Fourier Series

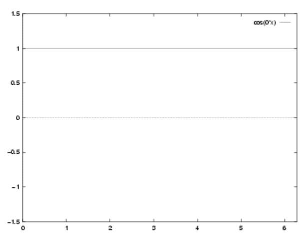
Suppose $f : [0, 2\pi] \to \mathbb{C}$ and:

$$f(x) = \sum_{n=0} a_n \cos nx$$

How could we find the a_n if we know f ?

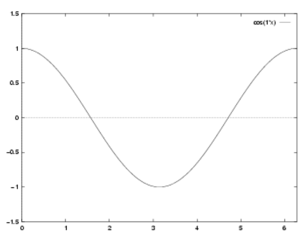
Having a look at cos :

$\cos(0x)$



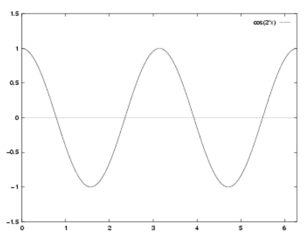
Average 1

$\cos(1x)$



Average 0

$\cos(2x)$



Average 0

$$\int_0^{2\pi} f(x)\,dx = \sum_{n=0} a_n \int_0^{2\pi} \cos nx\,dx = 2\pi a_0$$

How do we find the rest of the a_n ?

$\cos nx \cos mx = 1/2(\cos(m+n)x + \cos(m-n)x)$

$$\begin{aligned}
&\int_0^{2\pi} f(x)\cos mx \, dx \\
&= \sum_{n=0} a_n \int_0^{2\pi} \cos nx \cos mx \, dx \\
&= \pi a_m
\end{aligned}$$

We can do a complex version. If:

$$f(x) = \sum_{n=-\infty}^{\infty} a_n e^{inx}$$

$$\Rightarrow \int_0^{2\pi} e^{-inx} f(x)\, dx = 2\pi a_n$$

And an integral version. If:

$$f(x) = \int_{-\infty}^{\infty} a(\omega) e^{i\omega x}\, d\omega$$

$$\Rightarrow \int_{-\infty}^{\infty} e^{-i\omega x} f(x)\, dx = 2\pi a(\omega)$$

So what functions can we write as $\sum a_n e^{inx}$?

L^p and friends

The sets of functions people look at:

$$L^2([0,2\pi]) = \left\{ f : [0,2\pi] \to \mathbb{C} | \int_0^{2\pi} |f(x)|^2 \, dx < \infty \right\}$$

$$L^1(\mathbb{R}) = \left\{ f : \mathbb{R} \to \mathbb{C} | \int_{-\infty}^{\infty} |f(x)| \, dx < \infty \right\}$$

$$L^3(\mathbb{N}) = \left\{ a_n | n \in \mathbb{N}, \sum_{n=0}^{\infty} |a_n|^3 < \infty \right\}$$

$\{e^{inx} : n \in \mathbb{Z}\}$ are a basis for $L^2([0,2\pi])$.
$\{e^{i\omega x} : \omega \in \mathbb{R}\}$ are a bit like a basis for $L^2(\mathbb{R})$.

So by adding up our e^{inx}'s we get quite a lot of functions.

This seems a bit pointless.

If $v(x) = e^{i\omega x}$ then :

$$\frac{d}{dx}v = \frac{d}{dx}\left(e^{i\omega x}\right) = i\omega e^{i\omega x} = i\omega v$$

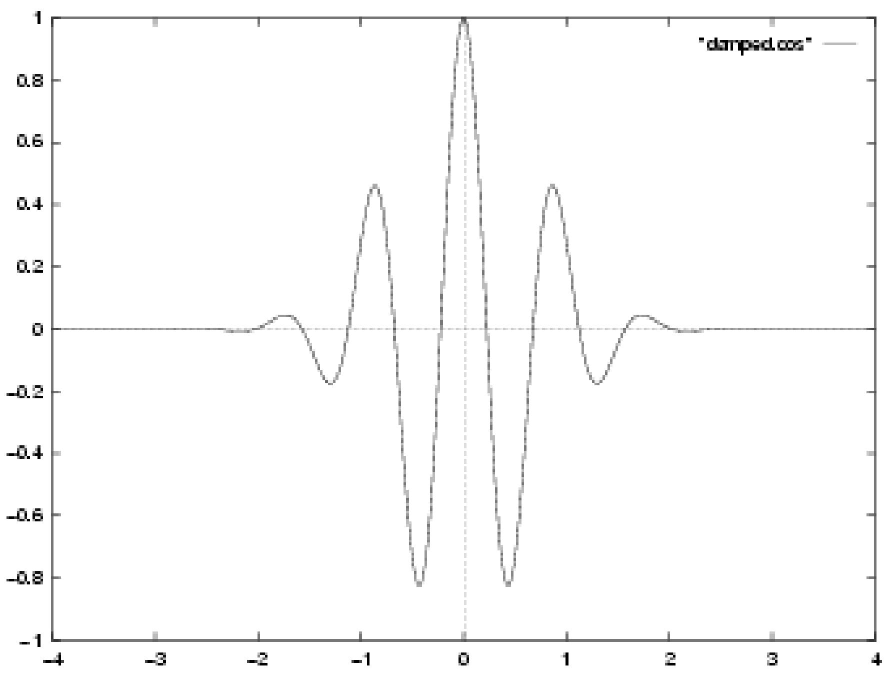
So with Fourier analysis we can write things is terms of eigenvectors for differentiation.

Good for solving stuff like the wave equation:

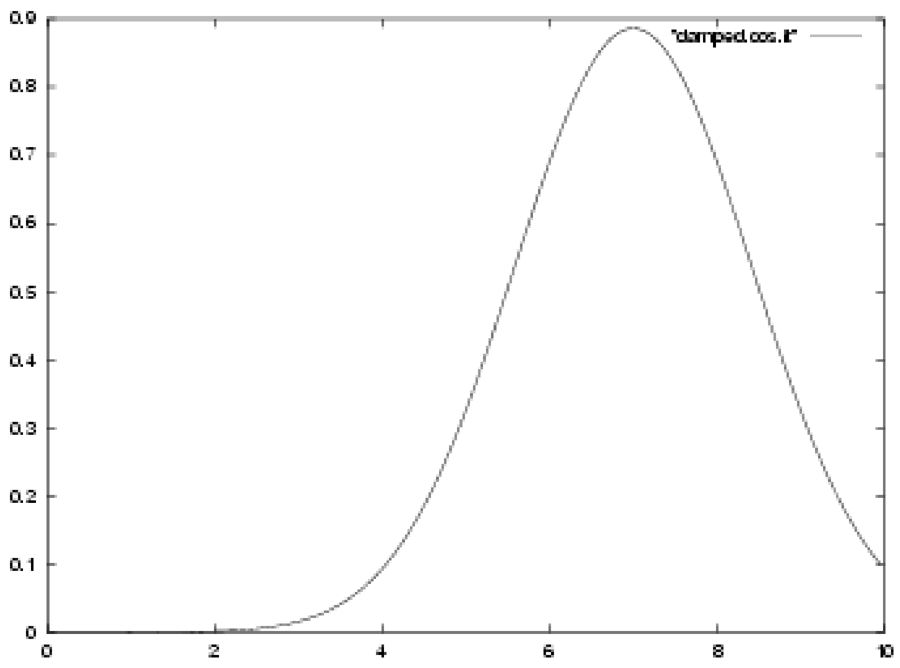
$$\frac{\partial^2}{\partial x^2} - \frac{1}{v^2}\frac{\partial^2}{\partial t^2} = 0$$

And so for signal processing. It also tells us about the frequencies present in a signal.

$e^{-x^2}\cos(7x)$:



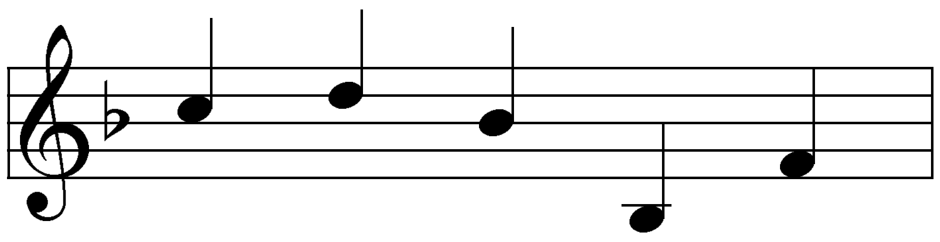
$\mathcal{F}(e^{-x^2}\cos(7x))$



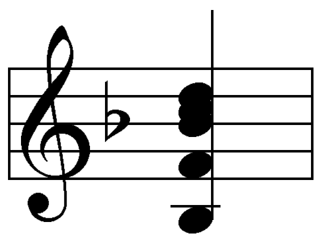
The catch

The Fourier transform tells us nothing about when the frequencies occur. It only tells us how much they occur on the whole.

This means at a glance we can't tell the difference between:

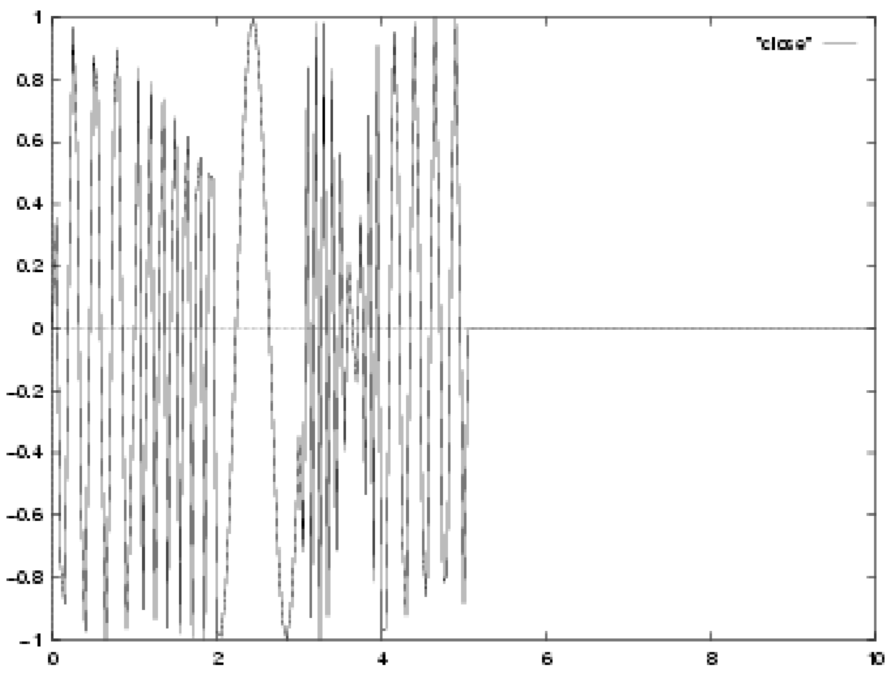


and:

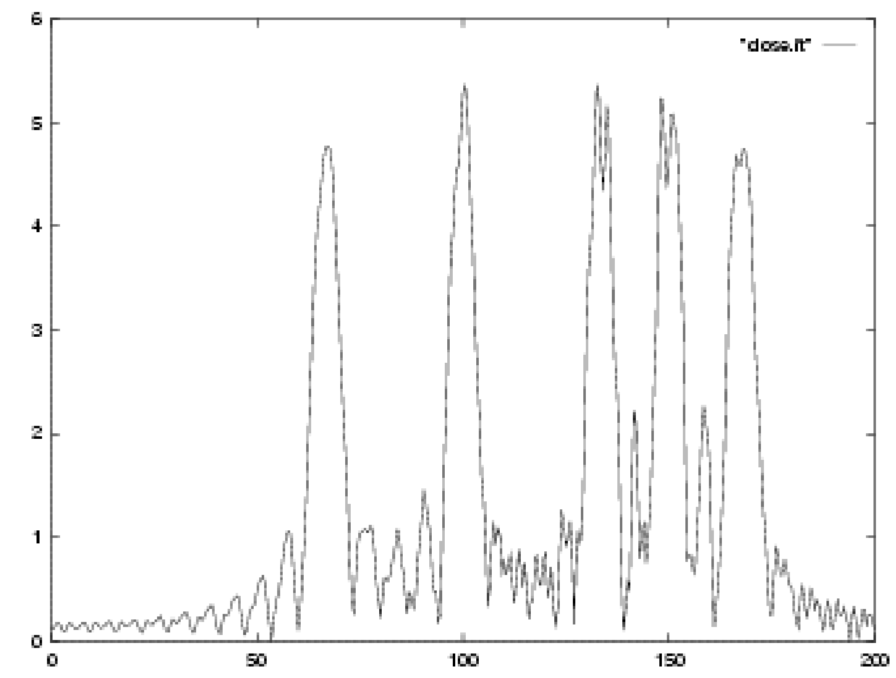


($\alpha^7 = 1.498$, $\alpha^9 = 1.681$, $\alpha^5 = 1.334$, $\alpha^{-7} = 0.667$, $\alpha^0 = 1$)

Close Encounter:



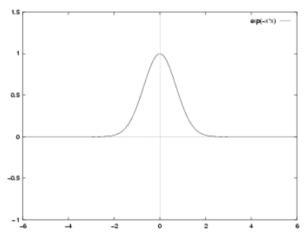
$\mathcal{F}$(Close Encounter):



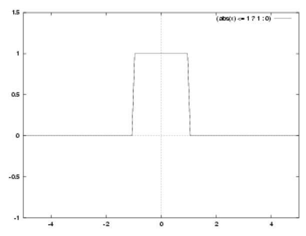
Windowed Fourier Transforms

Cut your signal into little bits, and look at what frequencies they have. You use a 'Window' to do the cutting.

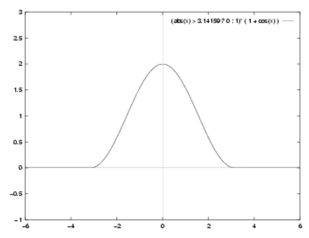
e^{-x^2}



$\chi_{[-1,1]}$

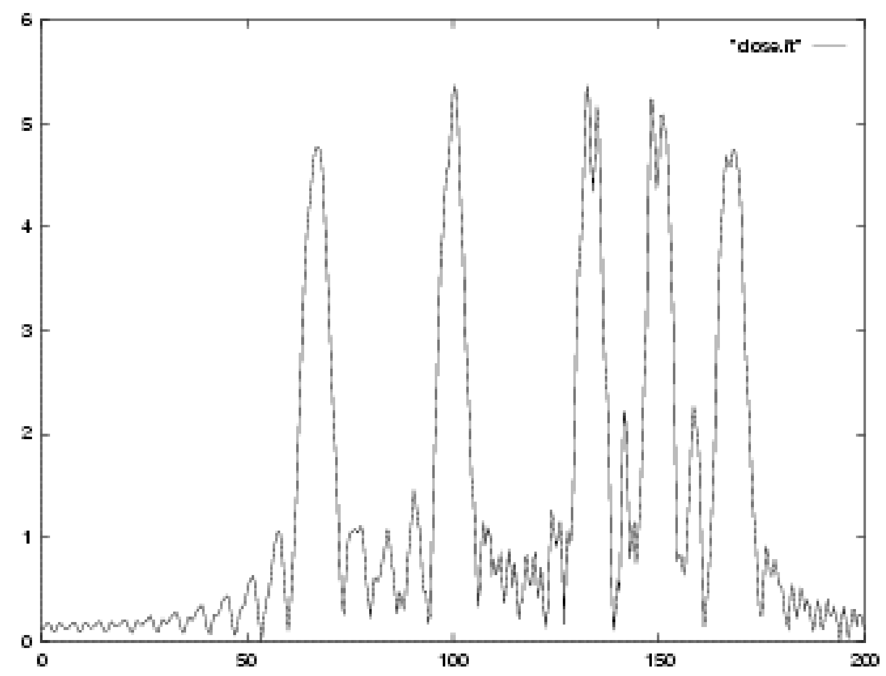


$1 + \cos x$

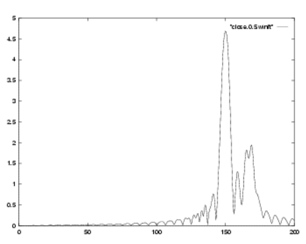


So you center your window over the bits you're interested in, multiply and take the Fourier transform.

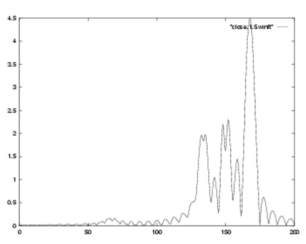
$\mathcal{F}$(Close Encounter):



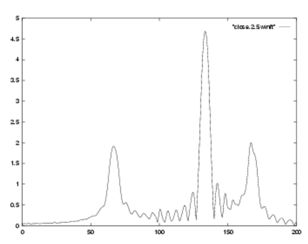
0.5



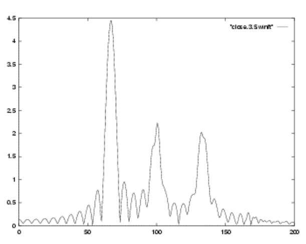
1.5



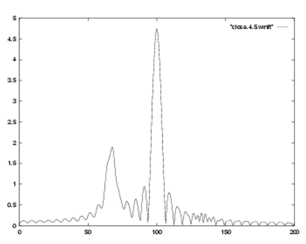
2.5



3.5



4.5



Why Wavelets ?

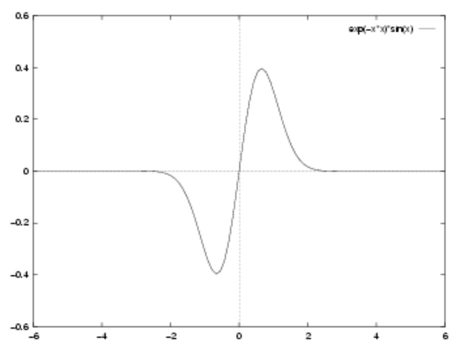
Well you have to choose how wide your window is. If you don't know in advance you're in trouble.

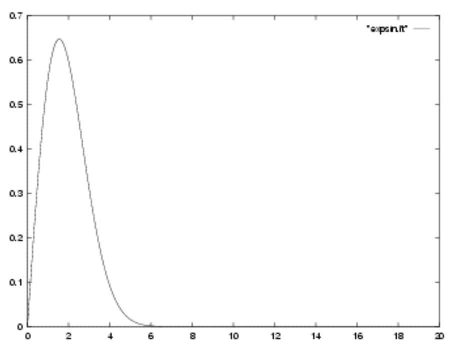
Also if the frequency you're interested in has period longer than your window you're in trouble!

With wavelets you link your window size to the frequency you are looking for. We can take a window (e^{-x^2}) and a wave ($\sin x$) and glue them together.

$$\psi(x) = e^{-x^2} \sin x$$

ψ $\mathcal{F}(\psi)$





If you're looking for frequency μ you scale :

$$\psi(\mu x) = e^{-(\mu x)^2} \sin \mu x$$

And if you want to look at a certain position x_0 you slide :

$$\psi(\mu(x - x_0)) = e^{-(\mu(x-x_0))^2} \sin \mu(x - x_0)$$

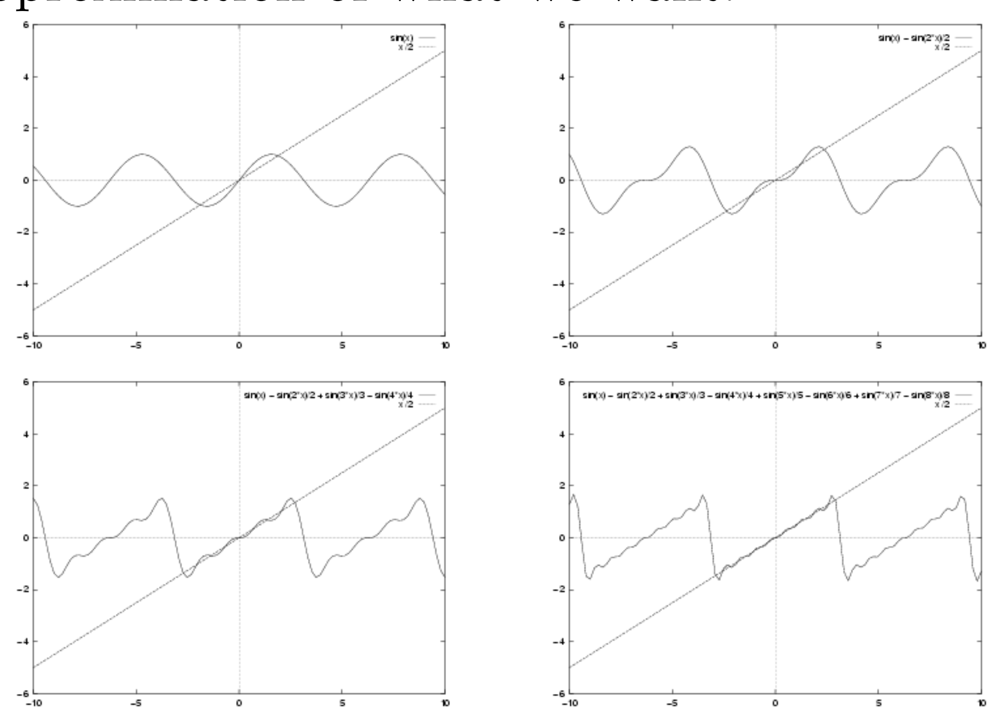
This approach often works, and is responsible for much of the industry related to wavelets.

There is another way to make wavelets....

...using multi-resolution analysis.

Multi-Resolution Approximation

With traditional Fourier series, we can chop our sum and hopefully get a good approximation of what we want.



With multi-resolution approximation we want to get something 'twice' as good as the last at each level.

To approximate something:

1. take a function ($g(x)$),
2. move it to your 'nodes' ($g(x-n)$),
3. multiply ($a_n g(x-n)$),
4. and add.

$$V_0 = \{\sum a_n g(x-n) : n \in \mathbb{Z}\}$$

To improve your resolution, move your nodes twice as close together:

$$V_{j+1} = \{f(2x) : f \in V_j\}$$

And the next level should be at least as good as the last:

$$V_j \subset V_{j+1}$$

Our approximations are like ‘averages’ over length $\frac{1}{2^j}$. At each stage we add on the local detail at the new scale $\frac{1}{2^{j+1}}$.

$$\begin{aligned} f_{j+1}(x) &= f_j(x) + d_j(x) \\ V_{j+1} &= V_j \oplus W_j \end{aligned}$$

We know that the V_j were related by $V_{j+1} = \{f(2x) : f \in V_j\}$ so we make the W_j be related by the same.

We also know that V_0 is the span of the $g(x-n)$, so we hope W_0 is spanned by some $w(x-n)$.

This $w(x)$ will be out wavelet!

So how do we find suitable g and w ?

Dilation Equations

Remember:

$$g \in V_0 \subset V_1$$

but V_1 is the span of $g(2x-k)$, so:

$$g(x) = \sum_k c_k g(2x-k)$$

Suppose we have 4 c_k's. Then $g(x)$ is:

$c_0 g(2x) + c_1 g(2x-1) + c_2 g(2x-2) + c_3 g(2x-3)$

Try $w(x)$:

$c_3 g(2x) - c_2 g(2x-1) + c_1 g(2x-2) - c_0 g(2x-3)$

This $w(x)$ is in V_1, but is it in V_0 ? Check if $w(x)$ was orthogonal to the $g(x-n)$ and get some conditions on c_k.

1. Integrating the dilation equation gives:

$$\begin{aligned}\int g(x)\,dx &= \int \sum c_k g(2x-k)\,dx \\ &= \sum c_k \frac{1}{2} \int g(x)\,dx\end{aligned}$$

   So $\sum c_k = 2$.

2. Wanting the $g(x-n)$ to be orthonormal means $\sum c_k c_{k-2m} = \delta_{0m}$ for any m.

3. Wanting the wavelet to be orthogonal to the $g(x-n)$ produces another condition.

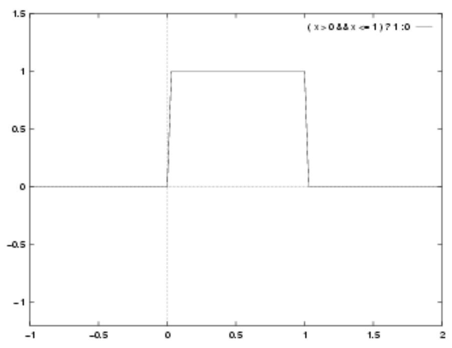
4. If $\sum (-1)^k k^m c_k = 0$ for $m = 0, 1, \ldots p-1$ gives some very interesting info:
   - $1, x, x^2, \ldots x^{p-1}$ are in your space.
   - Error $\approx O(\frac{1}{2^{pj}})$ in V_j.

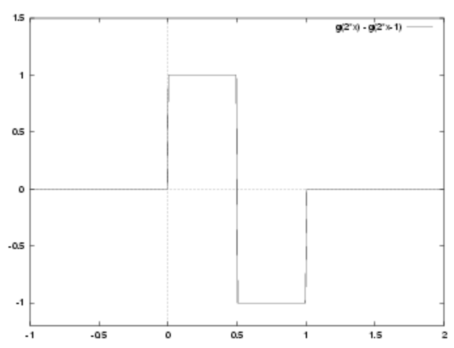
5. Other properties mean other conditions.

Famous values of c_k

Haar Wavelets have $c_0 = 1$, $c_1 = 1$:

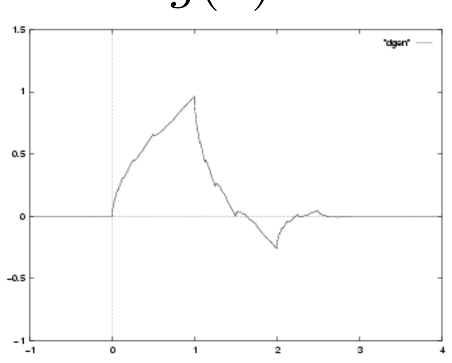
$g(x)$ $w(x)$

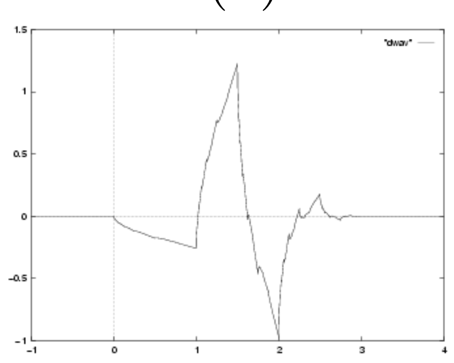




Daubechies Wavelets have $c_0 = \frac{1}{4}(1+\sqrt{3})$,
$c_1 = \frac{1}{4}(3+\sqrt{3})$, $c_2 = \frac{1}{4}(3-\sqrt{3})$, $c_3 = \frac{1}{4}(1-\sqrt{3})$:

$g(x)$ $w(x)$





How to draw these beasties.

Given $c_0, c_1, \ldots, c_N$:

1. Assume g(x) is zero outside $(0, N+1)$.
2. Try to find $g(1), \ldots g(N)$:

$$\begin{aligned} g(1) &= c_1 g(1) + c_0 g(2) \\ g(2) &= c_3 g(1) + c_2 g(2) + c_1 g(3) + c_0 g(1) \\ &\vdots \end{aligned}$$

   This is a matter of finding the eigenvector $(g(1), \ldots, g(N))$.

3. Recursively use $g(x) = \sum_k c_k g(2x-k)$ for values at half values what you already have.

These aren't the only solutions - check $\log(|1 - 1/x|)$ in the Haar equation.

Trying to find a formula.

Taking the Fourier transform of the dilation equation:

$$\begin{aligned} \mathcal{F}(g(x)) &= \mathcal{F}(\sum c_k g(2x-k)) \\ \hat{g}(\omega) &= \sum c_k e^{\frac{-ik\omega}{2}} \hat{g}(\frac{\omega}{2}) \end{aligned}$$

Let $P(\omega) = \sum c_k e^{\frac{-ik\omega}{2}}$, then:

$$\begin{aligned} \hat{g}(\omega) &= P(\omega)\hat{g}(\frac{\omega}{2}) \\ &= P(\omega)P(\frac{\omega}{2})\hat{g}(\frac{\omega}{4}) \\ &= \left(\prod_{j=0}^{n} P(\frac{\omega}{2^j})\right) \hat{g}\left(\frac{\omega}{2^{n+1}}\right) \\ &= \left(\prod_{j=0}^{\infty} P(\frac{\omega}{2^j})\right) \hat{g}(0) \end{aligned}$$

So if you can un-transform that - you're away.

And back again.

If $g = \chi_{[0,1]}$ (the g for Haar wavelets), then define $\mathcal{G}$ by :

1. $\mathcal{G}(g(x)) = \frac{1-e^{-i\omega}}{i\omega}$
2. $\mathcal{G}(g(x-n)) = e^{-i\omega n}\mathcal{G}(g(x))$.
3. $\mathcal{G}$ is linear.
4. $\mathcal{G}(f(2x)) = \frac{1}{2}\mathcal{G}(f)(\frac{\omega}{2})$

This defines $\mathcal{G}$ on V_j. It is in fact the Fourier transform.