LINEAR ALGEBRA. LECTURE 9

The Fast Fourier Transform

The Fourier series is linear algebra in infinite dimensions. The "vectors" are functions $f(x)$; they are projected onto the sines and cosines; that produces the Fourier coefficients a_k and b_k. From this infinite sequence of sines and cosines, multiplied by a_k and b_k, we can reconstruct $f(x)$. That is the classical case, which Fourier dreamt about, but in actual calculations it is the ***discrete Fourier transform*** that we compute. Fourier still lives, but in finite dimensions.

This is pure linear algebra, based on orthogonality. The input is a sequence of numbers $y_0, \dots, y_{n-1}$, instead of a function $f(x)$. The output $c_0, \dots, c_{n-1}$ has the same length n. The relation between y and c is linear, so it must be given by a matrix. This is the ***Fourier matrix*** F, and the whole technology of digital signal processing depends on it. The Fourier matrix has remarkable properties.

Signals are digitized, whether they come from speech or images or sonar or TV (or even oil exploration). The signals are transformed by the matrix F, and later they can be transformed back—to reconstruct. What is crucially important is that F and F^{-1} can be quick:

$$\boxed{F^{-1} \textbf{\textit{ must be simple. The multiplications by }} F \textbf{\textit{ and }} F^{-1} \textbf{\textit{ must be fast.}}}$$

Those are both true. F^{-1} has been known for years, and it looks just like F. In fact, F is symmetric and orthogonal (apart from a factor $\sqrt{n}$), and it has only one drawback: Its entries are ***complex numbers***. That is a small price to pay, and we pay it below. The difficulties are minimized by the fact that *all entries of F and F^{-1} tare powers of a single number w*. That number has $w^n = 1$.

Complex Roots of Unity

Real equations can have complex solutions. The equation $x^2 + 1 = 0$ led to the invention of i (and also to $-i$!). That was declared to be a solution, and the case was closed. If someone asked about $x^2 - i = 0$, there was an answer: The square roots of a complex number are again complex numbers. You must allow combinations $x + iy$, with a real part x and an imaginary part y, but no further inventions are necessary. Every real or complex polynomial of degree n has a full set of n roots (possibly complex and possibly repeated). That is the fundamental theorem of algebra.

We are interested in equations like $x^4 = 1$. That has four solutions—the ***fourth roots of unity***. The two square roots of unity are 1 and -1. The fourth roots are the square roots of the square roots, 1 and -1, i and $-i$. The number i will satisfy $i^4 = 1$ because it satisfies $i^2 = -1$. For the eighth roots of unity we need the square roots of i, and that brings us to $w = (1+i)/\sqrt{2}$. Squaring w produces $(1 + 2i + i^2)/2$, which is i—because $1 + i^2$ is zero. Then $w^8 = i^4 = 1$. There has to be a system here.

The complex numbers $\cos\theta + i\sin\theta$ in the Fourier matrix are extremely special. The real part is plotted on the x-axis and the imaginary part on the y-axis (Figure 3.11). Then the number w lies on the ***unit circle***; its distance from the origin is $\cos^2\theta + \sin^2\theta = 1$.

Here we need only special points w, all of them on the unit circle, in order to solve $w^n = 1$.

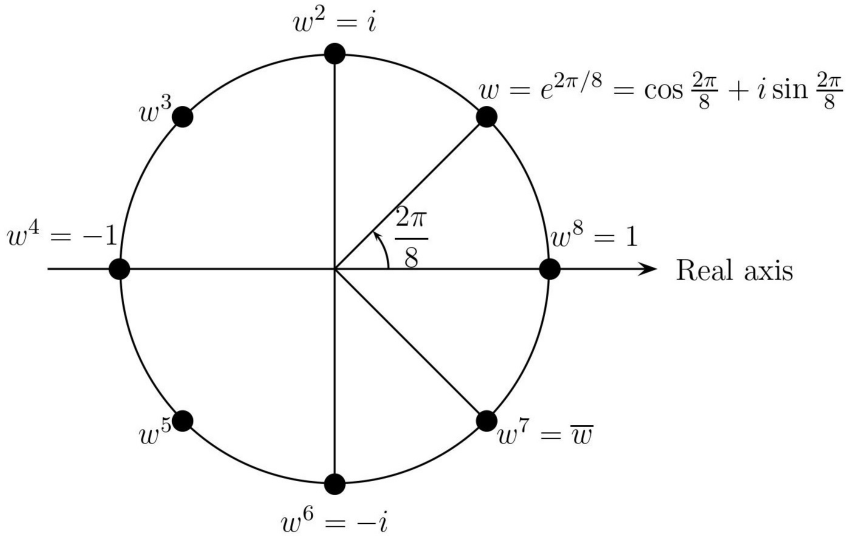

Figure 3.11: The eight solutions to $z^8 = 1$ are $1, w, w^2, \ldots, w^7$ with $w = (1+i)/\sqrt{2}$.

The square of w can be found directly (it just doubles the angle):

$$w^2 = (\cos\theta + i\sin\theta)^2 = \cos^2\theta - \sin^2\theta + 2i\sin\theta\cos\theta.$$

The real part $\cos^2\theta - \sin^2\theta$ is $\cos 2\theta$, and the imaginary part $2\,\sin\theta\cos\theta$ is $\sin 2\theta$. (Note that i is not included; the imaginary part is a real number.) Thus $w^2 = \cos 2\theta + i\sin 2\theta$. The square of w is still on the unit circle, but ***at the double angle*** 2θ. That makes us suspect that w^n lies at the angle $n\theta$, and we are right.

There is a better way to take powers of w. The combination of cosine and sine is a complex exponential, with amplitude one and phase angle θ:

$$\cos\theta + i\sin\theta = e^{i\theta}. \tag{2}$$

The rules for multiplying, like $(e^2)(e^3) = e^5$, continue to hold when the exponents $i\theta$ are imaginary. ***The powers of*** $w = e^{i\theta}$ ***stay on the unit circle***:

$$\textbf{Powers of } w \qquad w^2 = e^{i2\theta}, \quad w^n = e^{in\theta}, \quad \frac{1}{w} = e^{-i\theta}. \tag{3}$$

The nth power is at the angle $n\theta$. When $n = -1$, ***the reciprocal*** $1/w$ ***has angle*** $-\theta$. If we multiply $\cos\theta + i\sin\theta$ by $\cos(-\theta) + i\sin(-\theta)$, we get the answer 1:

$$e^{i\theta}e^{-i\theta} = (\cos\theta + i\sin\theta)(\cos\theta - i\sin\theta) = \cos^2\theta + \sin^2\theta = 1.$$

With this formula, we can solve $w^n = 1$. It becomes $e^{in\theta} = 1$, so that $n\theta$ must carry us around the unit circle and back to the start. The solution is to choose $\theta = 2\pi/n$: *The "primitive" nth root of unity is*

$$w_n = e^{2\pi i/n} = \cos\frac{2\pi}{n} + i\sin\frac{2\pi}{n}. \tag{4}$$

Its nth power is $e^{2\pi i}$, which equals 1. For $n = 8$, this root is $(1+i)/\sqrt{2}$:

$$w_4 = \cos\frac{\pi}{2} + i\sin\frac{\pi}{2} = i \qquad \text{and} \qquad w_8 = \cos\frac{\pi}{4} + i\sin\frac{\pi}{4} = \frac{1+i}{\sqrt{2}}$$

The Fourier Matrix and Its Inverse

In the continuous case, the Fourier series can reproduce $f(x)$ over a whole interval. It uses infinitely many sines and cosines (or exponentials). In the discrete case, with only n coefficients $c_0, \ldots, c_{n-1}$ to choose, we only ask for *equality at n points*. That gives n equations. We reproduce the four values $y = 2,4,6,8$ when $Fc = y$:

$$Fc = y \qquad \begin{array}{ccccccccc} c_0 & + & c_1 & + & c_2 & + & c_3 & = & 2 \\ c_0 & + & ic_1 & + & i^2c_2 & + & i^3c_3 & = & 4 \\ c_0 & + & i^2c_1 & + & i^4c_2 & + & i^6c_3 & = & 6 \\ c_0 & + & i^3c_1 & + & i^6c_2 & + & i^9c_3 & = & 8. \end{array} \tag{6}$$

The input sequence is $y = 2,4,6,8$. The output sequence is c_0, c_1, c_2, c_3. The four equations (6) look for a four-term Fourier series that matches the inputs at four equally spaced points x on the interval from 0 to 2π:

Discrete Fourier Series

$$c_0 + c_1e^{ix} + c_2e^{2ix} + c_3e^{3ix} = \begin{cases} 2 & \text{at} \quad x = 0 \\ 4 & \text{at} \quad x = \pi/2 \\ 6 & \text{at} \quad x = \pi \\ 8 & \text{at} \quad x = 3\pi/2. \end{cases}$$

Those are the four equations in system (6). At $x = 2\pi$ the series returns $y_0 = 2$ and continues periodically. The Discrete Fourier Series is best written in this *complex* form, as a combination of exponentials e^{ikx} rather than $\sin kx$ and $\cos kx$.

For every n, the matrix connecting y to c can be inverted. It represents n equations, requiring the finite series $c_0 + c_1e^{ix} + \cdots$ ***(n terms)*** to agree with y ***(at n points)***. The first agreement is at $x = 0$, where $c_0 + \cdots + c_{n-1} = y_0$. The remaining points bring powers of w, and the full problem is $Fc = y$:

$$Fc = y \qquad \begin{bmatrix} 1 & 1 & 1 & \cdot & 1 \\ 1 & w & w^2 & \cdot & w^{n-1} \\ 1 & w^2 & w^4 & \cdot & w^{2(n-1)} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & w^{n-1} & w^{2(n-1)} & \cdot & w^{(n-1)^2} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \cdot \\ c_{n-1} \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \cdot \\ y_{n-1} \end{bmatrix}. \tag{7}$$

There stands the Fourier matrix F with entries $F_{jk} = w^{jk}$. It is natural to number the rows and columns from 0 to $n-1$, instead of 1 to n. The first row has $j = 0$, the first column has $k = 0$, and all their entries are $w^0 = 1$.

To find the c's we have to invert F. In the 4 by 4 case, F^{-1} was built from $1/i = -i$. That is the general rule, that F^{-1} comes from the complex number $w^{-1} = \overline{w}$. It lies at the angle $-2\pi/n$, where w was at the angle $+2\pi/n$:

3V The inverse matrix is built from the powers of $w^{-1} = 1/w = \overline{w}$:

$$F^{-1} = \frac{1}{n}\begin{bmatrix} 1 & 1 & 1 & \cdot & 1 \\ 1 & w^{-1} & w^{-2} & \cdot & w^{-(n-1)} \\ 1 & w^{-2} & 1 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & w^{-(n-1)} & w^{-2(n-1)} & \cdot & w^{-(n-1)^2} \end{bmatrix} = \frac{\overline{F}}{n}. \tag{8}$$

$$\text{Thus} \quad F = \begin{bmatrix} 1 & 1 & 1 \\ 1 & e^{2\pi i/3} & e^{4\pi i/3} \\ 1 & e^{4\pi i/3} & e^{8\pi i/3} \end{bmatrix} \quad \text{has} \quad F^{-1} = \frac{1}{3}\begin{bmatrix} 1 & 1 & 1 \\ 1 & e^{-2\pi i/3} & e^{-4\pi i/3} \\ 1 & e^{-4\pi i/3} & e^{-8\pi i/3} \end{bmatrix}.$$

Row j of F times column j of F^{-1} is always $(1+1+\cdots+1)/n = 1$. The harder part is off the diagonal, to show that row j of F times column k of F^{-1} gives zero:

$$1 \cdot 1 + w^j w^{-k} + w^{2j} w^{-2k} + \cdots + w^{(n-1)j} w^{-(n-1)k} = 0 \quad \text{if} \quad j \neq k. \tag{9}$$

The key is to notice that those terms are the powers of $W = w^j w^{-k}$:

$$1 + W + W^2 + \cdots + W^{n-1} = 0. \tag{10}$$

This number W is still a root of unity: $W^n = w^{nj} w^{-nk}$ is equal to $1^j 1^{-k} = 1$. Since j is different from k, W is different from 1. It is one of the *other* roots on the unit circle. *Those roots all satisfy* $1 + W + \cdots + W^{n-1} = 0$. Another proof comes from

$$1 - W^n = (1 - W)(1 + W + W^2 + \cdots + W^{n-1}). \tag{11}$$

Since $W^n = 1$, the left side is zero. But W is not 1, so the last factor must be zero. ***The columns of F are orthogonal.***

The Fast Fourier Transform

Fourier analysis is a beautiful theory, and it is also very practical. To analyze a waveform into its frequencies is the best way to take a signal apart. The reverse process brings it back. For physical and mathematical reasons the exponentials are special, and we can pinpoint one central cause: ***If you differentiate e^{ikx}, or integrate it, or translate x to $x+h$, the result is still a multiple of e^{ikx}.*** Exponentials are exactly suited to differential equations, integral equations, and difference equations. Each frequency component goes its own way, as an eigenvector, and then they recombine into the solution. The analysis and synthesis of signals—computing c from y and y from c—is a central part of scientific computing.

We want to show that Fc and $F^{-1}y$ can be done quickly. The key is in the relation of F_4 to F_2—or rather to *two copies* of F_2, which go into a matrix F_2^*:

$$F_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & i^2 & i^3 \\ 1 & i^2 & i^4 & i^6 \\ 1 & i^3 & i^6 & i^9 \end{bmatrix} \quad \text{is close to} \quad F_2^* = \begin{bmatrix} 1 & 1 & & \\ 1 & -1 & & \\ & & 1 & 1 \\ & & 1 & -1 \end{bmatrix}.$$

F_4 contains the powers of $w_4 = i$, the *fourth root* of 1. F_2^* contains the powers of $w_2 = -1$, the *square root* of 1. Note especially that half the entries in F_2^* are zero. The 2 by 2 transform, done twice, requires only half as much work as a direct 4 by 4 transform. If 64 by 64 transform could be replaced by two 32 by 32 transforms, the work would be cut in half (plus the cost of reassembling the results). What makes this true, and possible in practice, is the simple connection between w_{64} and w_{32}:

$$(w_{64})^2 = w_{32}, \qquad \text{or} \qquad \left(e^{2\pi i/64}\right)^2 = e^{2\pi i/32}.$$

The 32nd root is twice as far around the circle as the 64th root. If $w^{64} = 1$, then $(w^2)^{32} = 1$. The mth root is the square of the nth root, if m is half of n:

$$w_n^2 = w_m \qquad \text{if} \qquad m = \tfrac{1}{2}n. \tag{12}$$

The speed of the FFT, in the standard form presented here, depends on working with highly composite numbers like $2^{10} = 1024$. Without the fast transform, it takes $(1024)^2$ multiplications to produce F times c (which we want to do often). By contrast, a fast transform can do each multiplication in only $5 \cdot 1024$ steps. ***It is 200 times faster***, because it replaces one factor of 1024 by 5. In general it replaces n^2 multiplications by $\frac{1}{2}n\ell$, when n is 2^ℓ. By connecting F_n to two copies of $F_{n/2}$, and then to four copies of $F_{n/4}$, and eventually to a very small F, the usual n^2 steps are reduced to $\frac{1}{2}n\log_2 n$.

We need to see how $y = F_n c$ (a vector with n components) can be recovered from two vectors that are only half as long. The first step is to divide c itself, by separating its even-numbered components from its odd-numbered components:

$$c' = (c_0, c_2, \ldots, c_{n-2}) \qquad \text{and} \qquad c'' = (c_1, c_3, \ldots, c_{n-1}).$$

The coefficients just go alternately into c' and c''. From those vectors, the half-size transform gives $y' = F_m c'$ and $y'' = F_m c''$. *Those are the two multiplications by the smaller matrix F_m.* The central problem is to recover y from the half-size vectors y' and y'', and Cooley and Tukey noticed how it could be done:

3W The first m and the last m components of the vector $y = F_n c$ are

$$\begin{aligned} y_j &= y'_j + w_n^j y''_j, \qquad & j &= 0, \ldots, m-1 \\ y_{j+m} &= y'_j - w_n^j y''_j, \qquad & j &= 0, \ldots, m-1. \end{aligned} \tag{13}$$

Thus the three steps are: split c into c' and c'', transform them by F_m into y' and y'', and reconstruct y from equation (13).

We verify in a moment that this gives the correct y. (You may prefer the flow graph to the algebra.) ***This idea can be repeated. We go from F_{1024} to F_{512} to F_{256}.*** The final count is $\frac{1}{2}n\ell$, when starting with the power $n = 2^\ell$ and going all the way to $n = 1$—where no multiplication is needed. This number $\frac{1}{4}n\ell$ satisfies the rule given above: *twice the count for m, plus m extra multiplications, produces the count for n*:

$$2\left(\frac{1}{2}m(\ell-1)\right) + m = \frac{1}{2}n\ell.$$

Another way to count: There are ℓ steps from $n = 2^\ell$ to $n = 1$. Each step needs $n/2$ multiplications by $D_{n/2}$ in equation (13), which is really a factorization of F_n:

$$\textbf{One FFT step} \qquad F_{1024} = \begin{bmatrix} I_{512} & D_{512} \\ I_{512} & -D_{512} \end{bmatrix} \begin{bmatrix} F_{512} & \\ & F_{512} \end{bmatrix} \begin{bmatrix} \textbf{even-odd} \\ \textbf{permutation} \end{bmatrix}. \tag{14}$$

The cost is only slightly more than linear. Fourier analysis has been completely transformed by the FFT. To verify equation (13), split y_j into *even* and *odd*:

$$y_j = \sum_{k=0}^{n-1} w_n^{jk} c_k \quad \text{is identical to} \quad \sum_{k=0}^{m-1} w_n^{2kj} c_{2k} + \sum_{k=0}^{m-1} w_n^{(2k+1)j} c_{2k+1}.$$

Each sum on the right has $m = \frac{1}{2}n$ terms. Since w_n^2 is w_m, the two sums are

$$y_j = \sum_{k=0}^{m-1} w_m^{kj} c'_k + w_n^j \sum_{k=0}^{m-1} w_m^{kj} c''_k = y'_j + w_n^j y''_j. \tag{15}$$

For the second part of equation (13), $j+m$ in place of j produces a sign change:

Inside the sums, $w_m^{k(j+1)}$ remains w_m^{kj} since $w_m^{km} = 1^k = 1$.
Outside, $w_n^{j+m} = -w_n^j$ because $w_n^m = e^{2\pi i m/n} = e^{\pi i} = -1$.

The FFT idea is easily modified to allow other prime factors of n (not only powers of 2). If n itself is a prime, a completely different algorithm is used.

Example 1. The steps from $n = 4$ to $m = 2$ are

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} \to \begin{bmatrix} c_0 \\ c_2 \\ c_1 \\ c_3 \end{bmatrix} \to \begin{bmatrix} F_2 c' \\ \\ F_2 c'' \end{bmatrix} \to \begin{bmatrix} \\ y \\ \\ \end{bmatrix}.$$

Combined, the three steps multiply c by F_4 to give y. Since each step is linear, it must come from a matrix, and the product of those matrices must be F_4:

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & i^2 & i^3 \\ 1 & i^2 & i^4 & i^6 \\ 1 & i^3 & i^6 & i^9 \end{bmatrix} = \begin{bmatrix} 1 & & 1 & \\ & 1 & & i \\ 1 & & -1 & \\ & 1 & & -i \end{bmatrix} \begin{bmatrix} 1 & 1 & & \\ 1 & -1 & & \\ & & 1 & 1 \\ & & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & & 1 & \\ & 1 & & \\ & & & 1 \end{bmatrix}. \tag{16}$$

You recognize the two copies of F_2 in the center. At the right is the permutation matrix that separates c into c' and c''. At the left is the matrix that multiplies by w_n^j. If we started with F_8, the middle matrix would contain two copies of F_4. ***Each of those would be split as above***. Thus the FFT amounts to a giant factorization of the Fourier matrix! The single matrix F with n^2 nonzeros is a product of approximately $\ell = \log_2 n$ matrices (and a permutation) with a total of only $n\ell$ nonzeros.