The Fourier Setup

Earlier we studied Fourier series expansions using basis functions from the collection

$$\{1, \cos x, \sin x, \cos(2x), \sin(2x), \ldots\}.$$

Instead of using this orthogonal basis of $L^2(-\pi, \pi)$, where the basis functions are *real*, we turn to a basis consisting of complex exponential functions. The full basis consists of functions

$$\mathcal{B} \;=\; \{\ldots,\, \phi_{-3}(x),\, \phi_{-2}(x),\, \phi_{-1}(x),\, \phi_0(x),\, \phi_1(x),\, \phi_2(x),\, \ldots\},$$

where each $\phi_k(x) := e^{ikx}$, $k = 0, \pm 1, \pm 2, \ldots$. Those who have seen Euler's Formula should suspect there is a relationship between these two bases. In fact, by Euler's Formula,

$$e^{ikx} \;:=\; \begin{cases} \cos(kx) + i\sin(kx), & k > 0 \\ 1, & k = 0 \\ \cos(-kx) - i\sin(-kx), & k < 0 \end{cases},$$

so, for $k > 0$,

$$\cos(kx) \;=\; \frac{1}{2}\left(e^{ikx} + e^{-ikx}\right) \quad \text{and} \quad \sin(kx) \;=\; \frac{1}{2i}\left(e^{ikx} - e^{-ikx}\right).$$

We will not explore this relationship further, except possibly in homework.

Some claims:

(i) For any given $f \in L^2(0, \pi)$, there exists a function $u \in \operatorname{span}(\mathcal{B})$

$$u \;=\; \sum_{n=-\infty}^{\infty} c_n \phi_n(x)$$

for which $\|f - u\| = 0$ (in the $L^2(0, 2\pi)$ norm). Essentially, u is the same function as f (except possibly on a set of measure zero, a statement whose meaning is explored in MATH 362).

(ii) The ϕ_k are mutually orthogonal in $L^2(0, 2\pi)$. More specifically, given any two integers k, ℓ,

$$\langle \phi_k, \phi_\ell \rangle \;=\; \int_0^{2\pi} \phi_k(x)\overline{\phi_\ell(x)}\, dx \;=\; \begin{cases} 0, & \text{if } k \neq \ell \\ 2\pi, & \text{if } k = \ell \end{cases}$$

(iii) Let S be a *proper* subset (that is, part, but not all) of the integers and $W = \operatorname{span}(\{\phi_k \,|\, k \in S\})$. The elements of W are linear combinations of those ϕ_k included in the basis of W:

$$\sum_{k \in S} c_k \phi_k(t). \tag{1}$$

Given an $f \in L^2(0, 2\pi)$ one cannot expect a $u \in W$—i.e., one of the form (1)— to equal $f(t)$ at all $t \in (0, 2\pi)$. We have two options:

- Seek to minimize the L^2-norm of the difference $(u - f)$. That is, we seek to minimize the difference $(u - f)$ over all x-values in the interval:
$$\|u - f\| \;=\; \left(\int_0^{\pi} |u(x) - f(x)|^2 \, dx \right)^{1/2},$$
which leads to normal equations of the nice variety (due to the orthogonality of the ϕ_k). This is the approach from which we get complex Fourier series expansions of f.
- Pick particular x-values in $[0, 2\pi)$, and *sample* the functions ϕ_k at these x. This is called **discretization**: it substitutes vectors for the functions ϕ_k.

The discrete Fourier transform

We follow the second of the two options listed in bullet points above. For a chosen size n we take our set $S = \{0, 1, \ldots, n-1\}$, which is tantamount to selecting W to have basis $\beta = \{\phi_0, \phi_1, \ldots, \phi_{n-1}\}$. Then, to discretize each of these functions as a vector in $\mathbb{R}^n$, we partition the interval $[0, 2\pi)$ into n evenly-spaced points
$$x_\ell \;=\; \ell \, \frac{2\pi}{n}, \qquad \ell = 0, 1, \ldots, n-1.$$
We sample $\phi_k(x)$ at the x_ℓ-values, obtaining the discretized vector
$$\mathbf{v}_k \;=\; \begin{bmatrix} \phi_k(x_0) \\ \phi_k(x_1) \\ \phi_k(x_2) \\ \vdots \\ \phi_k(x_{n-1}) \end{bmatrix} \;=\; \begin{bmatrix} e^0 \\ (e^{i2\pi/n})^k \\ (e^{i2\pi/n})^{2k} \\ \vdots \\ (e^{i2\pi/n})^{(n-1)k} \end{bmatrix} \;=\; \begin{bmatrix} 1 \\ w^k \\ w^{2k} \\ \vdots \\ w^{(n-1)k} \end{bmatrix},$$
where $w = \exp(i2\pi/n)$ is the principal n^{th} root of unity. Following Strang[1], we call the square matrix $\mathbf{F}$ whose columns are, in order, $\mathbf{v}_0$, $\mathbf{v}_1$, ..., $\mathbf{v}_{n-1}$, the **Fourier matrix**:
$$\mathbf{F}_n \;=\; \begin{bmatrix} 1 & 1 & \cdots & 1 & \cdots & 1 \\ 1 & w & \cdots & w^k & \cdots & w^{n-1} \\ 1 & w^2 & \cdots & w^{2k} & \cdots & w^{2(n-1)} \\ \vdots & \vdots & & \vdots & & \vdots \\ 1 & w^{n-1} & \cdots & w^{(n-1)k} & \cdots & w^{(n-1)^2} \end{bmatrix}.$$

Below is Octave code to create this *symmetric* matrix from input n.

[1] In some texts, the focus is on the complex conjugate $\omega = \overline{w} = \exp(-i2\pi/n)$, also an n^{th} root of unity whose powers generate all others. The matrix, built in the same fashion as above but now from ω, is, in fact the adjoint of the one above: $\mathbf{F}^{\mathrm{H}} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{n-1} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(n-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \cdots & \omega^{(n-1)^2} \end{bmatrix}$.

```
octave:579> function outmat = fourierMat(enn)
>  w = exp(2*i*pi/enn);       # principal  nth root
>  rexp =  [0:(enn−1)];        # yields  row [0 1 2  ...  (n−1)]
>  cexp = rexp';
>  outmat = w.^(cexp * rexp);
> end
```

For reusability, one might want to place this short script in a text file called `fourierMat.m`. Having defined this function, we can look at the 2-by-2 and 3-by-3 Fourier matrices,

```
octave:580> fourierMat(2)
ans =

   1.0000  + 0.0000i    1.0000  + 0.0000i
   1.0000  + 0.0000i   −1.0000 + 0.0000i

octave:581> fourierMat(3)
ans =

   1.00000 + 0.00000i    1.00000 + 0.00000i    1.00000 + 0.00000i
   1.00000 + 0.00000i   −0.50000 + 0.86603i  −0.50000 − 0.86603i
   1.00000 + 0.00000i   −0.50000 − 0.86603i  −0.50000 + 0.86603i
```

which are

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad \text{and} \qquad \begin{bmatrix} 1 & 1 & 1 \\ 1 & \frac{1}{2}(-1+i\sqrt{3}) & \frac{1}{2}(-1-i\sqrt{3}) \\ 1 & \frac{1}{2}(-1-i\sqrt{3}) & \frac{1}{2}(-1+i\sqrt{3}) \end{bmatrix},$$

respectively. Since $\mathbf{F}_n$ has nonreal entries, its tranpose and its adjoint are not the same. We use $n = 4$ to demonstrate that the Fourier matrix is not orthogonal (i.e., $\mathbf{F}_n^{\mathrm{T}}\mathbf{F}_n$ does not equal $\mathbf{I}_n$), but is nearly *unitary*.

```
octave:583> transpose(fourierMat(4))*fourierMat(4)
ans =

   4.00000 + 0.00000i   −0.00000 + 0.00000i    0.00000 + 0.00000i    0.00000 + 0.00000i
  −0.00000 + 0.00000i    0.00000 + 0.00000i    0.00000 + 0.00000i    4.00000 − 0.00000i
   0.00000 + 0.00000i    0.00000 + 0.00000i    4.00000 − 0.00000i   −0.00000 + 0.00000i
   0.00000 + 0.00000i    4.00000 − 0.00000i   −0.00000 + 0.00000i    0.00000 + 0.00000i

octave:584> fourierMat(4)'*fourierMat(4)
ans =

   4.00000 + 0.00000i   −0.00000 + 0.00000i    0.00000 + 0.00000i    0.00000 + 0.00000i
  −0.00000 − 0.00000i    4.00000 + 0.00000i   −0.00000 + 0.00000i    0.00000 + 0.00000i
   0.00000 − 0.00000i   −0.00000 − 0.00000i    4.00000 + 0.00000i   −0.00000 + 0.00000i
   0.00000 − 0.00000i    0.00000 − 0.00000i   −0.00000 + 0.00000i    4.00000 + 0.00000i
```

A square complex matrix $\mathbf{B}$ is called **unitary** if $\mathbf{B}^{\mathrm{H}}\mathbf{B} = \mathbf{I}$ or, equivalently, $\mathbf{B}^{-1} = \mathbf{B}^{\mathrm{H}}$. In the case of the Fourier matrix $\mathbf{F}_n$, we have $\mathbf{F}_n^{\mathrm{H}}\mathbf{F}_n = n\mathbf{I}_n$, so $\mathbf{F}_n^{-1} = \frac{1}{n}\mathbf{F}_n^{\mathrm{H}}$.

Now, $\mathbb{C}^n$ is an n-dimensional vector space. $E = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n\}$ forms a basis (called, as usual, the **standard basis**) of $\mathbb{C}^n$, as does β. One may view $\mathbf{F}_n$ as a change-of-basis matrix $[\mathrm{id}]_\beta^E$, the one from β to E. Of course, $\frac{1}{n}\mathbf{F}_n^{\mathrm{H}}$, its inverse, is the change-of-basis matrix $[\mathrm{id}]_E^\beta$.

It would make most sense to say that the discrete (or **finite**) Fourier transform is what we do when we write ordinary vectors of $\mathbb{R}^n$ in terms of β-coordinates—i.e., $[\mathbf{x}]_\beta$. We know, in general,

$$[\mathbf{x}]_\beta \;=\; [\mathrm{id}]_E^\beta x \;=\; \frac{1}{n}\,\mathbf{F}_n^{\mathrm{H}}\mathbf{x}.$$

Different authors choose to deal with the factor $(1/n)$ in different ways. Many, like Strang, prefer to apply it when converting back to standard coordinates. Thus, we write the DFT of $\mathbf{x}$ as

$$\mathbf{X} \;=\; \mathbf{F}_n^{\mathrm{H}}\mathbf{x}.$$