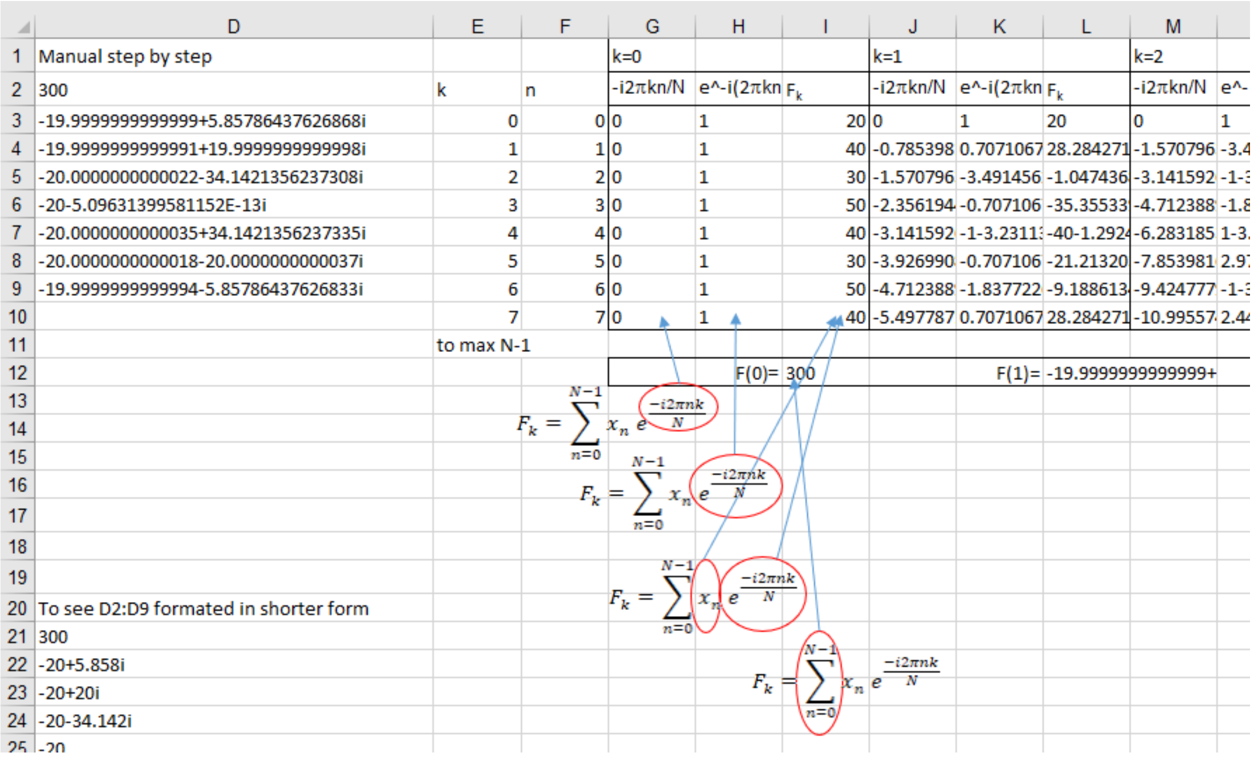
Fig. 13.1.20

The formula for C2 is very long and it reads:

=IMSUM(IMPRODUCT(A2,IMEXP(COMPLEX(0,-(IMPRODUCT(2*PI()*0*0)/8)))),IMPRODUCT(A3,IMEXP(COMPLEX(0,-(IMPRODUCT(2*PI()*0*1)/8)))),IMPRODUCT(A4,IMEXP(COMPLEX(0,-(IMPRODUCT(2*PI()*0*2)/8)))),IMPRODUCT(A5,IMEXP(COMPLEX(0,-(IMPRODUCT(2*PI()*0*3)/8)))),IMPRODUCT(A6,IMEXP(COMPLEX(0,-(IMPRODUCT(2*PI()*0*4)/8)))),IMPRODUCT(A7,IMEXP(COMPLEX(0,-(IMPRODUCT(2*PI()*0*5)/8)))),IMPRODUCT(A8,IMEXP(COMPLEX(0,-(IMPRODUCT(2*PI()*0*6)/8)))),IMPRODUCT(A9,IMEXP(COMPLEX(0,-(IMPRODUCT(2*PI()*0*7)/8)))))

The cells below C3:C9 repeat this formula, but with the relevant change in parameters. Essentially, we multiplying two complex numbers, hence the formula =IMPRODUCT(complex number1, complex number2). In our case the first complex number is given by A2, A3, etc. The second complex number is given by function =COMPLEX(IMPRODUCT(0, (2*PI()*k*n/N). As n goes from 0 to 7, we used the function =IMSUM() to collect all the values.

If this formula scares you, you can see in the enclosed spreadsheet how it was built. In D2:D9 we are showing gradual build up that resulted with the identical numbers as obtained with one single formula, or as with automated Excel routine for Fourier analysis.



	D	E	F	G	H	I	J	K	L	M	
1	Manual step by step			k=0			k=1			k=2	
2	300	k	n	-i2πkn/N	e^-i(2πkn	F_k	-i2πkn/N	e^-i(2πkn	F_k	-i2πkn/N	e^-
3	-19.9999999999999+5.85786437626868i	0	0	0	1	20	0	1	20	0	1
4	-19.9999999999991+19.9999999999998i	1	1	0	1	40	-0.785398	0.7071067	28.284271	-1.570796	-3.4
5	-20.0000000000022-34.1421356237308i	2	2	0	1	30	-1.570796	-3.491456	-1.047436	-3.141592	-1-3
6	-20-5.09631399581152E-13i	3	3	0	1	50	-2.356194	-0.707106	-35.35533	-4.712388	-1.8
7	-20.0000000000035+34.1421356237335i	4	4	0	1	40	-3.141592	-1-3.2311	-40-1.292	-6.283185	1-3.
8	-20.0000000000018-20.0000000000037i	5	5	0	1	30	-3.926990	-0.707106	-21.21320	-7.853981	2.9
9	-19.9999999999994-5.85786437626833i	6	6	0	1	50	-4.712388	-1.837722	-9.188613	-9.424777	-1-3
10		7	7	0	1	40	-5.497787	0.7071067	28.284271	-10.99557	2.4
11		to max N-1									
12					F(0)=	300		F(1)=	-19.9999999999999+		
13											
14											
15											
16											
17											
18											
19											
20	To see D2:D9 formated in shorter form										
21	300										
22	-20+5.858i										
23	-20+20i										
24	-20-34.142i										
25	-20										

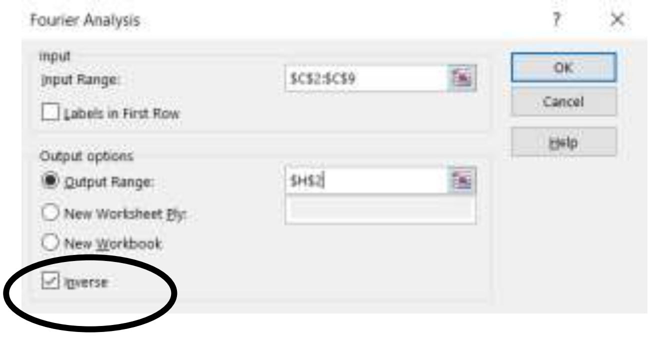
$F_k = \sum_{n=0}^{N-1} x_n e^{\frac{-i2\pi nk}{N}}$
$F_k = \sum_{n=0}^{N-1} x_n e^{\frac{-i2\pi nk}{N}}$
$F_k = \sum_{n=0}^{N-1} x_n e^{\frac{-i2\pi nk}{N}}$
$F_k = \sum_{n=0}^{N-1} x_n e^{\frac{-i2\pi nk}{N}}$

Fig. 13.1.21

As you can see, although we can do it, it does not mean that we should. We should get Excel calculate Fourier transforms and then just calculate the magnitudes from them (using the =IMABS() function) and finally the power spectrum coefficients.

Before we proceed, just a short digression. The complex coefficients in cells D3:D9 in Fig. 13.1.21 are not formatted by Excel and they appear to be long and messy numbers. If you would like to shorten them, a combination of several Excel functions can be used to achieve this. In Fig. 13.1.21 cells D21:D28 are identical to cells D2:D9, but they are now showing to just three decimal places. The formula we used in cell D21 and below is a combination of Excel functions:
=COMPLEX(VALUE(TEXT(IMREAL(D9),"0.000")),VALUE(TEXT(IMAGINARY(D9),"0.000")))
.

We said that Fourier transforms is just another way of representing the same data set. This means that we can revert the series of Fourier transforms back into the original time series from which they were created. The only difference is that in the dialogue box (see Fig. 13.1.22) the input range is the Transform series and we have to tick the Inverse box.

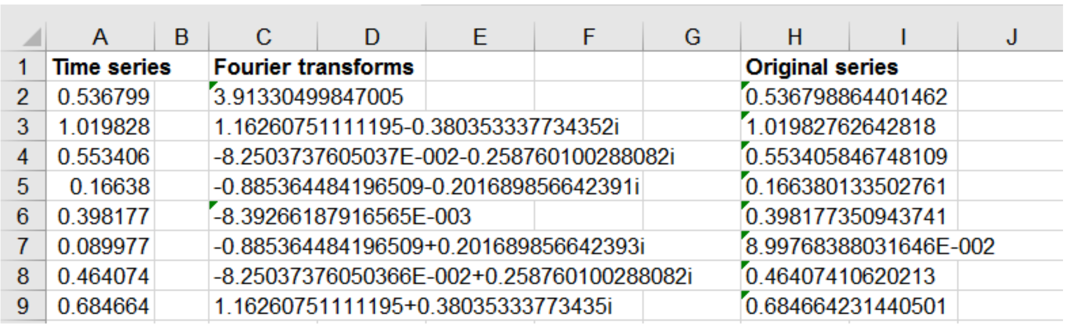


Input
Input Range: $C$2:$C$9
☐ Labels in First Row

Output options
● Output Range: $H$2
○ New Worksheet Ply:
○ New Workbook
☑ Inverse

Fig. 13.1.22

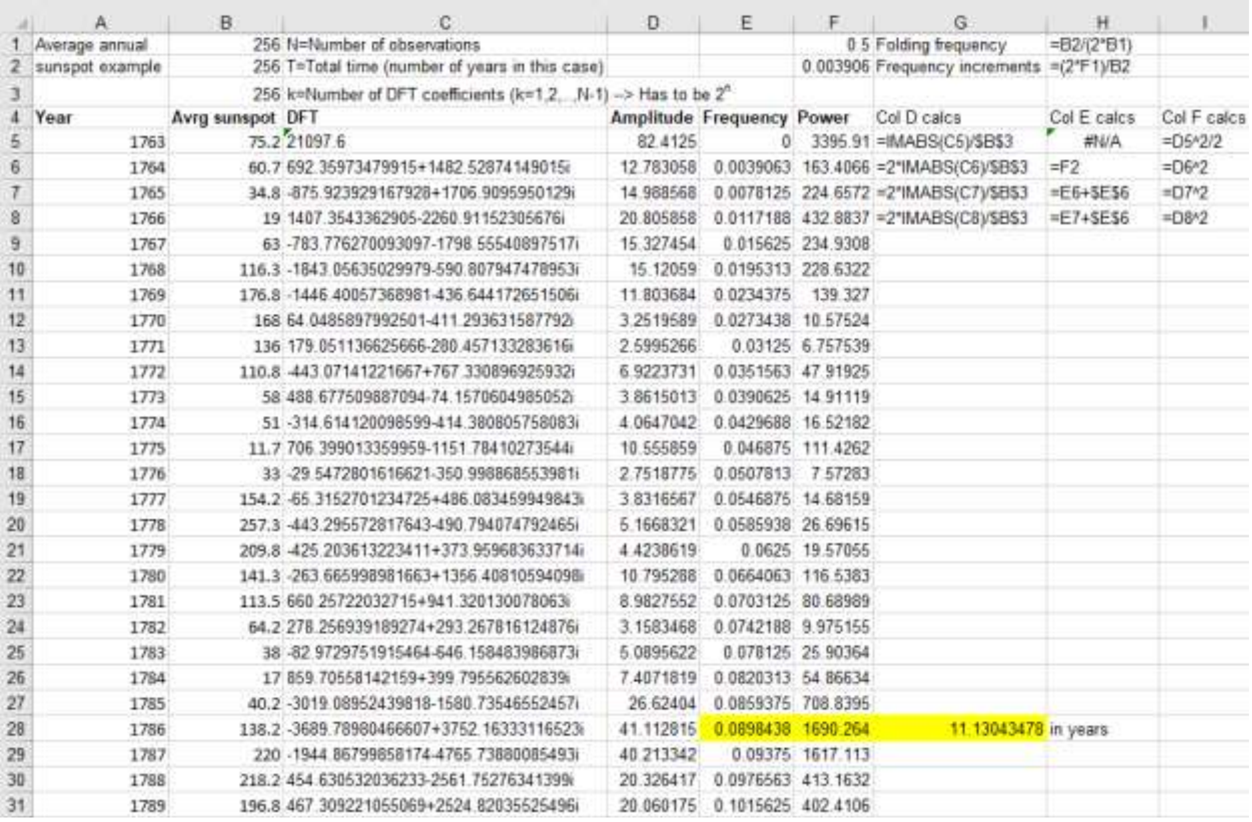
Below in Fig. 13.1.23 we used a simple example to demonstrate how the original series was reconstructed from the inverse Fourier transforms with almost perfection. In column C we have Fourier transforms calculated by Excel. Imagine that this was your starting point, i.e. someone gave you just the list of Fourier transforms. To re-construct the actual time series from them, as we did in column H, use the dialogue box as described in Fig. 13.1.22. See that our column H is practically identical to column A, which proves the point.



	A	B	C	D	E	F	G	H	I	J
1	Time series		Fourier transforms					Original series		
2	0.536799		3.91330499847005					0.536798864401462		
3	1.019828		1.16260751111195-0.380353337734352i					1.01982762642818		
4	0.553406		-8.2503737605037E-002-0.258760100288082i					0.553405846748109		
5	0.16638		-0.885364484196509-0.201689856642391i					0.166380133502761		
6	0.398177		-8.39266187916565E-003					0.398177350943741		
7	0.089977		-0.885364484196509+0.201689856642393i					8.99768388031646E-002		
8	0.464074		-8.25037376050366E-002+0.258760100288082i					0.46407410620213		
9	0.684664		1.16260751111195+0.38035333773435i					0.684664231440501		

Fig. 13.1.23 Fourier analysis results

To close the chapter, we’ll do one more example and show how to correctly interpret the frequencies in the case where the frequency interval is not Hz, but years. Our example shows average annual sunspot number from 1763 until 2018. Fig 13.1.24 shows only the first 27 rows.



	A	B	C	D	E	F	G	H	I
1	Average annual	256	N=Number of observations			0.5	Folding frequency	=B2/(2*B1)	
2	sunspot example	256	T=Total time (number of years in this case)			0.003906	Frequency increments	=(2*F1)/B2	
3		256	k=Number of DFT coefficients (k=1,2,...,N-1) --> Has to be 2^n						
4	**Year**	**Avrg sunspot**	DFT	**Amplitude**	**Frequency**	**Power**	Col D calcs	Col E calcs	Col F calcs
5	1763	75.2	21097.6	82.4125	0	3395.91	=IMABS(C5)/$B$3	#N/A	=D5^2/2
6	1764	60.7	692.35973479915+1482.52874149015i	12.783058	0.0039063	163.4066	=2*IMABS(C6)/$B$3	=F2	=D6^2
7	1765	34.8	-875.923929167928+1706.9095950129i	14.988568	0.0078125	224.6572	=2*IMABS(C7)/$B$3	=E6+$E$6	=D7^2
8	1766	19	1407.3543362905-2260.91152305676i	20.805858	0.0117188	432.8837	=2*IMABS(C8)/$B$3	=E7+$E$6	=D8^2
9	1767	63	-783.776270093097-1798.55540897517i	15.327454	0.015625	234.9308			
10	1768	116.3	-1843.05635029979-590.807947478953i	15.12059	0.0195313	228.6322			
11	1769	176.8	-1446.40057368981-436.644172651506i	11.803684	0.0234375	139.327			
12	1770	168	64.0485897992501-411.293631587792i	3.2519589	0.0273438	10.57524			
13	1771	136	179.051136625666-280.457133283616i	2.5995266	0.03125	6.757539			
14	1772	110.8	-443.07141221667+767.330896925932i	6.9223731	0.0351563	47.91925			
15	1773	58	488.677509887094-74.1570604985052i	3.8615013	0.0390625	14.91119			
16	1774	51	-314.614120098599-414.380805758083i	4.0647042	0.0429688	16.52182			
17	1775	11.7	706.399013359959-1151.78410273544i	10.555859	0.046875	111.4262			
18	1776	33	-29.5472801616621-350.998868553981i	2.7518775	0.0507813	7.57283			
19	1777	154.2	-65.3152701234725+486.083459949843i	3.8316567	0.0546875	14.68159			
20	1778	257.3	-443.295572817643-490.794074792465i	5.1668321	0.0585938	26.69615			
21	1779	209.8	-425.203613223411+373.959683633714i	4.4238619	0.0625	19.57055			
22	1780	141.3	-263.665998981663+1356.40810594098i	10.795288	0.0664063	116.5383			
23	1781	113.5	660.25722032715+941.320130078063i	8.9827552	0.0703125	80.68989			
24	1782	64.2	278.256939189274+293.267816124876i	3.1583468	0.0742188	9.975155			
25	1783	38	-82.9729751915464-646.158483986873i	5.0895622	0.078125	25.90364			
26	1784	17	859.70558142159+399.795562602839i	7.4071819	0.0820313	54.86634			
27	1785	40.2	-3019.08952439818-1580.73546552457i	26.62404	0.0859375	708.8395			
28	1786	138.2	-3689.78980466607+3752.16333116523i	41.112815	0.0898438	1690.264	11.13043478	in years	
29	1787	220	-1944.86799858174-4765.73880085493i	40.213342	0.09375	1617.113			
30	1788	218.2	454.630532036233-2561.75276341399i	20.326417	0.0976563	413.1632			
31	1789	196.8	467.309221055069+2524.82035525496i	20.060175	0.1015625	402.4106			

Fig. 13.1.24

As we can see from cells B1:B3, we have 256 observations taken over 256 years. Cell F1 tells us that the folding frequency is 0.5, which means that we will calculate the power coefficients only up to this frequency. We will start with the first frequency equal to 0 and then add 0.003906 increments (see cell F2) until we reach the value of 0.5, which is the folding frequency.

From Fig. 13.1.24 in column C we can see the DFT coefficients calculated by Excel, column D shows the amplitudes, column E the frequencies, calculated as stated in the paragraph above, and column F shows the power coefficients. All of this was calculated as explained previously in this chapter, so we will not repeat explanations here. However, we will show the final results and provide some further explanations. Fig. 13.1.25 depicts the final result as a chart.

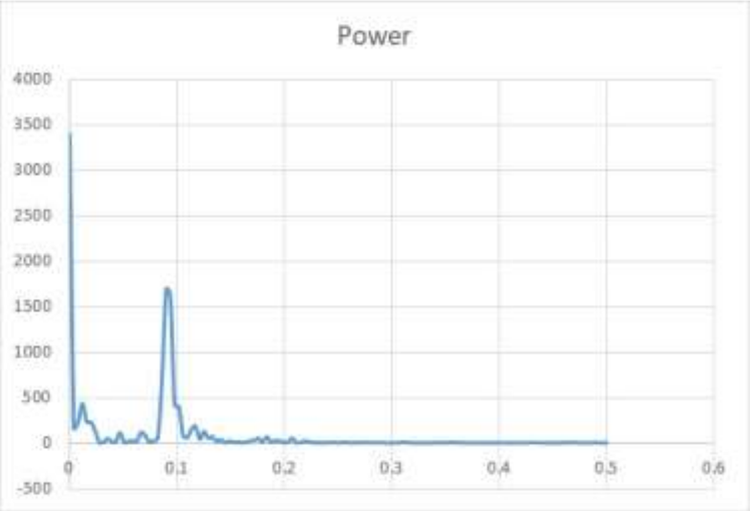


Fig. 13.1.25

In Fig. 13.1.25 we are showing all 128 coefficients against the frequency. It stops at 0.5. We know that if we continued until all 256 coefficients were shown, we would get a mirror image picture (see fig. 13.1.15) of the first 128 coefficients, which is the reason we stopped at 128th coefficient, which corresponds to the frequency of 0.5.

If we eliminate the zero frequency, to see better, we get a graph as in Fig. 13.1.26. Next to it is the same graph, but the x axis is shown as logarithmic. Some software packages do this, so we just wanted to show that it has no relevance to our results.

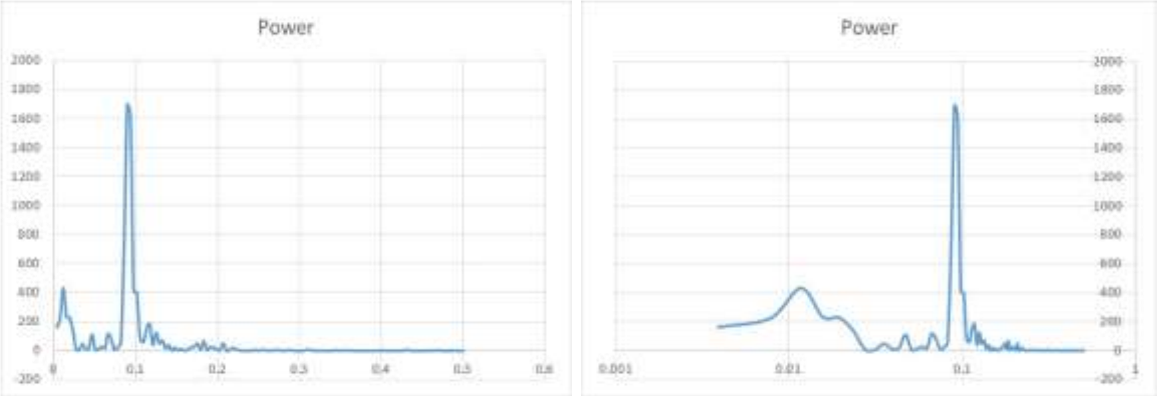


Fig. 13.1.26

What we can is that the highest magnitude takes place at the frequency of 0.0898 (see graph and cell F28 in Fig. 13.1.24). But what does that mean? Simple, if you take a reciprocal of this number (remember T=1/f), you will get the value of 11.13. This simple step has moved you back from the world of frequencies into the world of time but expressed as periodogram. Because our values are annual, 11.13 are the years and it means that sunspots have periodicity of approximately 11 years, which any astronomer can confirm is the fact.

I conducted another experiment, not shown here but you can find it in the attached spreadsheet where I calculated the power coefficients for monthly sunspot data starting from May 1934 until August 2019. This covers 1024 observations, so I calculated the same amount of power coefficients. However, the point to remember is how the frequencies were calculated and converted back into the time domain. We have the following calculation:

N = 1024 observations

Δt = 1/12 = 0.08333 this is the sampling interval, because we have monthly data

f_f = 1/2Δt = 1/(2*0.08333) = 6 is the folding frequency

f_s = 1/Δt = 1/0.08333 = 12 is the sampling frequency in months

Δf = (2f_f)/N = (2*6/1024) = 0.01171875 is a frequency increment

T = (NΔt) = 1024*0.08333 = 85.3333 is the number of years as a decimal number

If you subtract from the end data the start date, you get 85 years, or 1024 months as the total time. As a decimal number, this can be expressed as 85.3333. This number is also a reciprocal if you take 1/Δf (1/0.01171875).

When I completed my calculations, the highest amplitude is associated with the frequency value of 0.09375. If we take the reciprocal value of that number (1/0.09375), we get 10.7 years, or if expressed as an integer this is ≈11 years. Again, the same periodicity as for the annual data. The key here is to realize how to create and convert frequencies from the time domain and back.

To assist with different time units, sampling frequencies and increments in time, as well as in frequency, we list here some of the fundamental equations to be remembered. To begin with, here are the symbols that we will use:

N = Total number of discrete data points taken
Δt = Time between data points (sampling period)
T = Total sampling time
f_s = Sampling rate (in Hz, for example)
f_f = Folding frequency or Nyquist limit
Δf = Frequency increments

We'll define the time increments Δt in which the signal is recorded as:

$$\Delta t = \frac{T}{N} = \frac{1}{f_s} \qquad (13.1.30)$$

From there, we can extract any of the variables:

$$T = N\Delta t = \frac{N}{f_s} \qquad (13.1.31)$$

$$N = \frac{T}{\Delta t} = f_s T \qquad (13.1.32)$$

$$f_s = \frac{1}{\Delta t} = \frac{N}{T} \qquad (13.1.33)$$

The folding frequency (Nyquist limit frequency) is defined as:

$$f_f = \frac{f_s}{2} = \frac{1}{2\Delta t} = \frac{N}{2T} \tag{13.1.34}$$

The frequency increments Δf, which will be used to create frequency bins are defined as:

$$\Delta f = \frac{1}{T} = \frac{1}{N\Delta t} = \frac{f_s}{N} = \frac{2f_f}{N} \tag{13.1.35}$$

As we already know, according to the Nyquist criterion, the maximum number of DFTs that should be calculated should not exceed $f_s/2$. This mean that there is a number k, which we multiply with the number of frequency increments Δf, i.e. kΔf, which should equal to $f_s/2$.

We define kΔf as:

$$k\Delta f = \frac{f_s}{2} \tag{13.1.36}$$

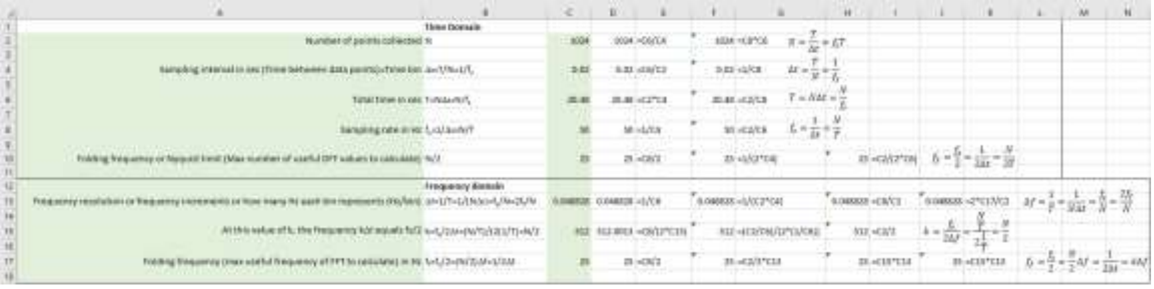
From (13.1.36), we can extract k as:

$$k = \frac{f_s}{2\Delta f} = \frac{\frac{N}{T}}{2\frac{1}{T}} = \frac{N}{2} = \frac{f_f}{\Delta f} \tag{13.1.37}$$

From there we can provide alternative equations for (13.1.34) as:

$$f_f = \frac{f_s}{2} = \frac{N}{2}\Delta f = k\Delta f \tag{13.1.38}$$

In Fig. 13.1.27 below we show how some of these variables can be extracted from one another. The printout is too small, but the actual spreadsheet with all the solutions is available together with all other Excel resources for this book.



A	B	C	D–E	F–G	H–I	J–K	L–N
	Time Domain						
Number of points collected	N	1024	1024 =C6/C4	1024 =C8*C6	$N = \frac{T}{\Delta t} = f_s T$		
Sampling interval in sec (Time between data points)=Time bin	$\Delta t = T/N = 1/f_s$	0.02	0.02 =C6/C2	0.02 =1/C8	$\Delta t = \frac{T}{N} = \frac{1}{f_s}$		
Total time in sec	$T = N\Delta t = N/f_s$	20.48	20.48 =C2*C4	20.48 =C2/C8	$T = N\Delta t = \frac{N}{f_s}$		
Sampling rate in Hz	$f_s = 1/\Delta t = N/T$	50	50 =1/C4	50 =C2/C6	$f_s = \frac{1}{\Delta t} = \frac{N}{T}$		
Folding frequency or Nyquist limit (Max number of useful DFT values to calculate)	fs/2	25	25 =C8/2	25 =1/(2*C4)	25 =C2/(2*C6)	$f_f = \frac{f_s}{2} = \frac{1}{2\Delta t} = \frac{N}{2T}$	
	Frequency domain						
Frequency resolution or frequency increments or how many Hz each bin represents (Hz/bin)	$\Delta f = 1/T = 1/(N\Delta t) = f_s/N = 2f_f/N$	0.048828	0.048828 =1/C6	0.048828 =1/(C2*C4)	0.048828 =C8/C2	0.048828 =2*C17/C2	$\Delta f = \frac{1}{T} = \frac{1}{N\Delta t} = \frac{f_s}{N} = \frac{2f_f}{N}$
At this value of k, the frequency kΔf equals fs/2	$k = f_s/2\Delta f = (N/T)/(2(1/T)) = N/2$	512	512.0001 =C8/(2*C13)	512 =(C2/C6)/(2*(1/C6))	512 =C2/2	$k = \frac{f_s}{2\Delta f} = \frac{\frac{N}{T}}{2\frac{1}{T}} = \frac{N}{2}$	
Folding frequency (max useful frequency of FFT to calculate) in Hz	$f_f = f_s/2 = (N/2)\Delta f = 1/2\Delta t$	25	25 =C8/2	25 =C2/2*C13	25 =C15*C13	25 =C15*C13	$f_f = \frac{f_s}{2} = \frac{N}{2}\Delta f = \frac{1}{2\Delta t} = k\Delta f$

Fig. 13.1.27 Excel solutions for extracting various variables

Just to have total clarity about how some of these variables are used and interpreted, we’ll go through one more brief example, this time typical for the engineering community (using Hz).

As an example, we take measurements over 2.56 seconds (T=2.56) at the rate of 100 Hz, which is f_s, i.e. 100 readings per second. This means that in total we have 256 observations ($N = T \times f_s$). From there, we can calculate that we were taking samples every 0.01 second ($\Delta t = T/N = 2.56/256$).

As we will be using Excel that calculates DFTs using the FFT algorithm (more about that very soon), only 2^n number of Fourier transforms can be calculated (8, 16,32,64, 128, etc.), so we decide to go for 256 DFT coefficients (k=256).

From there, we calculate what our frequency increments will be ($\Delta f=1/T$ or $\Delta f=f_s/N$), which is 0.39. This means that the frequency bins will be:

$f_0=0$,
f1=0+0.39=0.39,
$f_2=0.39+0.39=0.78$,
$f_3=0.78+0.39=1.17$, …,
$f_{256}=99.22+0.39=99.61$ and
$f_{257}=99.61+0.39=100.00$.

This covers the full range of frequencies from 0 to 100 Hz. However, we know that we do not need to cover the whole range. We can calculate discrete Fourier transforms up to the Nyquist frequency, or the folding frequency ($f_f=f_s/2=50$Hz). The total number of discrete Fourier transforms that we need is 128 (k=128). Now that we have the frequency bins and know that we need only 128 Fourier transform coefficients, we can proceed with the calculations.

Fortunately, if we do not handle engineering data, we do not have to worry about majority of these details. We still just need the folding frequency and frequency increments.

And finally, what happens if you have longer, or shorter time series than 2^n number of observations? If it is longer, you just do not take into calculations any observations beyond 2^n. If it is short, you just add zeros to the missing time intervals to reach the 2^n number. Here is an example.

You have two annual time series, one starts in the year 1953 and finishes in 2019 and the other one starts in 1963 and finishes in 2019. This means the first one is 66 years long and the second one is 56 years long. You want to calculate 64 discrete Fourier transforms for each time series and compare them. For the first one, ignore the first two years and calculate DFTs between 1955 and 2019. For the second one, just add eight zeros below the actual readings of the observation so that in total you have 64 observations (though the last 8 will be zeros). This is called “zero padding” and will give you the correct value of the DFTs even if your time series was shorter.