AMSC/CMSC 661 Scientific Computing II
Spring 2005
Transforms and Wavelets
Dianne P. O'Leary

Transform methods

Recall: We have already encountered the **Fourier transform**

$$\mathcal{F}v(\xi) = \hat{v}(\xi) = \int_{\mathcal{R}^d} v(x) e^{-ix\cdot\xi} dx,$$

and its **inverse**.

We used it to analyze certain **time-dependent PDEs**.

In Homework 4, Part 1, we also encountered the **Discrete Sine transform**

$$F(s) = \alpha_s \sum_{\mathbf{x=1}}^{\mathbf{n}} f(x) \sin(sx\pi/(n+1))$$

and its inverse and used it to solve a matrix problem derived from discretizing an elliptic PDE.

We now return to these transform methods, and introduce some new ones.

The Plan

- Some useful transforms
  - Fourier transforms
  - Discrete Fourier transforms
  - Wavelet transforms
- Properties
- Wavelets
- Applications of transforms

Some useful transforms

- The Fourier transform
- The discrete Fourier transform

- The discrete sine transform
- The Haar transform
- The discrete Haar transform

There are **many, many** other useful transforms that we don't have time to consider (Laplace transforms, Z-transforms, ...), but the tools we develop should make it easier to add new transforms to your toolbox.

Reference: A First Course in Fourier Analysis, by David W. Kammler.

The setup

Given:

- A **domain** Ω
- An **inner product** (u, v) on Ω (using a variable x)
- A set of **basis functions** $z(s, x)$ (usually orthogonal)
- A **function** $f(x)$ of interest

Define the **transform** of f to be

$$F(s) = (f(x), z(s, x))$$

and define the **inverse transform** to be the function that maps $F(s)$ back to $f(x)$.

Example 1a: The Fourier Transform (as before, but $d = 1$)

- Domain: $\Omega = (-\infty, \infty)$
- Inner product:
$$(u, v) = \int_{-\infty}^{\infty} u(x)v(x)dx$$
- Basis functions:
$$z(s, x) = e^{-isx}$$
- Definition of the transform:
$$F(s) = \int_{\mathcal{R}} f(x)e^{-ixs}dx,$$

- Definition of the inverse transform:

$$f(x) = \frac{\mathbf{1}}{(\mathbf{2}\pi)} \int_{\mathcal{R}} F(s) e^{+ixs} ds.$$

The Larsson and Thomée book uses this definition.

Example 1b: The Fourier Transform (an alternate definition)

(This version has the 2π in the exponent, moves the minus sign, and avoids the normalization constant.)

- Domain: $\Omega = (-\infty, \infty)$
- Inner product:

$$(u, v) = \int_{-\infty}^{\infty} u(x) v(x) dx$$

- Basis functions:

$$z(s, x) = e^{2\pi isx}$$

- Definition of the transform:

$$F(s) = \int_{-\infty}^{\infty} f(x) e^{\mathbf{2}\pi isx} dx$$

- Definition of the inverse transform:

$$f(x) = \int_{-\infty}^{\infty} F(s) e^{-\mathbf{2}\pi isx} ds$$

The Kammler book uses this definition.

Example 2: The Discrete Fourier Transform

- Domain: $\Omega = \{0, 1, \ldots, n-1\}$
- Inner product:

$$(u, v) = \sum_{x=0}^{n-1} u(x) v(x)$$

- Basis functions:

$$z(s, x) = e^{2\pi isx/n}$$

- Definition of the transform:

$$F(s) = \sum_{x=0}^{n-1} f(x) e^{2\pi isx/n}$$

- Definition of the inverse transform:

$$f(x) = \sum_{s=0}^{n-1} F(s)e^{-2\pi isx/n}$$

Example 3: The Discrete Sine Transform

- Domain: $\Omega = \{1, \ldots, n\}$
- Inner product:

$$(u, v) = \sum_{x=1}^{n} u(x)v(x)$$

- Basis functions:

$$z(s, x) = \sin(sx\pi/(n+1))$$

- Definition of the transform:

$$F(s) = \alpha_s \sum_{\mathbf{x=1}}^{\mathbf{n}} f(x)\sin(sx\pi/(n+1))$$

  where α_s is a normalization factor.

- Definition of the inverse transform:

$$f(x) = \alpha_x \sum_{\mathbf{s=1}}^{\mathbf{n}} F(s)\sin(sx\pi/(n+1))$$

Example 4: The Haar Transform

- Domain: $\Omega = [0, 1]$
- Inner product:

$$(u, v) = \int_0^1 u(x)v(x)dx$$

- Basis functions:

$$z(s, x) = h(sx)$$

  where

$$h(x) = \begin{cases} 1 & \text{if } 2k \le x < 2k+1 \\ -1 & \text{if } 2k+1 \le x < 2(k+1) \end{cases}$$

  where k is an integer.

- Definition of the transform:

$$F(s) = \alpha_s \int_0^1 f(x)z(s, x)dx$$

  where α_s is a normalization factor.

- Definition of the inverse transform:

$$f(x) = \alpha_x \int_0^1 F(s) z(x, s) ds$$

Example 5: The Discrete Haar Transform

Most of this slide has changed.

- Domain: $\Omega = \{0, 1/n, \ldots, (n-1)/n\}$ where $n = 2^k$
- Inner product:

$$(u, v) = \sum_{j=0}^{n-1} u(j/n) v(j/n)$$

- Basis functions:

$$z(s, x) = h(sx)$$

- Definition of the transform:

$$F(s) = \alpha_s \sum_{j=0}^{n-1} f(j/n) z(s, j/n)$$

  where α_s is a normalization factor.

- Definition of the inverse transform:

$$f(x) = \alpha_x \sum_{k=0}^{n-1} F(k/n) z(x, k/n)$$

Properties of the transforms

Review: Important properties of the Fourier Transform

In the Kammler notation:

- **Fourier inversion formula:** $\mathcal{F}^{-1}(\mathcal{F}v) = v$.
- **Parseval's formula:**

$$\int_{-\infty}^{\infty} f(x) \bar{g}(x) dx = \int_{-\infty}^{\infty} F(s) \bar{G}(s) ds$$

- **A norm relation:** $\|v\| = \|V\|$.

- **A translation relation:** if $w(x) = v(x-y)$ where y is fixed, then $W(s) = e^{-2\pi iys}V(s)$
- **A scaling relation:** if $w(x) = v(ax)$ where $a > 0$ is a fixed scalar, then $W(s) = a^{-1}V(a^{-1}s)$.
- **A convolution relation:** Define

  $$(v * w)(x) = \int_{\mathcal{R}} v(y)w(x-y)dy.$$

  Then $\mathcal{F}(v * w)(s) = V(s)W(s)$.

- **A differentiation formula** that holds as long as v and its derivatives go to zero for large $|x|$:

  $$\mathcal{F}v'(s) = 2\pi isV(s)$$

Another property of the Fourier transform

$$F(s) = \int_{-\infty}^{\infty} f(x)e^{\mathbf{2}\pi isx}dx$$

Sometimes the function $F(s)$ is zero almost **everywhere**.

For example, if f is **periodic**, so that $f(x+p) = f(x)$, then (it can be shown that) $F(s)$ is nonzero only for $s = k/p$ for $k = 0, \pm 1, \pm 2, \ldots$.

In this case, the integral becomes a sum, and we have a representation

$$f(x) = \sum_{k=-\infty}^{\infty} F[k]e^{2\pi kx/p}.$$

The sine transform

In certain cases (for example, when we are considering real functions that are periodic on $[0, 2\pi]$ and 0 at the boundary), we can substitute the sine transform for the Fourier transform without loss of information.

For this restricted class of functions, all of the above properties hold.

Fast computation of the discrete transforms

The discrete transforms that we have listed (Fourier, sine, and Haar) can all be computed very quickly, in much less than the $O(n^2)$ time that seems to be necessary. This gives rise to algorithms like the **Fast Fourier Transform (FFT)** (re)discovered by Cooley and Tukey (1965) but actually due to Gauss (1805).

There are many ways to understand this, but one of them is through **matrix factorizations**.

FFT, a matrix understanding

See www.cs.umd.edu/users/oleary/c460/460matrixhand.pdf

The discrete Haar transform, a matrix understanding

Let

$$\mathbf{H}_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

The Haar transform of $\mathbf{x} \in \mathcal{R}^2$ is $\mathbf{H}_1\mathbf{x}$.

Define

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{H}_{k-1} & \mathbf{H}_{k-1} \\ \mathbf{H}_{k-1} & -\mathbf{H}_{k-1} \end{bmatrix}.$$

The Haar transform of $\mathbf{x} \in \mathcal{R}^{2^k}$ is $\mathbf{H}_k\mathbf{x}$.

Important observation: To compute $\mathbf{H}_k\mathbf{x}$, we need

- $\mathbf{H}_{k-1}\mathbf{x}_1$ and $\mathbf{H}_{k-1}\mathbf{x}_2$ (where $\mathbf{x}_1$ contains the top half of $\mathbf{x}$ and $\mathbf{x}_2$ contains the rest)
- 2^k additions/subtractions.

This reduces the work from $O(n^2)$ to $O(n \log_2 n)$ where $n = 2^k$.

Wavelets

The **Haar transforms** that we considered are examples of **wavelet transforms**.

In wavelet analysis, we decompose a function into its frequency components, just as in Fourier analysis, but then we break the function up into spatial components scaled to the frequency.

Wavelet analysis grew up in several branches of engineering, and Ingrid Daubechies deserves much of the credit for writing a comprehensive theory of the subject.

But because of the checkered history, there is a **lot** of jargon and several competing but equivalent formulations.

Reference: Kammler, Chapter 10.
http://www.amara.com/current/wavelet.html
Local Expert: Prof. John Benedetto, Math Dept.

The mother wavelet

Wavelet people tend to think of the transform as starting with a **mother function** or **analyzing wavelet**; for example,

$$\psi(x) = \begin{cases} 1 & \text{if } 0 \leq x < 1/2 \\ -1 & \text{if } 1/2 \leq x < 1 \\ 0 & \text{otherwise} \end{cases}$$

(Note that this is an ingredient we used in the Haar transform.)

The important properties are:

- $$\int_{-\infty}^{\infty} \psi(x) dx = 0$$
- $\psi(x)$ has **width** 1.

The father wavelet

Then define a **father wavelet** or **scaling function**; for example

$$\phi(x) = \begin{cases} 1 & \text{if } 0 \leq x < 1 \\ 0 & \text{otherwise} \end{cases}$$

Dilations and translations

Then define **dilations** of the mother function: $\psi(x/2), \psi(x/4), \ldots$, and **translations** $\psi(2^m x - k)$.

Notice:

- $\phi(x/2^j)$ has width 2^j.
- **Wide** dilations can be defined in terms of **narrow** ones: for example,

$$\begin{aligned} \phi(x/2) &= \phi(x) + \phi(x-1) \\ \psi(x/2) &= \phi(x) - \phi(x-1) \end{aligned}$$

- The collection of translations of the dilations forms a **basis** for a class of functions defined on $\mathcal{R}$.

Wavelet expansions

Finally, we express a function $f(x)$ in terms of the wavelet basis:

$$f(x) = \sum_{m=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} F[m,k]\psi(2^m x - k).$$

for some coefficients

$$F[m,k] = 2^m \int_{-\infty}^{\infty} f(x)\psi(2^m x - k)dx.$$

Note: In Fourier analysis, the mth coefficient gives information about the function at frequency scale $1/m$. In wavelet analysis, $F[m,k]$ gives information about the function near $k2^{-m}$ at scale 2^{-m}.

- If we are only interested in a **finite interval**, then we truncate the sum in k.
- If we are only interested in approximating to a **finite spatial scale**, then we truncate the sum in m.

Applications of the transforms

We'll consider 6 examples:

- Function approximation
- Analytic solution of PDEs
- Eigendecomposition of certain matrices
- Spectral analysis
- Denoising and filtering
- Data compression

Application 1: Function approximation

We'll illustrate this by wavelet frames.

A **frame** is an approximation to a given function obtained from a single value of m; instead of

$$f(x) = \sum_{m=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} F[m,k]\psi(2^m x - k),$$

we use

$$f_m(x) = \sum_{\mathbf{k=-\infty}}^{\infty} \alpha_m[k]\phi(2^m x - k).$$

The coefficients are defined by taking an average value of f:

$$\alpha_m[k] = 2^m \int_{-\infty}^{\infty} f(x)\phi(2^m x - k)dx = 2^m \int_{k2^{-m}}^{(k+1)2^{-m}} f(x)dx.$$

Notice that for nice functions, $|f(x) - \alpha_m[k]|$ stays small on the interval $[k2^{-m}, (k+1)2^{-m}]$ for m big enough.

The usefulness of frames is partly due to a nice recursion:

$$f_m(x) = f_{m-1}(x) + d_{m-1}(x)$$

where the **detail function** is defined by

$$d_m(x) = \sum_{k=-\infty}^{\infty} \beta_m[k]\psi(2^m x - k)$$

with

$$\begin{aligned} \beta_m[k] &= 2^m \int_{-\infty}^{\infty} f(x)\psi(2^m x - k)dx \\ &= \frac{1}{2}\left(\alpha_{m+1}[2k] - \alpha_{m+1}[2k+1]\right). \end{aligned}$$

Example: Figure 10-6, p.603 in Kammler.

Application 2: Analytic solution of PDEs

We have already seen this:

- The Fourier transform can be used to solve the IVP for **parabolic problems**: we used a Fourier transform in the d x-variables in the equation $u_t - \Delta u_x$ in x to obtain an ODE in t only.
- We could use a similar method on the **wave equation**.

Application 3: Eigendecomposition of certain matrices

The **discrete sine transform** provides the eigenvectors of any symmetric tridiagonal Toeplitz matrix, and from this, we can obtain the eigenvalues and eigenvectors of many related matrices.

Example: Notes on Fast Poisson Solvers.

Application 4: Spectral analysis

Suppose we have taken n observations of some physical phenomenon, for instance, sunspot activity.

The **discrete Fourier Transform (DFT)** is a way to break the vector x of n observations into its **frequency** components, to determine, for instance, whether there is a cycle of 4 years in the observations.

Application 5: Denoising and filtering

In signal and image processing, our data is often contaminated by noise.

For example, we may have **white noise** (data drawn from a normal distribution with mean zero) contaminating our true data.

In a picture, this looks like "snow". In an audio signal, it gives snaps and pops.

Fortunately, it is easy to rid our data of this noise. We take the Fourier transform of the data, set the high order coefficient to zero, and take the inverse Fourier transform. This operation is called **filtering the data**.

Unfortunately, this also removes any part of the true signal that is of high frequency, but this is usually tolerable.

Example: Matlab's `firdemo, nrfiltdemo`.

In a similar way, we can **design** a filter to remove any specified frequency components.

- If we remove low frequency components, this is called **high pass filtering**.
- If we remove high frequency components, this is called **low pass filtering**.

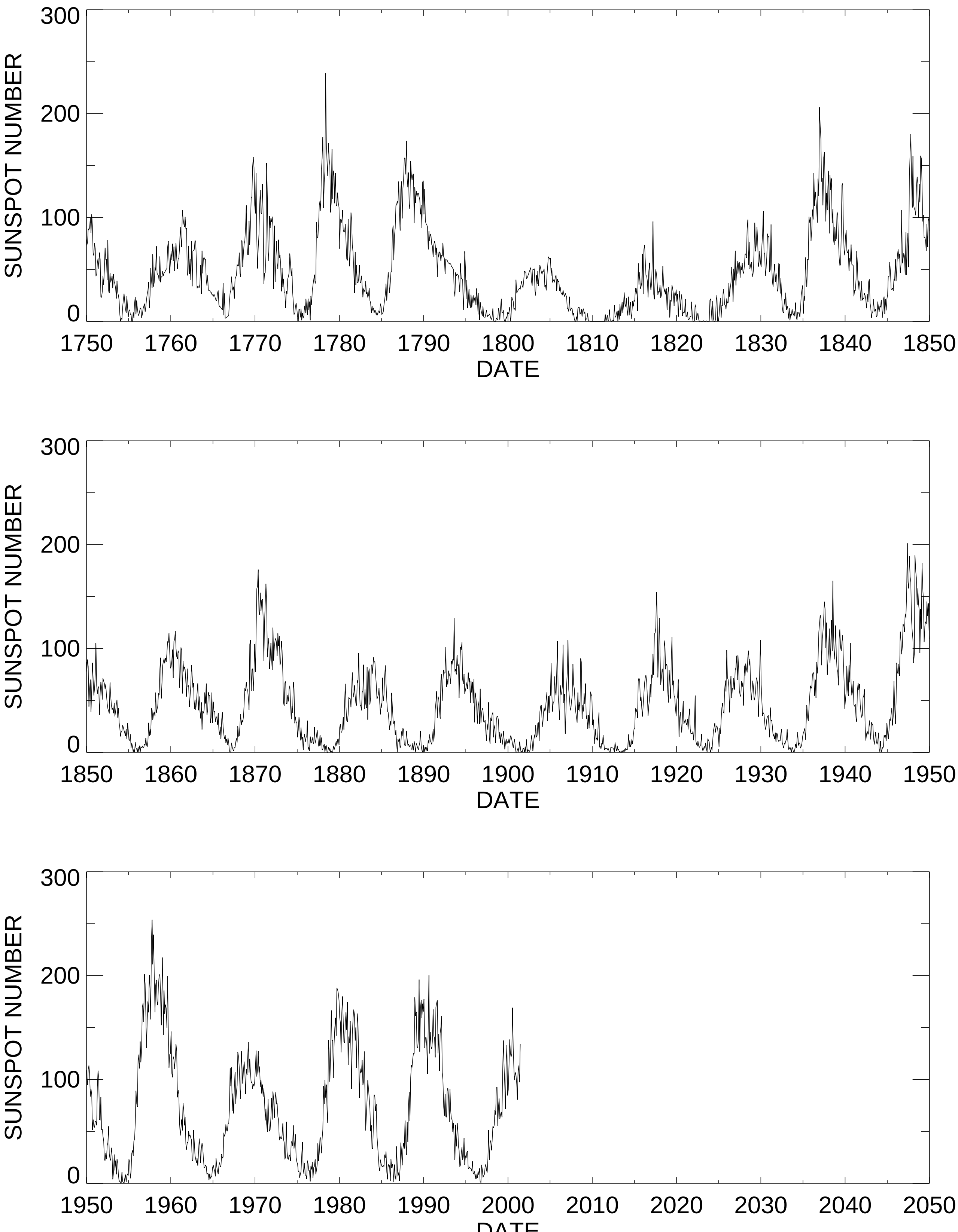

Figure 1. Sunspot activity 1750-2001. Data from

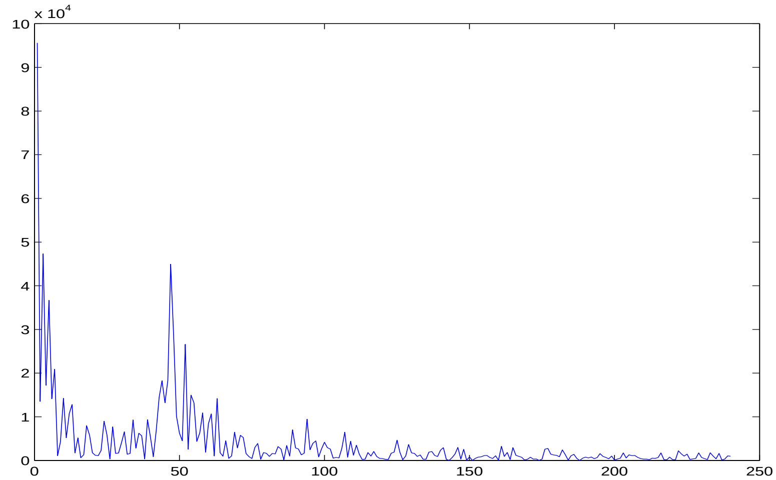

Figure 2: Coefficients from discrete sin transform of sunspot data.

Filter design is an important topic in signal and image processing. Two complicating factors:

- aliasing
- Gibbs phenomenon

Application 6: Data compression

In our previous example, we changed our image by removing its high frequency components. If we chose to store the image in transform coordinates, we could **save space** compared to the original image, since the number of nonzero coefficients is less.

This is a general principle: we can transform a signal, time series, or image, drop the coefficients we don't care about, and store the others. Then when we want to display the data, we take the inverse transform.

This is particularly useful if we need to **transmit** the data before display.

Example: Figure 10.24, p.669 in Kammler.
Matlab's `dctdemo`.

Conclusions:

- A transform is just a change of coordinates.
- There are many transforms, continuous and discrete.
- They are of interest because they are used, for example, to produce closed-form solutions of equations, approximate data, filter data, and compress data.