Motivation for wavelets and some simple examples

Øyvind Ryan

Feb 14, 2017

Google earth type example, Figure 5.1

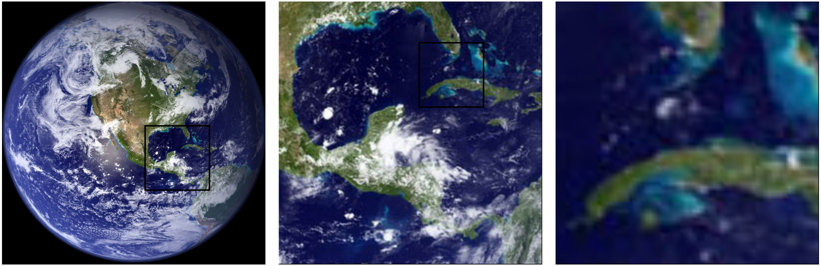


Figure: A view of Earth from space, together with versions of the image where we have zoomed in.

Definition 5.2 (The resolution space V_0): Let N be a natural number. The resolution space V_0 is defined as the space of functions defined on the interval $[0, N)$ that are constant on each subinterval $[n, n+1)$ for $n = 0, \ldots, N-1$.

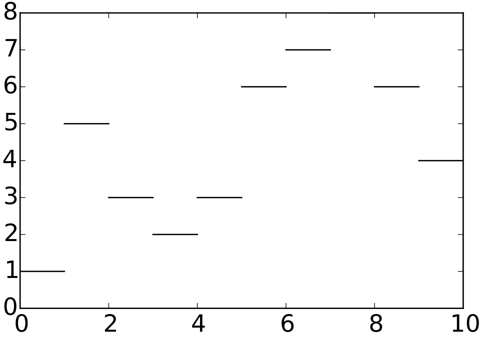


Figure: A piecewise constant function.

The function ϕ, Lemma 5.3

Define the function $\phi(t)$ by

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

and set $\phi_n(t) = \phi(t - n)$ for any integer n. The space V_0 has dimension N, and the N functions $\{\phi_n\}_{n=0}^{N-1}$ form an orthonormal basis for V_0 with respect to the standard inner product

$$\langle f, g \rangle = \int_0^N f(t)g(t)\, dt.$$

In particular, any $f \in V_0$ can be represented as

$$f(t) = \sum_{n=0}^{N-1} c_n \phi_n(t)$$

for suitable coefficients $(c_n)_{n=0}^{N-1}$. The function ϕ_n is referred to as the *characteristic* function of the interval $[n, n+1)$.

Refined resolution spaces, Definition 5.4

The space V_m for the interval $[0, N)$ is the space of piecewise linear functions defined on $[0, N)$ that are constant on each subinterval $[n/2^m, (n+1)/2^m)$ for $n = 0,\ 1,\ \ldots,\ 2^m N - 1$.

Basis for V_m, Lemma 5.5

Let $[0, N)$ be a given interval with N some positive integer. Then the dimension of V_m is $2^m N$. The functions

$$\phi_{m,n}(t) = 2^{m/2}\phi(2^m t - n), \quad \text{for } n = 0, 1, \ldots, 2^m N - 1$$

form an orthonormal basis for V_m, which we will denote by ϕ_m. Any function $f \in V_m$ can thus be represented uniquely as

$$f(t) = \sum_{n=0}^{2^m N - 1} c_{m,n}\phi_{m,n}(t).$$

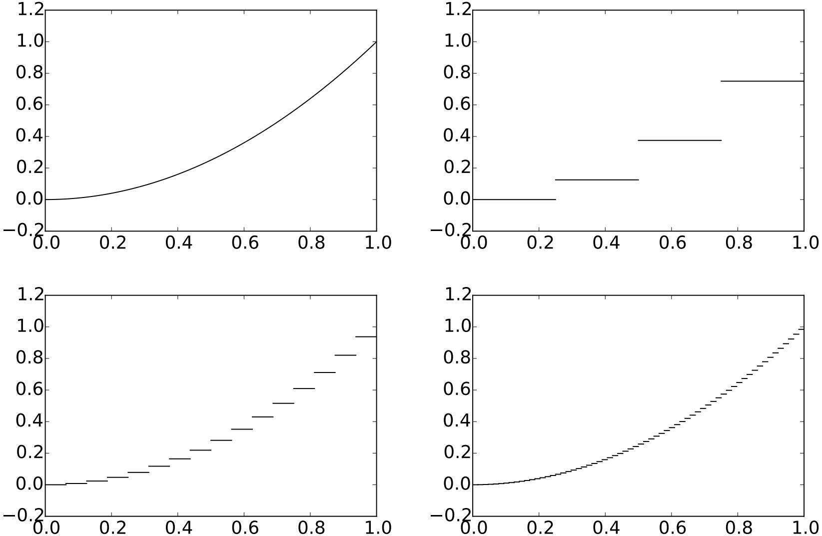
Let f be a given function that is continuous on the interval $[0, N]$. Given $\epsilon > 0$, there exists an integer $m \geq 0$ and a function $g \in V_m$ such that

$$\left|f(t) - g(t)\right| \leq \epsilon$$

for all t in $[0, N]$.

Let f be a given continuous function on the interval $[0, N]$. Then

$$\lim_{m\to\infty} \|f - \mathrm{proj}_{V_m}(f)\| = 0.$$



Resolution spaces are nested, Lemma 5.8

The spaces V_0, V_1, ..., V_m, ... are nested, i.e.

$$V_0 \subset V_1 \subset V_2 \subset \cdots \subset V_m \cdots .$$

Detail spaces Definition 5.9

The orthogonal complement of V_{m-1} in V_m is denoted W_{m-1}. All the spaces W_k are also called detail spaces, or error spaces.

The function ψ, Definition 5.10

We define

$$\psi(t) = \big(\phi_{1,0}(t) - \phi_{1,1}(t)\big)/\sqrt{2} = \phi(2t) - \phi(2t-1),$$

and

$$\psi_{m,n}(t) = 2^{m/2}\psi(2^m t - n), \quad \text{for } n = 0,\, 1,\, \ldots,\, 2^m N - 1.$$

For $0 \leq n < N$ we have that

$$\mathrm{proj}_{V_0}(\phi_{1,n}) = \begin{cases} \phi_{0,n/2}/\sqrt{2}, & \text{if } n \text{ is even;} \\ \phi_{0,(n-1)/2}/\sqrt{2}, & \text{if } n \text{ is odd.} \end{cases}$$

$$\mathrm{proj}_{W_0}(\phi_{1,n}) = \begin{cases} \psi_{0,n/2}/\sqrt{2}, & \text{if } n \text{ is even;} \\ -\psi_{0,(n-1)/2}/\sqrt{2}, & \text{if } n \text{ is odd.} \end{cases}$$

In particular, $\boldsymbol{\psi}_0$ is an orthonormal basis for W_0. More generally, if $g_1 = \sum_{n=0}^{2N-1} c_{1,n}\phi_{1,n} \in V_1$, then

$$\mathrm{proj}_{V_0}(g_1) = \sum_{n=0}^{N-1} c_{0,n}\phi_{0,n}, \text{ where } c_{0,n} = \frac{c_{1,2n} + c_{1,2n+1}}{\sqrt{2}}$$

$$\mathrm{proj}_{W_0}(g_1) = \sum_{n=0}^{N-1} w_{0,n}\psi_{0,n}, \text{ where } w_{0,n} = \frac{c_{1,2n} - c_{1,2n+1}}{\sqrt{2}}.$$

Projections, Proposition 5.12

Let $f(t) \in V_1$, and let $f_{n,1}$ be the value f attains on $[n, n+1/2)$, and $f_{n,2}$ the value f attains on $[n+1/2, n+1)$. Then $\text{proj}_{V_0}(f)$ is the function in V_0 which equals $(f_{n,1} + f_{n,2})/2$ on the interval $[n, n+1)$. Moreover, $\text{proj}_{W_0}(f)$ is the function in W_0 which is $(f_{n,1} - f_{n,2})/2$ on $[n, n+1/2)$, and $-(f_{n,1} - f_{n,2})/2$ on $[n+1/2, n+1)$.

In the same way as in Lemma 5.11, it is possible to show that

$$\mathrm{proj}_{W_{m-1}}(\phi_{m,n}) = \begin{cases} \psi_{m-1,n/2}/\sqrt{2}, & \text{if } n \text{ is even;} \\ -\psi_{m-1,(n-1)/2}/\sqrt{2}, & \text{if } n \text{ is odd.} \end{cases}$$

From this it follows as before that $\boldsymbol{\psi}_m$ is an orthonormal basis for W_m. If $\{\mathcal{B}_i\}_{i=1}^n$ are mutually independent bases, we will in the following write $(\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_n)$ for the basis where the basis vectors from $\mathcal{B}_i$ are included before $\mathcal{B}_j$ when $i < j$. With this notation, the decomposition in Equation (5.7) can be restated as follows

Theorem 5.13 (Bases for V_m): $\boldsymbol{\phi}_m$ and $(\boldsymbol{\phi}_0, \boldsymbol{\psi}_0, \boldsymbol{\psi}_1, \cdots, \boldsymbol{\psi}_{m-1})$ are both bases for V_m.

Vanishing moment, Observation 5.14

We have that $\int_0^N \psi(t)dt = 0$.

The DWT (Discrete Wavelet Transform) is defined as the change of coordinates from ϕ_1 to (ϕ_0, ψ_0). More generally, the m-level DWT is defined as the change of coordinates from ϕ_m to $(\phi_0, \psi_0, \psi_1, \cdots, \psi_{m-1})$. In an m-level DWT, the change of coordinates from

$$(\phi_{m-k+1}, \psi_{m-k+1}, \psi_{m-k+2}, \cdots, \psi_{m-1})$$

to

$$(\phi_{m-k}, \psi_{m-k}, \psi_{m-k+1}, \cdots, \psi_{m-1})$$

is also called the k'th stage. The (m-level) IDWT (Inverse Discrete Wavelet Transform) is defined as the change of coordinates the opposite way.

If $g_m = g_{m-1} + e_{m-1}$ with

$$g_m = \sum_{n=0}^{2^m N-1} c_{m,n}\phi_{m,n} \in V_m,$$

$$g_{m-1} = \sum_{n=0}^{2^{m-1}N-1} c_{m-1,n}\phi_{m-1,n} \in V_{m-1}$$

$$e_{m-1} = \sum_{n=0}^{2^{m-1}N-1} w_{m-1,n}\psi_{m-1,n} \in W_{m-1},$$

then the change of coordinates from ϕ_m to (ϕ_{m-1}, ψ_{m-1}) (i.e. first stage in a DWT) is given by

$$\begin{pmatrix} c_{m-1,n} \\ w_{m-1,n} \end{pmatrix} = \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{pmatrix} \begin{pmatrix} c_{m,2n} \\ c_{m,2n+1} \end{pmatrix}$$

Expression for the IDWT

Conversely, the change of coordinates from (ϕ_{m-1}, ψ_{m-1}) to ϕ_m (i.e. the last stage in an IDWT) is given by

$$\begin{pmatrix} c_{m,2n} \\ c_{m,2n+1} \end{pmatrix} = \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{pmatrix} \begin{pmatrix} c_{m-1,n} \\ w_{m-1,n} \end{pmatrix}$$

Reordering of basis

If we had defined

$$\mathcal{C}_m = \{\phi_{m-1,0}, \psi_{m-1,0}, \phi_{m-1,1}, \psi_{m-1,1}, \cdots,$$
$$\phi_{m-1,2^{m-1}N-1}, \psi_{m-1,2^{m-1}N-1}\}.$$

i.e. we have reordered the basis vectors in (ϕ_{m-1}, ψ_{m-1}) (the subscript m is used since $\mathcal{C}_m$ is a basis for V_m), we have that $G = P_{\phi_m \leftarrow \mathcal{C}_m}$ is the matrix where

$$\begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$$

is repeated along the main diagonal $2^{m-1}N$ times. Also, $H = P_{\mathcal{C}_m \leftarrow \phi_m}$ is the same matrix. Such matrices are called *block diagonal matrices*. This particular block diagonal matrix is clearly orthogonal.

DWT and IDWT kernel transformations, Definition 5.17

The matrices $H = P_{\mathcal{C}_m \leftarrow \phi_m}$ and $G = P_{\phi_m \leftarrow \mathcal{C}_m}$ are called the *DWT and IDWT kernel transformations*. The DWT and the IDWT can be expressed in terms of these kernel transformations by

$$
\begin{aligned}
\text{DWT} &= P_{(\phi_{m-1},\psi_{m-1}) \leftarrow \mathcal{C}_m} H \\
\text{IDWT} &= G P_{\mathcal{C}_m \leftarrow (\phi_{m-1},\psi_{m-1})},
\end{aligned}
$$

respectively, where

- $P_{(\phi_{m-1},\psi_{m-1}) \leftarrow \mathcal{C}_m}$ is a permutation matrix which groups the even elements first, then the odd elements,
- $P_{\mathcal{C}_m \leftarrow (\phi_{m-1},\psi_{m-1})}$ is a permutation matrix which places the first half at the even indices, the last half at the odd indices.

Illustration of the wavelet transform

$$
\begin{tikzcd}
\boldsymbol{\phi}_m \arrow[r] \arrow[dr] & \boldsymbol{\phi}_{m-1} \arrow[r] \arrow[dr] & \boldsymbol{\phi}_{m-2} \arrow[r] \arrow[dr] & \cdots \arrow[r] & \boldsymbol{\phi}_1 \arrow[r] \arrow[dr] & \boldsymbol{\phi}_0 \\
 & \boldsymbol{\psi}_{m-1} & \boldsymbol{\psi}_{m-2} & \boldsymbol{\psi}_{m-3} & & \boldsymbol{\psi}_0
\end{tikzcd}
$$

Figure: Illustration of a wavelet transform.

Kernel transformation for the Haar wavelet, Matlab version

We will use a DWT kernel function which takes as input the coordinates $(c_{m,0}, c_{m,1}, \ldots)$, and returns the coordinates $(c_{m-1,0}, w_{m-1,0}, c_{m-1,1}, w_{m-1,1}, \ldots)$, i.e. computes one stage of the DWT. This is a different order for the coordinates than that given by the basis (ϕ_m, ψ_m). The reason is that it is easier with this new order to compute the DWT in-place. We assume for simplicity that N is even:

```
function x = dwt_kernel_haar(x, bd_mode)
    x = x/sqrt(2);
    N = size(x, 1);
    for k = 1:2:(N-1)
        x(k:(k+1), :) = [x(k, :) + x(k+1, :); x(k, :) - x(k+1,
    end
```

Kernel transformation for the Haar wavelet, Python version

```
def dwt_kernel_haar(x, bd_mode):
    x /= sqrt(2)
    for k in range(2,len(x) - 1,2):
        a, b = x[k] + x[k+1], x[k] - x[k+1]
        x[k], x[k+1] = a, b
```

Remarks

- The code above accepts two-dimensional data. Thus, the function may be applied simultaneously to all channels in a sound, as the FFT.
- The mysterious parameters `bd_mode` and `dual` will be explained later in Chapter 6.
- When N is even, `idwt_kernel_haar` can be implemented with the exact same code.
- The reason for using a general kernel function will be apparent later, when we change to different types of wavelets.

It is not meant that you call this kernel function directly. Instead every time you apply the DWT call the function

```
DWTImpl(x, m, wave_name, bd_mode, dual)
```

- `x` is the input to the DWT
- `m` is the number of levels.
- `wave_name` is a name identifying the wavelet. A function called `find_kernel` maps this name to a kernel function (`find_kernel` maps `haar` to the kernel function `dwt_kernel_haar`).

General DWT implementation, Matlab version

The kernel function is then used as input to the following function:

```
function x=DWTImpl_internal(x, m, dwt_kernel, bd_mode)
    for res=0:(m - 1)
        x(1:2^res:end, :) = dwt_kernel(x(1:2^res:end, :), bd_mo
    end
    x = reorganize_coeffs_forward(x, m);
end
```

- The kernel function is invoked one time for each resolution.
- The function `reorganize_coeffs_forward` reorders the coordinates (i.e. makes the coordinate change between $\mathcal{C}_m$ and (ϕ_m, ψ_m).

General DWT implementation, Python version

```
def DWTImpl_internal(x, m, f, bd_mode):
    for res in range(m):
        f(x[0::2**res], bd_mode)
    reorganize_coeffs_forward(x, m)
```

General IDWT implementation

```
IDWTImpl(x, m, wave_name, bd_mode, dual)
```

```
function x=IDWTImpl_internal(x, m, f, bd_mode)
    x = reorganize_coeffs_reverse(x, m);
    for res = (m - 1):(-1):0
        x(1:2^res:end, :) = f(x(1:2^res:end, :), bd_mode);
    end
end
```

Example 5.10, plotting a sound and its DWT

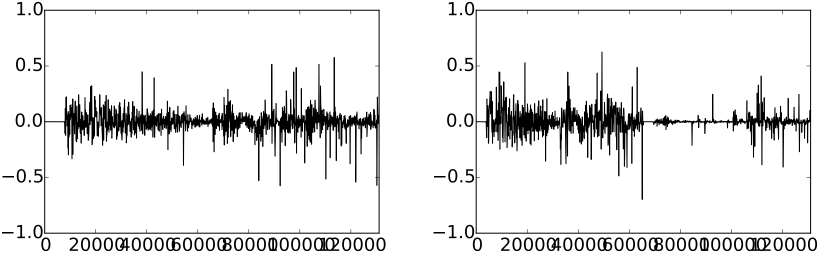


Figure: The 2^{17} first sound samples (left) and the DWT coefficients (right) of the sound `castanets.wav`.

Example 5.10, plotting the error

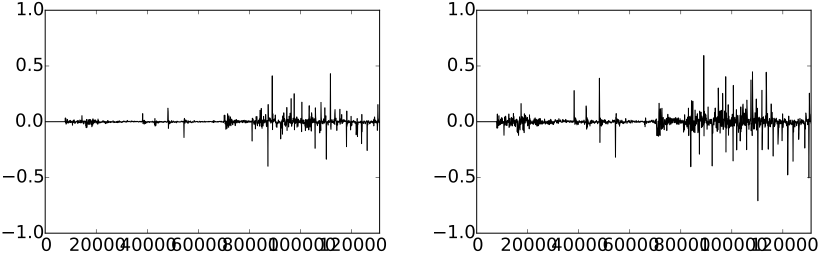


Figure: The error (i.e. the contribution from $W_0 \oplus W_1 \oplus \cdots \oplus W_{m-1}$) in the sound file `castanets.wav`, for $m = 1$ and $m = 2$, respectively.

Example 5.11

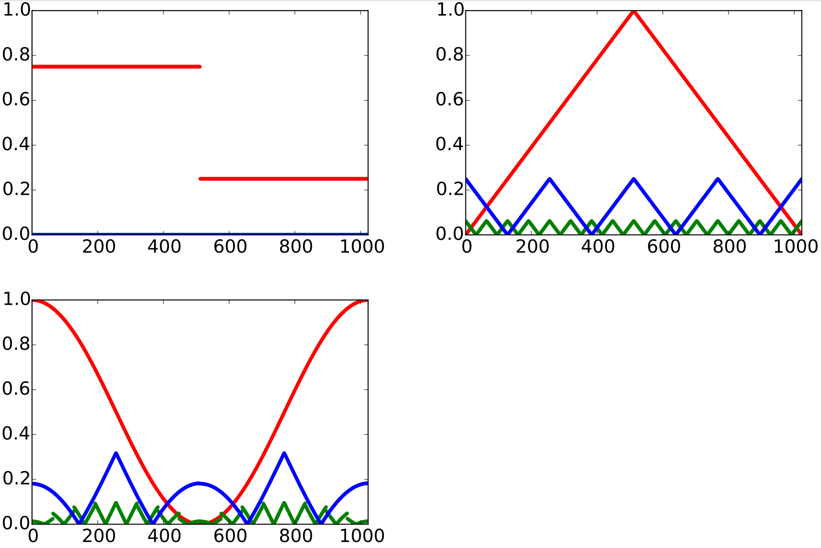


Figure: The error (i.e. the contribution from $W_0 \oplus W_1 \oplus \cdots \oplus W_{m-1}$) for $N = 1024$ when f is a square wave, the linear function $f(t) = 1 - 2|1/2 - t/N|$, and $f(t) = 1/2 + \cos(2\pi t/N)/2$, respectively.