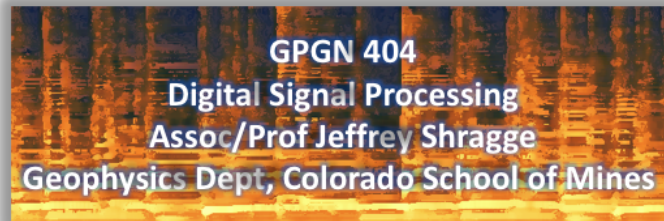


Out[18]: The raw code for this Jupyter notebook is by default hidden for easier reading. To toggle on/off the raw code, click [here](#).



Module 7: Linear Time-Invariant (LTI) Systems

[Linear time-invariant](https://en.wikipedia.org/wiki/Linear_time-invariant_theory) (LTI) systems are important in many branches of science and engineering (including geophysics!) because they can be completely characterized by the [impulse response](https://en.wikipedia.org/wiki/Impulse_response), which is how a system responds to an impulsive $\delta(t)$ function input. One example of a linear system with an impulse response would be if you gave an impulsive push to a friend who is sitting on a swing. What would the response of the system be? Damped oscillation back and forth until eventually coming to rest (where the duration of oscillation depends on the swing's effective damping constant). This time series effectively represents the impulse response of the swing system.

Examples of LTI systems include:

- Spring-mass systems
- Electrical circuits
- 1D Seismic/GPR convolutional modeling

In fact, the [convolution](https://en.wikipedia.org/wiki/Convolution) operator (and its counterpart [correlation](https://en.wikipedia.org/wiki/Cross-correlation)) in the latter example is one of the key concepts that you will learn and apply in this course. Thus, it is worth spending some time on developing your intuition of what this is. Let's first discuss the geophysical example of convolutional modeling that is one way to think about forward modeling both seismic and GPR data.

Convolutional Modeling of Seismic Data

The Earth's P-wave acoustic impedance ($I_p = \rho V_p$) is commonly represented in terms of 'reflectivity' (i.e., $r(t) \approx \frac{V_p}{2} \frac{d}{dz} \ln(\rho V_p)$). One straightforward approach to modeling a 1D (or higher dimension) seismic section is to **convolve** a **seismic wavelet**, $s(t)$, with an **Earth reflectivity sequence**, $r(t)$. The **output** of this operation, $y(t)$, is approximately the **seismic trace** you would record between a co-located source and receiver pair.

Mathematically, this operation is denoted as:

$$y(t) = s(t) * r(t) + n(t), \quad (1)$$

where $*$ represents the convolution operation here between $s(t)$ and $r(t)$, and $n(t)$ is some additive noise. Figure 1 illustrates the seismic convolutional modeling approach.

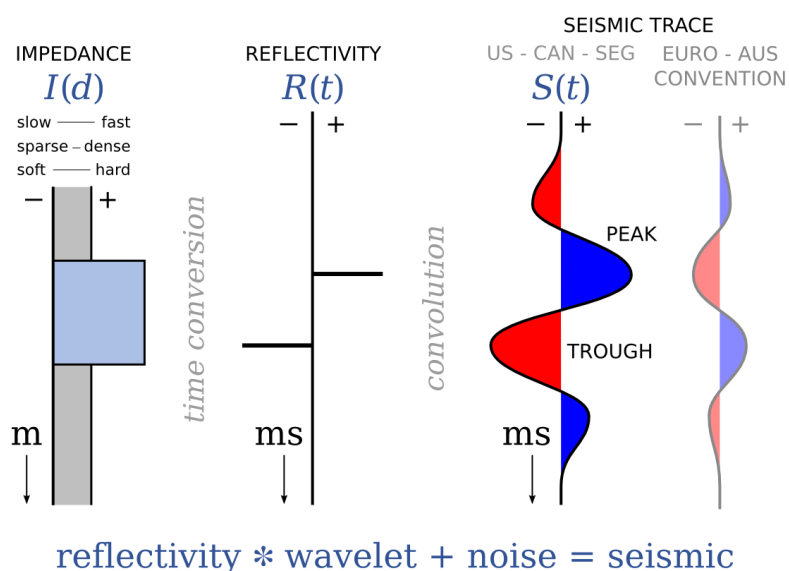


Figure 1. Illustration of the convolutional seismic modelling as a LTI system. The leftmost figure represents a 1D geological profile, while the second panel shows the locations of the discontinuities (i.e., the reflector locations) where the magnitude of the discontinuity is proportional to the impedance contrast. The third and fourth panels show the convolution of the reflectivity sequence with seismic wavelets of the North American and European conventions, respectively.

Two Key Properties of LTI systems

As per the name, a LTI system (here defined by $H[\cdot]$) has two key necessary ingredients:

- [Linearity](https://en.wikipedia.org/wiki/Linear_system) (https://en.wikipedia.org/wiki/Linear_system): if $[x_1(t), y_1(t)]$ and $[x_2(t), y_2(t)]$ are two input-output pairs (i.e., $y_1(t) = H[x_1(t)]$ and $y_2(t) = H[x_2(t)]$ for some LTI operator H), then:

$$ax_1(t) + bx_2(t) \rightarrow ay_1(t) + by_2(t). \quad (2)$$

This principle simply states that you may superimpose two different input signals and the resulting output signal will be the same superimposition. In the convolutional modeling example above, this would mean that you could: (1) convolve each component of the reflection sequence with the wavelet individually and then stack the result; or (2) convolve the wavelet with the entire reflectivity sequence all at once. Following either scenario will generate the same result.

- [Time-invariance](https://en.wikipedia.org/wiki/Time-invariant_system) (https://en.wikipedia.org/wiki/Time-invariant_system): If $x(t) \rightarrow y(t)$ then

$$x(t - t_0) \rightarrow y(t - t_0). \quad (3)$$

This implies that delaying or advancing the operation by a constant t_0 produces no affect on the resulting system. In the seismic convolutional modeling example above, this means that if you shifted the input reflectivity sequence $r(t)$ to $r(t - t_0)$, you would expect the output seismic trace to be similarly shifted from $y(t)$ to $y(t - t_0)$.

Convolution (Integral)

One of the most important aspects of an LTI system is that the input-output relationship is characterized by a [convolution](https://en.wikipedia.org/wiki/Convolution) (<https://en.wikipedia.org/wiki/Convolution>) operation alluded to - but not defined - above. The convolution integral is written in the following way:

$$y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} x(\tau) h(t - \tau) d\tau, \quad (4)$$

where $x(t)$, $y(t)$ and $h(t)$ are the input, output and [impulse response](https://en.wikipedia.org/wiki/Impulse_response) (https://en.wikipedia.org/wiki/Impulse_response) of the LTI system. In equation 4 we see that a dummy integration variable τ has been introduced. You'll also notice that one function, $h(t - \tau)$, is **time-reversed** relative to the other variable.

Here is a short pseudo-code to think about the mathematical operation in equation 4:

```
For each $t$ in the output space:
  For each $tau$ in the input space:
    Multiply $x(tau)$ with time-reversed $h(t-tau)$
    Accumulate result into output $y(t)$
```

Let's now look a couple of examples.

Evaluating the convolution integral as a movie

It is an instructive exercise to plot $x(t)$ and a time reversed $h(t)$ on two different pieces of paper. Initially line the up with no shift applied (i.e., $\tau = 0$). Mentally multiply two functions together and sum the result to get your output $y(0)$. You can then repeat this for different values of τ to build up $y(t)$.

Let's look at a few movie examples that illustrate this procedure. First, let's examine the **autoconvolution** of the boxcar with itself

Out[2]:

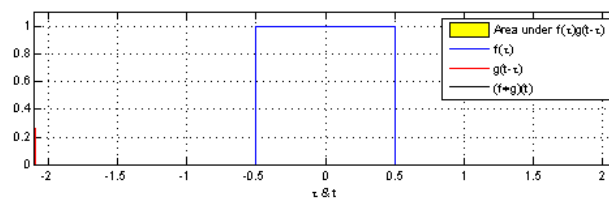


Figure 2. Illustration of convolution. The blue curve is the stationary boxcar $f(\tau)$ while red curve, $g(t - \tau)$, is the time-reversed signal. For each shift in the convolution (corresponding to the variable t), one must multiply the blue and red curves together to get the yellow area, which represents the convolution value at that shift. The overall value of $(f * g)(t) = f(t) * g(t)$ is shown by the black line shaped as a triangle. Note that the value of the yellow area at any particular time shift is the convolution result $(f * g)(t) = f(t) * g(t)$.

Let's now look at another example of a box car $g(t)$ convolved with a spikey decaying exponential function $f(t)$.

Out[3]:

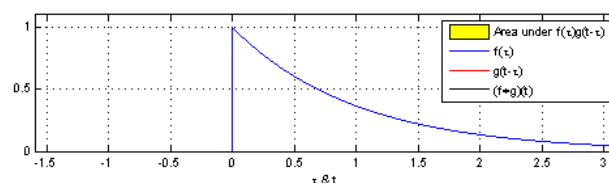


Figure 3. As in Figure 2 but with a one-sided function.

Example 1a - Spike-Gaussian Convolution

Let's look at the (numerical) convolution of a single input delta function spike $f(t)$ and a supposed Gaussian impulse response of the system $g(t)$ that generates our output $f(t) * g(t)$. We'll repeat this example four times where we apply a different delay to the input spike.

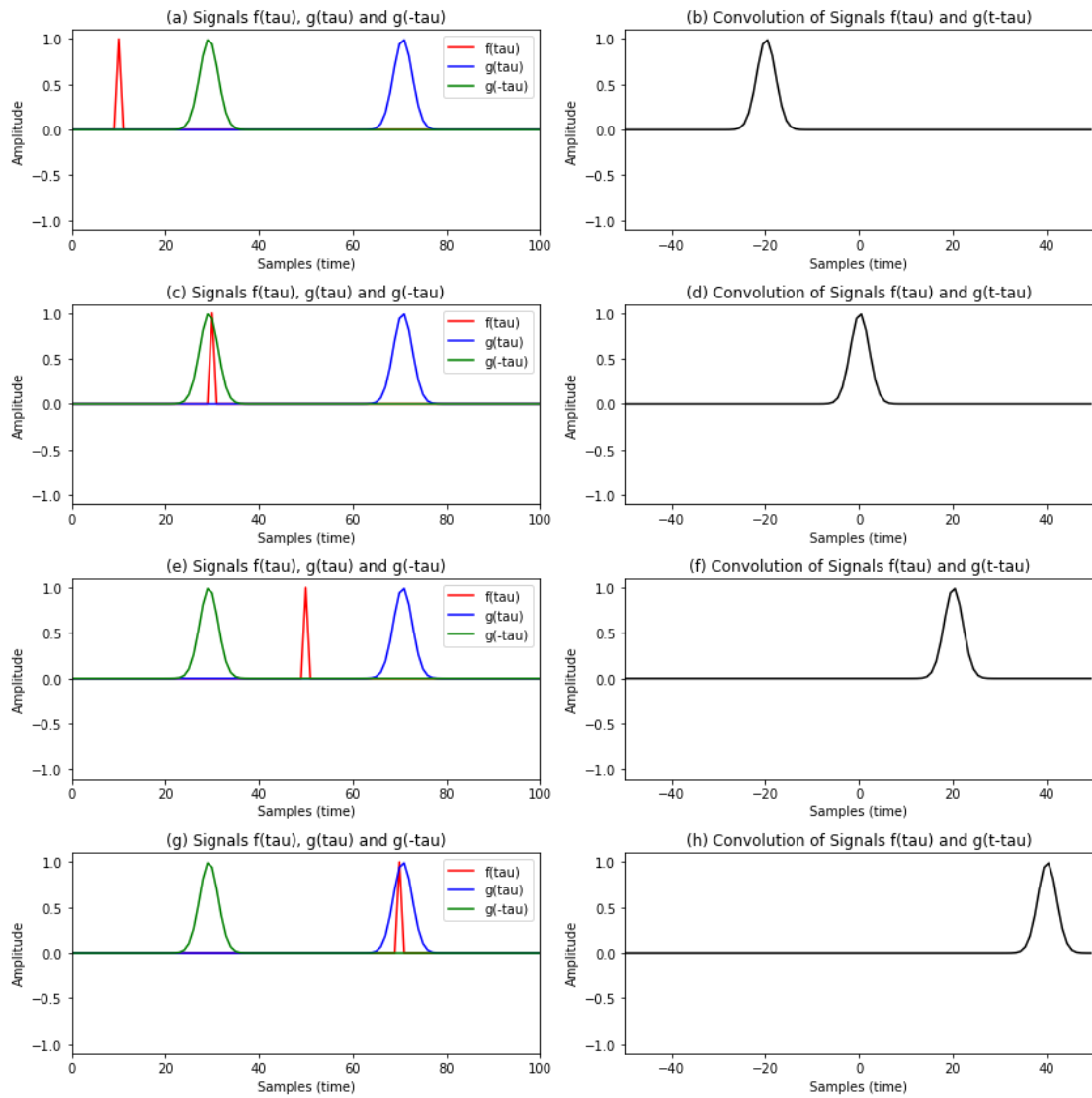


Figure 4. Examples of convolving a spike with a Gaussian Function. The left panels show the spike function $f(\tau)$ in red along with the Gaussian function in blue and its time-reversed version in Green. The right panels show the result of convolving the two functions. Note that if the red curve is left of the green curve this results in a negative values of the convolution. The reverse is true if the red curve is to the right of the green curve.

Let's now look at the convolution of a Gaussian with a sequence of pulses that are spread out like an earth reflectivity model described above:

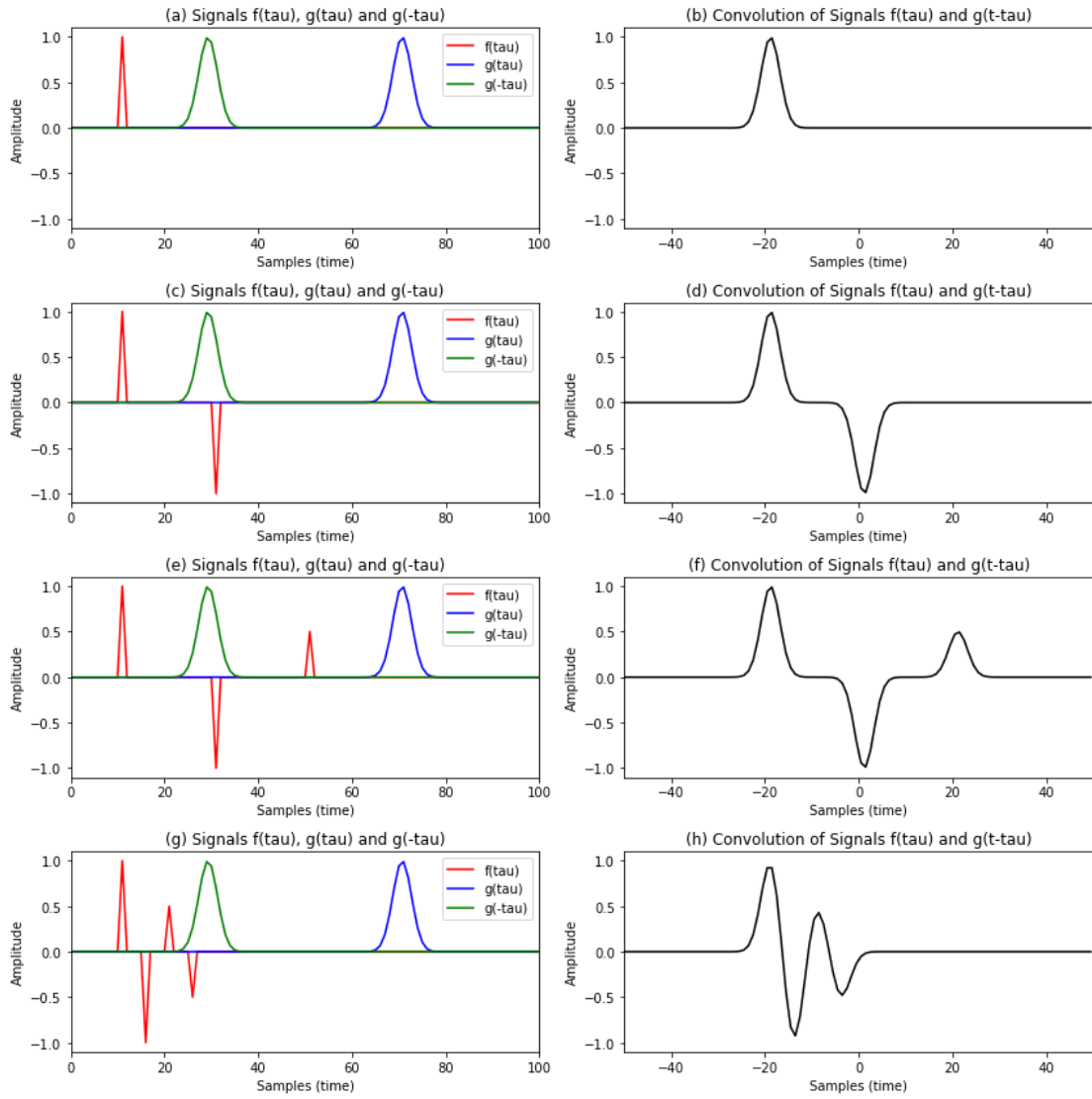


Figure 5. Similar to Figure 4, but now where I have introduced a number of spikes with different spacings.

Convolution in the Fourier Domain

An important property is that **convolution** in the time domain corresponds to **multiplication** in the Fourier transform domain:

$$y(t) = x(t) * h(t) \Leftrightarrow \hat{X}(\omega)\hat{H}(\omega) \quad (5)$$

Proof of the Convolution Theorem:

The Fourier transform of $y(t) = x(t) * h(t)$ is:

$$\begin{aligned} \hat{Y}(\omega) &= \mathcal{F}[x(t) * h(t)] \\ &= \mathcal{F}\left[\int_{-\infty}^{\infty} x(\tau) h(t - \tau) d\tau\right] \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(\tau) h(t - \tau) e^{-i\omega t} d\tau dt \end{aligned} \quad (6)$$

Let's now assert that $t = u + \tau$ where u is the new integration variable and τ is a constant shift. Thus, we may write $dt = du$. Inserting this into the equation above yields:

$$\begin{aligned} \hat{Y}(\omega) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(\tau) h(u) e^{-i\omega(u+\tau)} d\tau du \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(\tau) h(u) e^{-i\omega\tau} e^{-i\omega u} d\tau du \\ &= \left[\int_{-\infty}^{\infty} x(\tau) e^{-i\omega\tau} d\tau \right] \left[\int_{-\infty}^{\infty} h(u) e^{-i\omega u} du \right] \\ &= \hat{X}(\omega) \hat{H}(\omega) \end{aligned} \quad (7)$$

Thus, the output signal of a convolution in the frequency domain is equal to the product of the Fourier Transforms of input functions. This is one of the most important properties of convolution.

Example 2 - Convolution of Two Boxcars

This example looks at the (auto)convolution of two boxcar functions of equal width. You might recognize the result ... a triangle function! Recall from an earlier example that

$$\Pi_T(t) \Longleftrightarrow 2T \operatorname{sinc}(T\omega) \quad (8)$$

Therefore, by the convolution theorem:

$$\Pi_T(t) * \Pi_T(t) \Longleftrightarrow 4T^2 \operatorname{sinc}^2(T\omega) \quad (9)$$

You'll also recall that a sinc^2 function is the Fourier Transform of the triangle function. Therefore, you probably wouldn't be surprised to hear that the autoconvolution of a boxcar function is the triangle function! This is illustrated in the example below.

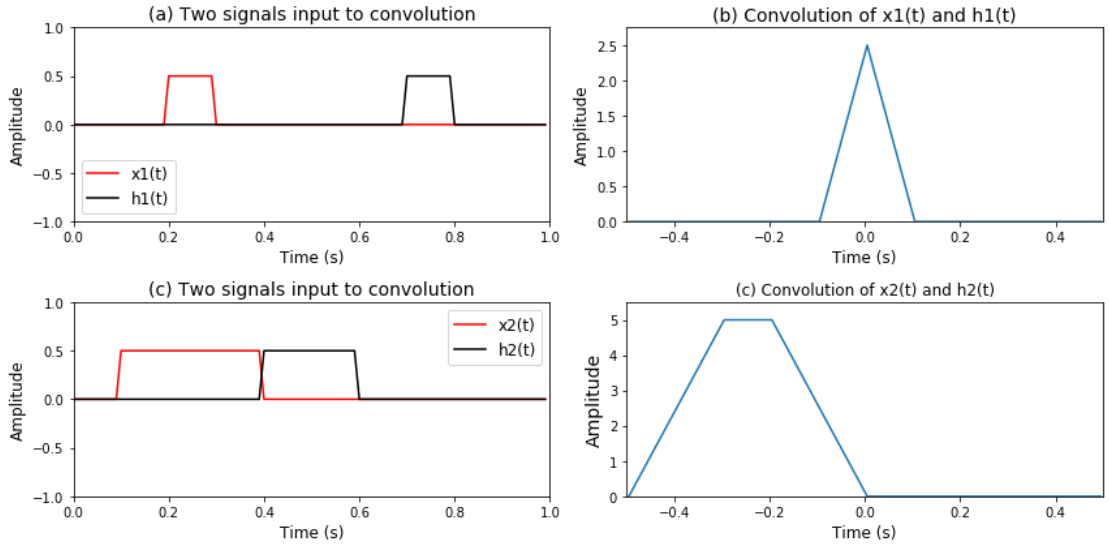


Figure 6. (a) Two box car functions where the time-reversed version would fall exactly on the red curve. (b) The convolution of the two functions in (a) effectively representing the autoconvolution of the signal. (c) Two box cars where the red curve is now longer and the black curve is shifted. (d) Convolution of the signals in (c) Note that the result is both broader and has a flat top corresponding to shifts where the black box car is completely within the red box car.

Thus, to compute the autoconvolution of the boxcar function you can do either one of two things:

1. Compute the convolution of $y(t) = h(t) * x(t)$ in the physical domain using a convolution program.
2. Compute the Fourier transforms of $h(t)$ and $x(t)$, multiply the results (i.e., $\widehat{Y}(\omega) = \widehat{X}(\omega)\widehat{H}(\omega)$), and then take the inverse Fourier transform of $\widehat{Y}(\omega)$ to get the desired result $y(t)$.

Commutativity of the convolution operator

Let's look at proving the following:

$$f(t) * g(t) = g(t) * f(t) \quad (10)$$

First, let's do it analytically:

COMMUTATIVITY PROOF:

The definition of convolution is given by:

$$f(t) * g(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau \quad (11)$$

However, we may perform a substitution of $t' = t - \tau$, $d\tau = -dt'$ and $\tau = t - t'$ in the above equation.

$$f(t) * g(t) = - \int_{t+\infty}^{t-\infty} f(t - t')g(t')dt' \quad (12)$$

However, care must be paid to the integration limits. First, we note that as $\tau \rightarrow -\infty$ we get $t' = t + \infty = +\infty$. Similarly, as $\tau \rightarrow +\infty$ we get $t' = t - \infty = -\infty$. This is why we have had to change the integration limits in equation B. The next step is that we reverse the integration limits, which changes the sign of the integral. Thus, we may write:

$$f(t) * g(t) = - \int_{+\infty}^{-\infty} f(t - t')g(t')dt' = \int_{-\infty}^{+\infty} g(t')f(t - t')dt' = g(t) * f(t) \quad (13)$$

Thus, the convolution operator is commutative. This should be expected from the Fourier convolution theorem which states:

$$f(t) * g(t) \Leftrightarrow \widehat{F}(\omega)\widehat{G}(\omega) = \widehat{G}(\omega)\widehat{F}(\omega) \Leftrightarrow g(t) * f(t) \quad (14)$$

Let's now show it computationally:

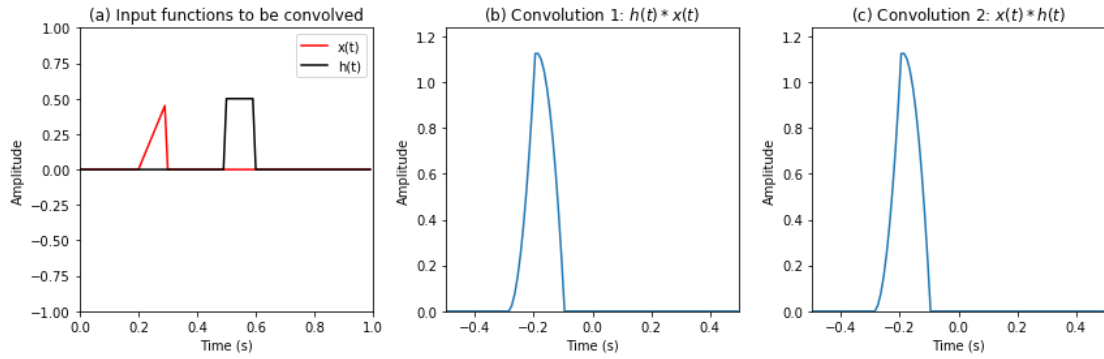


Figure 7. Illustrating the commutativity of the convolution operator. (a) Two function to be convolved. (b) Convolution $h(t) * x(t)$. (c) Convolution $x(t) * h(t)$.

Convolution (Computational Efficiency)

From a computational perspective, as the number of samples in a time series gets "large" it becomes more efficient to compute a (fast) Fourier Transform, perform Fourier domain multiplication and then take the inverse Fourier transform. The [computational complexity](https://en.wikipedia.org/wiki/Computational_complexity_theory) (https://en.wikipedia.org/wiki/Computational_complexity_theory) of this operation is $\mathcal{O}(n \log n)$ rather than $\mathcal{O}(n^2)$ of performing convolution directly in the time domain.

```
/Users/jeffreyshragge/miniconda3/lib/python3.5/site-packages/matplotlib/axes/_base.py:1708: UserWarning: Attempted
to set non-positive bottom ylim on a log-scaled axis.
Invalid limit will be ignored.
self.set_ylim([v[2], v[3]], emit=emit, auto=False)
```

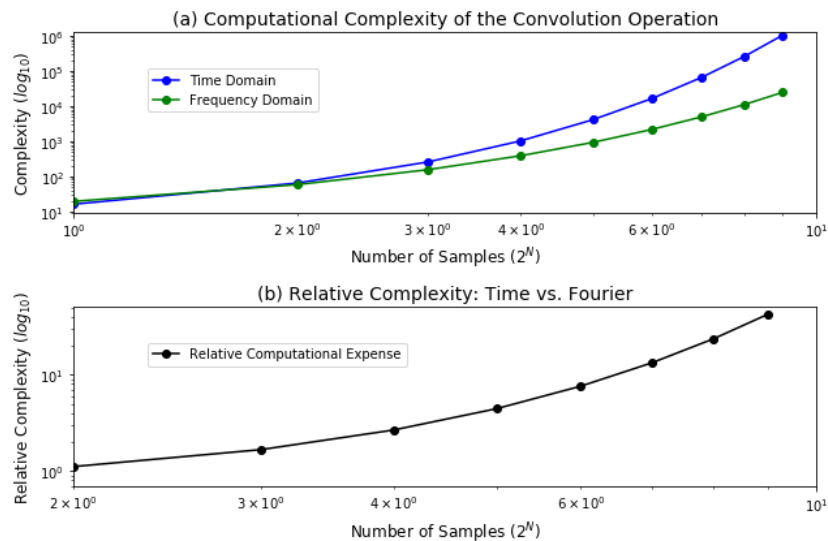


Figure 8. (a) Illustration of the computational complexity of time-domain (blue curve) and frequency-domain (green curve) computation of the convolution operation. (b) The relative cost of the two curves in (a).

Other examples of physical convolutional LTI systems

GPR Acquisition system

Consider that you are recording a geophysical signal (e.g., GPR wave) denoted by $x(t)$. The instrument (GPR receiving antenna) you are using has a natural response, $h(t)$. The recorded GPR data you measure, $y(t)$, will be affected by instrument you are using in a convolution process $y(t) = x(t) * h(t)$. Thus, recorded data $y(t)$ will not be the "true" response $x(t)$ unless: (1) you have a "perfectly non-distorting" instrument (i.e., $h(t) = \delta(t)$); or (2) you account for the instrument's response through a [deconvolution](https://en.wikipedia.org/wiki/Deconvolution) (<https://en.wikipedia.org/wiki/Deconvolution>) procedure to recover $x(t)$ (that is, assuming you know $h(t)$):

$$x(t) = \mathcal{F}^{-1} \left[\frac{\widehat{Y}(\omega)}{\widehat{H}(\omega)} \right] = \mathcal{F}^{-1} \left[\frac{\mathcal{F}[y(t)]}{\mathcal{F}[h(t)]} \right]. \quad (15)$$

Q: What are the potential problems or challenges with a deconvolutional approach?

Human hearing

There is a far-field acoustic source $x(t)$ in your presence. Your hearing system has two inputs $h_L(t)$ and $h_R(t)$ (i.e., your ears!). Unfortunately, they are an imperfect sensing instrument and thus what you actually "hear" is a convolution product.

$$y(t) = [h_L(t) + h_R(t)] * x(t) \quad (16)$$

Hearing aids with responses $a_L(t)$ and $a_R(t)$ can be introduced to improve your response!! This leads to a more complicated LTI model:

$$y(t) = [a_L(t) * h_L(t) + a_R(t) * h_R(t)] * x(t) \quad (17)$$

See Figure 9 for an illustration.

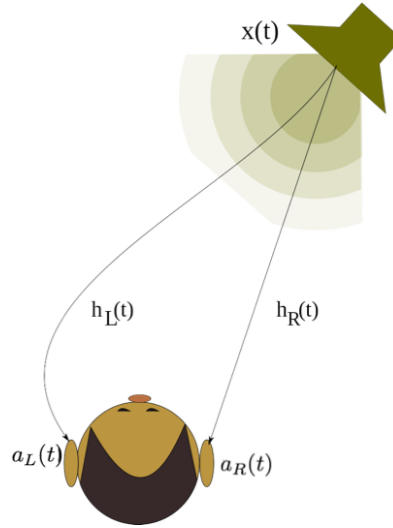


Figure 9. Illustration of the ears plus hearing aids as an LTI system. In this scenario an acoustic source produces a signal $x(t)$ that propagates through the air and is detected by you the listener. Your hearing system has two detectors, your left and right ears $h_L(t)$ and $h_R(t)$. Your brain thus detects the convolution of your ear's responses and the signal. Of course if you have hearing aids this is more complicated since your ear response is now the convolution of with the hearing aid response as well!

Convolution in Action

Convolution with a δ -function

Convolution of a function $f(t)$ with a δ -function shifted by a unit gives

$$f(t - a) = \int_{-\infty}^{\infty} f(t) \delta(t - a) dt \quad (18)$$

by virtue of the properties of the δ -function. We may write this symbolically as

$$f(t) * \delta(t - a) = f(t - a). \quad (19)$$

Applying the convolution theorem to this result [because $\delta(t - a) = \delta(a - t)$ in equation 13 above] since it yields the Fourier shift theorem:

$$f(t) \Leftrightarrow \widehat{F}(\omega); \quad \delta(t - a) \Leftrightarrow e^{-i\omega a}, \quad (20)$$

which yields the following:

$$f(t - a) = f(t) * \delta(t - a) \Leftrightarrow \widehat{F}(\omega) e^{-i\omega a}. \quad (21)$$

which is the multiplication of the two signals' Fourier spectra!

Convolution with a pair of δ -functions

Based on our previous exploration of the Fourier transform of δ -functions, we can identify the following:

$$[\delta(t - a) + \delta(t + a)] \Leftrightarrow 2 \cos(\omega a), \quad (22)$$

with a similar result holding if the delta functions were defined in the Fourier domain. The convolution of the expression in equation 17 with a function $f(t)$ is given by

$$[\delta(t - a) + \delta(t + a)] * f(t) \Leftrightarrow 2 \cos(\omega a) \widehat{F}(\omega), \quad (23)$$

which is again the product of the signals' respective Fourier transforms.

Two Gaussian functions

An interesting example that comes up quite frequently in probability theory (including geophysical inverse problems when modeling probability density functions) is the convolution of two Gaussian functions. Say we have two Gaussian functions of width a and b , $f(t) = e^{-t^2/a^2}$ and $g(t) = e^{-t^2/b^2}$ respectively. In the Fourier domain we know that the respective Fourier transforms are given by:

$$\widehat{F}(\omega) = a\sqrt{\pi} e^{-a^2\omega^2/4} \quad (24)$$

and

$$\widehat{G}(\omega) = b\sqrt{\pi} e^{-b^2\omega^2/4}. \quad (25)$$

Thus, according to the Fourier domain convolution theorem, we may write:

$$f(t) * g(t) = \mathcal{F}^{-1} \left[\widehat{F}(\omega) \cdot \widehat{G}(\omega) \right] \quad (26a)$$

$$= \mathcal{F}^{-1} \left[ab\pi e^{-(a^2+b^2)\omega^2/4} \right] \quad (26b)$$

$$= \frac{ab\pi}{\sqrt{a^2+b^2}} e^{-t^2/(a^2+b^2)}. \quad (26c)$$

Importantly, the convolution of two Gaussians with widths a and b is also a Gaussian of width $\sqrt{a^2+b^2}$. Let's test this out with a numerical example.

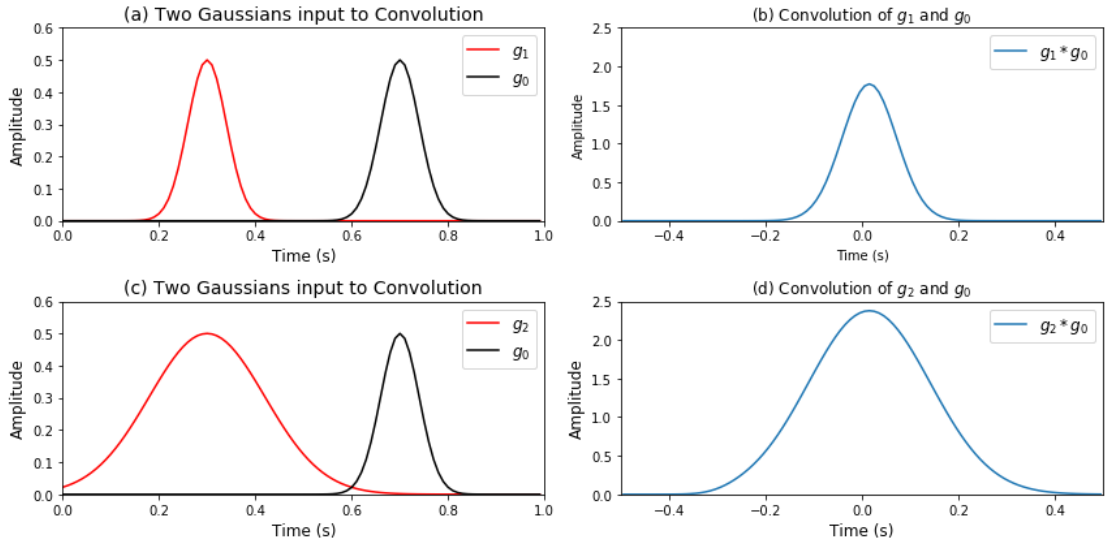


Figure 10. Illustrating the convolution of two Gaussian functions. (a) Two shifted but otherwise identical Gaussian functions. (b) The convolution of functions in (a). (c) Two shifted and unequal Gaussian Functions. (d) The convolution of functions in (c). Note that convolving Gaussian functions will broaden the resulting Gaussian function.

The Algebra of Convolutions

One has to take care when applying more complex convolution operators. For example, generally speaking:

$$[a(t) * b(t)] \cdot c(t) \neq a(t) * [b(t) \cdot c(t)] \quad (27)$$

One can use the properties of the convolution operation to derive some truly convoluted relationships:

$$[a(t) * b(t)] \cdot [c(t) * d(t)] \Leftrightarrow [\widehat{A}(\omega) \cdot \widehat{B}(\omega)] * [\widehat{C}(\omega) \cdot \widehat{D}(\omega)] \quad (28)$$

The Derivative Theorem

If $f(t) \Leftrightarrow \widehat{F}(\omega)$, then

$$\widehat{G}(\omega) = \mathcal{F} \left[\frac{df(t)}{dt} \right] = i\omega \widehat{F}(\omega). \quad (29)$$

which is sometimes written symbolically as a Fourier Transform Pair

$$\frac{d}{dt} \Leftrightarrow i\omega. \quad (30)$$

Let's now look at a proof of the Derivative Theorem:

Let's begin with our definition of the Fourier Transform of $f(t)$ and it's derivative:

$$\widehat{F}(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt \quad (31)$$

and

$$\widehat{G}(\omega) = \int_{-\infty}^{\infty} \frac{df(t)}{dt} e^{-i\omega t} dt. \quad (32)$$

Let's look at the second integral and apply integration by parts using $u = e^{-i\omega t}$, $dv = \frac{df}{dt} dt$, $v = f$ and $du = -i\omega e^{-i\omega t} dt$. Thus,

$$\int_{-\infty}^{\infty} u dv = uv|_{-\infty}^{\infty} - \int_{-\infty}^{\infty} v du \quad (33)$$

may be written:

$$\widehat{G}(\omega) = \int_{-\infty}^{\infty} \frac{df(t)}{dt} e^{-i\omega t} dt = f(t)e^{-i\omega t}|_{-\infty}^{\infty} + \int_{-\infty}^{\infty} f(t)i\omega e^{-i\omega t} dt \quad (34)$$

The middle term must be zero in both accounts in order for $f(t)$ to be an admissible function for Fourier transform. Thus, we may write:

$$\widehat{G}(\omega) = \int_{-\infty}^{\infty} \frac{df(t)}{dt} e^{-i\omega t} dt = i\omega \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt = i\omega \widehat{F}(\omega) \quad (35)$$

which is the equivalent to equation 29 above.

Note that if we defined our Fourier transform with the opposite convention (i.e., $e^{i\omega t}$ instead of $e^{-i\omega t}$) then the result would be $\frac{d}{dt} \Leftrightarrow -i\omega$, which is commonly found in mathematical physics textbooks.

The Fourier Derivative theorem also extends to higher dimensions:

$$\frac{d^n f(t)}{dt^n} = (i\omega)^n \widehat{F}(\omega). \quad (36)$$

This is a very important thing to remember!

Note that this also holds for partial differential equations. For example, if we have the 1D acoustic wave equation

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2 u}{\partial t^2} \quad (37)$$

we can use the fact that $\frac{\partial}{\partial t} \Leftrightarrow i\omega$ and $\frac{\partial}{\partial x} \Leftrightarrow ik_x$ to immediately write:

$$(ik_x)^2 = \frac{(i\omega)^2}{c^2} \quad (38)$$

or

$$k_x = \pm \frac{\omega}{c}, \quad (39)$$

which you may recognize as a dispersion relationship linking wavenumbers and frequencies!

The Convolution Derivative Theorem

The following is true:

$$\frac{d}{dt}[f(t) * g(t)] = \frac{df(t)}{dt} * g(t) = f(t) * \frac{dg(t)}{dt} \quad (40)$$

Extention Problem: Prove that this is correct.

Rayleigh's Theorem

The following is true:

$$\int_{-\infty}^{\infty} f(t)\overline{g(t)}dt = \int_{-\infty}^{\infty} \widehat{F}(\omega)\overline{\widehat{G}(\omega)}d\omega \quad (41)$$

Two particular cases of interest are Rayleigh's theorem

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \int_{-\infty}^{\infty} |\widehat{F}(\omega)|^2 d\omega, \quad (42)$$

and the power in a periodic signal (represented by a Fourier series):

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \frac{A_0^2}{4} + \frac{1}{2} \sum_{k=1}^{\infty} (A_k^2 + B_k^2). \quad (43)$$

2D Convolution Theorem

Like its 1D cousin, **2D convolution** may be expressed in the following way:

$$g(x, y) = h(x, y) * f(x, y) = f(x, y) * h(x, y) \quad (44)$$

In integral notation, 2D convolution may be expressed as

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\tau_x, \tau_y) h(x - \tau_x, y - \tau_y) d\tau_x d\tau_y. \quad (45)$$

Similarly to the 1D scenario, one can develop an expression for the **2D Convolution Theorem**:

$$f(x, y) * h(x, y) \Leftrightarrow \widehat{F}(k_x, k_y) \widehat{G}(k_x, k_y) \quad (46)$$

Time-domain example

Let's look at an example of a 2D convolution involving the a picture (i.e., $f(x, y)$) and a 2D Gaussian filter (i.e., $h(x, y)$) of 20 points in diameter. The result was generated by applying a 2D convolution directly in the time domain.

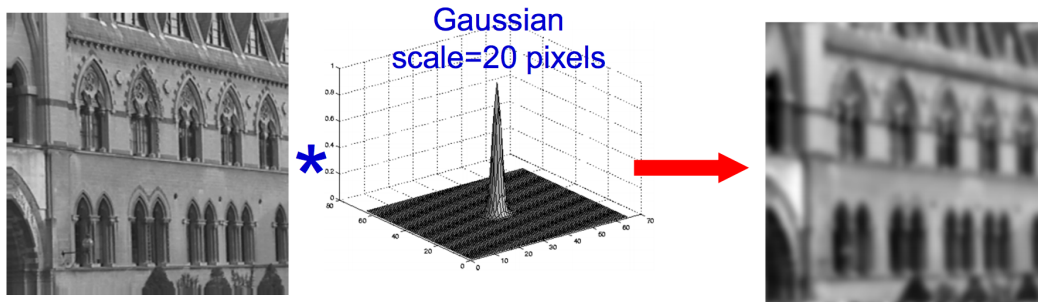


Figure 11. Example of a 2D Convolution involving a 2D Gaussian filter of 20 points in diameter. The left panel represents $f(x, y)$, the middle panel represents impulse response $h(x, y)$ and the right panel is the convolution of $f(x, y)$ and $g(x, y)$ (i.e., $g = h * f$).

Fourier-domain example

Now let's recompute the previous example, but now using the **2D Convolution Theorem**. This evaluation: (1) applies 2D Fourier transforms of both f and h to obtain \widehat{F} and \widehat{G} ; (2) multiplies these two functions together to obtain \widehat{G} ; and then (3) applies an inverse Fourier transform to recover g . Note that this example uses the following notation: $[u, v = k_x, k_y]$.**

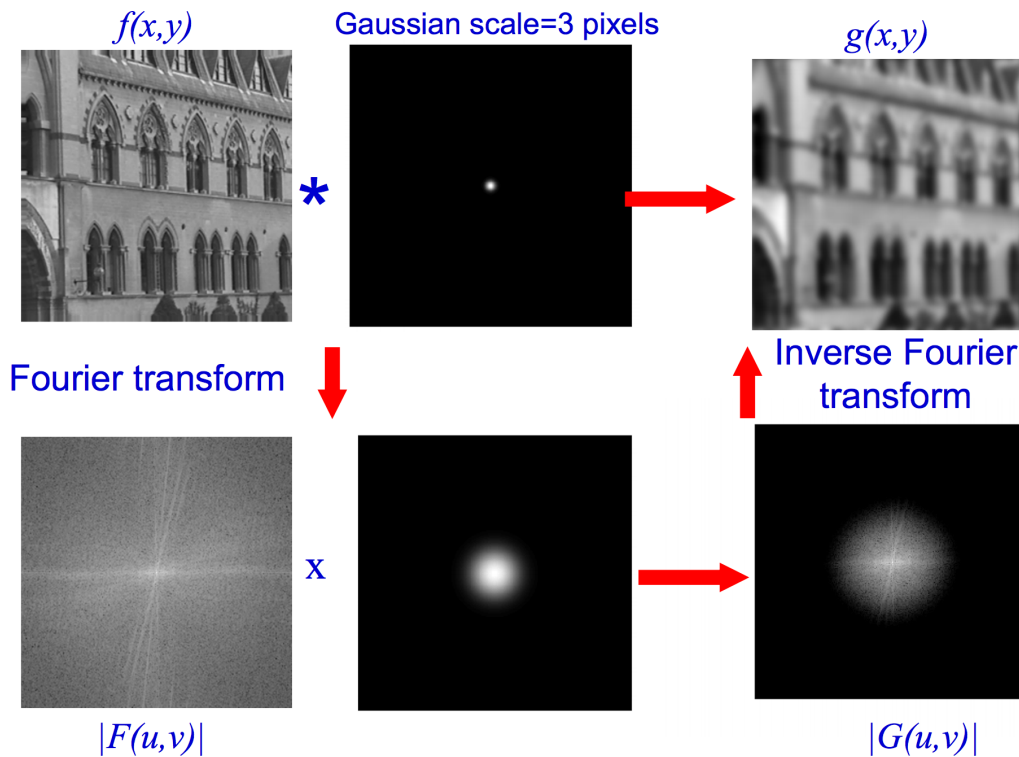


Figure 12. Example of a 2D Convolution involving a 2D Gaussian filter of 20 points in diameter using the 2D Convolution Theorem. Note that this example uses the alternative 2D FT notation where $[u, v]$ is exchanged for $[k_x, k_y]$.

Crosscorrelation

An important operation related to convolution is [cross correlation](https://en.wikipedia.org/wiki/Cross-correlation) (<https://en.wikipedia.org/wiki/Cross-correlation>). This operation is denoted by the \star operator and is closely related to convolution. The integral for cross correlation is given by:

$$f(\tau) \star g(\tau) = \int_{-\infty}^{\infty} \overline{f(t)} g(t + \tau) dt \quad (47)$$

where $\overline{f(t)}$ is the complex conjugate and unlike in convolution $g(t + \tau)$ is not time reversed. The τ axis may be considered as the **correlation lag** or **time delay** between the two signals f and g .

In numpy, the correlation of two signals f and g can be calculated via

```
np.correlate(f,g,mode='full')
```

where there is again the option for `mode='same'` to return a result of the same length as two equal-length input arrays (if applicable).

Let's look at an example using the `mode='full'` operation:

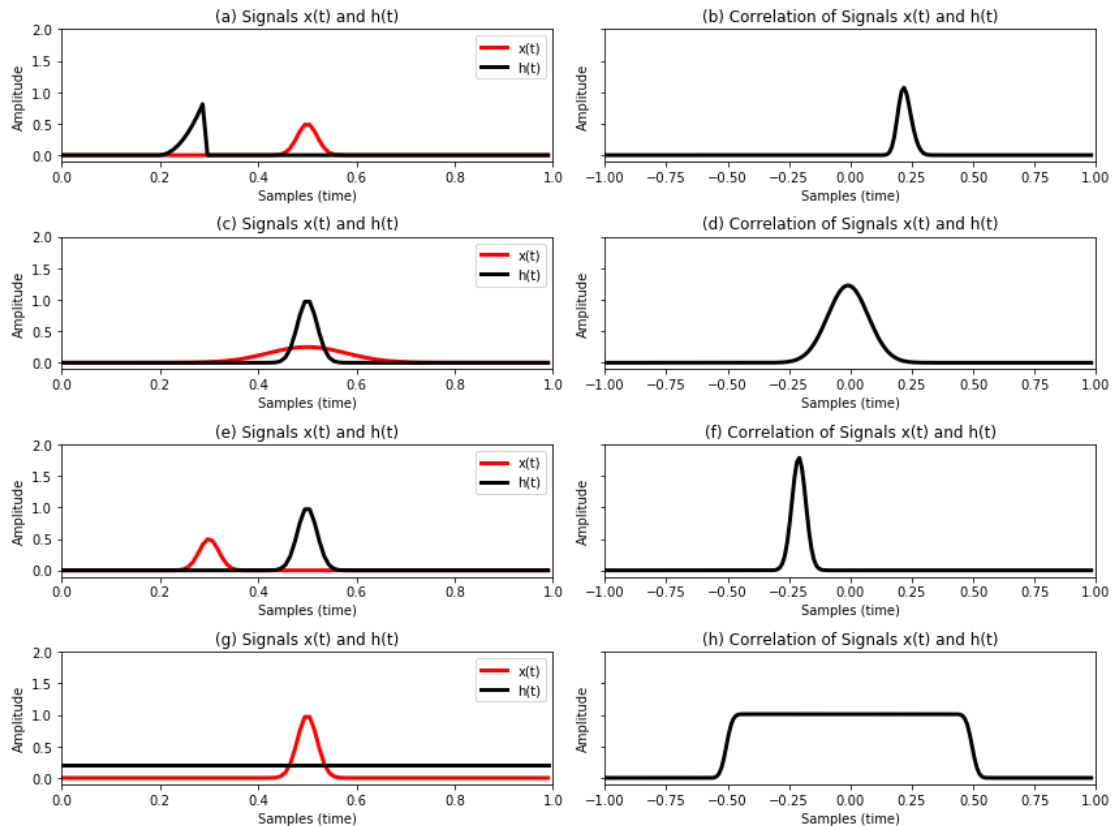


Figure 13. Illustration of crosscorrelation of two signals. Left panels: Two signals $x(t)$ and $h(t)$ to be crosscorrelated. Right panels: crosscorrelation result. Note that this operation is not commutative.

WARNING: You'll notice that the output of this operation is from -1s to 1s, which is different from the above. This is because in this case we have used `np.correlate(a,b,mode='full')` instead of `np.correlate(a,b,mode='same')` as in the convolution examples above. If you have arrays a and b of dimension N , then the 'same' option will return an array with dimension N ranging between $-N/2$ to $N/2$. Conversely, the 'full' option will return an array of dimension $2N-1$ ranging from $-N$ to N . In the above example we have arrays defined between 0-1s. Thus, because we have called the 'full' option, the function has returned the correlation result between ± 1 s. This is because the extra delays between the two signals can be ± 1 .

Crosscorrelation Properties

The correlation integral has the following important properties:

- (1) The cross-correlation of $f(t)$ and $g(t)$ is equivalent to the **convolution** of $\overline{f(-t)}$ and $g(t)$.

$$f(t) \star g(t) = \overline{f(-t)} * g(t) \quad (48)$$

- (2) If $f(t)$ is a [Hermitian function](https://en.wikipedia.org/wiki/Hermitian_function) (i.e., has the property of $\overline{f(t)} = f(-t)$) then the following holds:

$$f(t) \star g(t) = f(t) * g(t) \quad (49)$$

- (3) If $f(t)$ and $g(t)$ are both Hermitian functions, then

$$f(t) \star g(t) = g(t) \star f(t) \quad (50)$$

- (4) The cross-correlation of two functions [i.e., $h(t) = f(t) \star g(t)$] satisfies the following in the Fourier domain:

$$\widehat{H}(\omega) = \mathcal{F}[h(t)] = \mathcal{F}[f(t) \star g(t)] = \mathcal{F}[\overline{f(-t)} * g(t)] = \widehat{\overline{F}}(\omega) \widehat{G}(\omega) \quad (51)$$

The difference between convolution and correlation

Let's now look at an example showing the difference between the convolution and cross-correlation operations. In the image below, I have created two signals $x(t)$ and $h(t)$. The second panel shows the **convolution** $x(t) * h(t)$, while the third panel shows the **convolution** $h(t) * x(t)$. These are identical as should be expected from the commutivity of the convolution operator discussed above. The fourth panel shows the **correlation** $y_1(t) = x(t) \star h(t)$, while the fifth panel shows the **correlation** $y_2(t) = h(t) \star x(t)$. You'll notice in this case that $y_2(t) = y_1(-t)$.

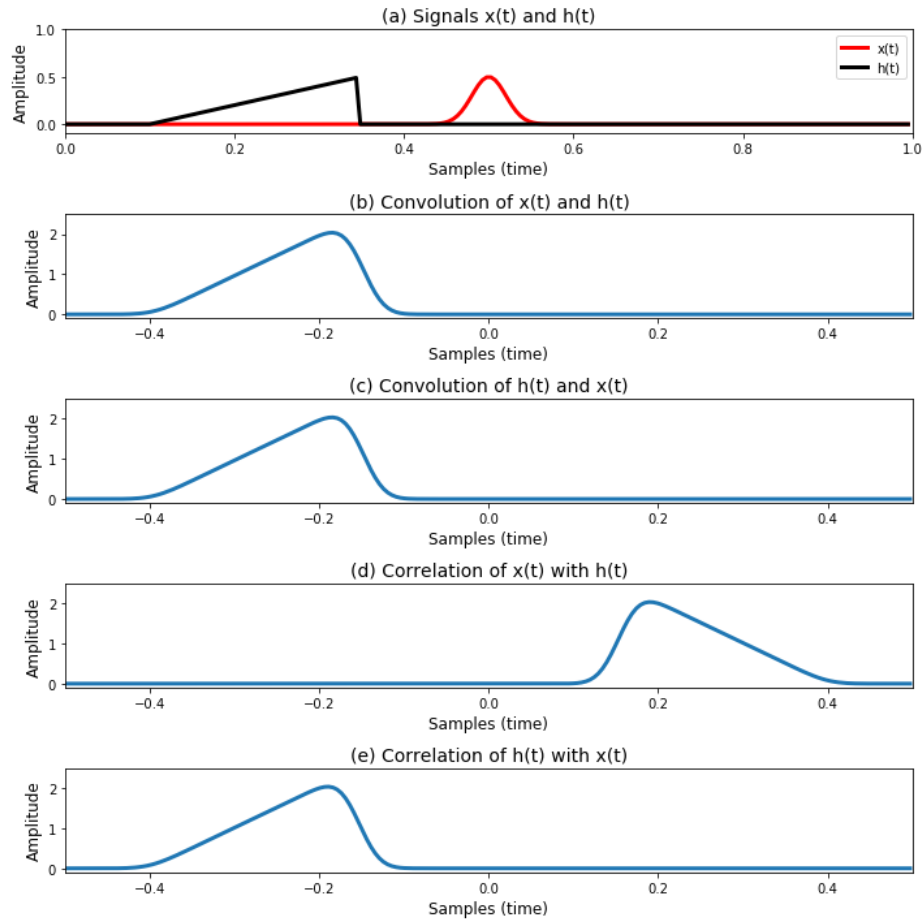


Figure 14. Illustrating the differences between convolution and crosscorrelation. (a) Two signals to be analyzed. (b) Convolution $x(t) * h(t)$. (c) Convolution $h(t) * x(t)$, which yields the same as (b). (d) Crosscorrelation of $x(t) \star h(t)$. (d) Crosscorrelation of $h(t) \star x(t)$, which yields a time-reversed version of (d).

Autocorrelation Theorem

The **autocorrelation** of a function $f(t)$, here denoted by $a_f(t) = f(t) \star f(t)$, has the following relationship:

$$\widehat{A_f}(\omega) = \widehat{F}(\omega) \overline{\widehat{F}(\omega)} = |\widehat{F}(\omega)|^2. \quad (52)$$

****PROOF:**** Let's start with the definition of cross-correlation:

$$a_f(t) = \int_{-\infty}^{\infty} \overline{f(t')} f(t+t') dt' \quad (53)$$

Applying a Fourier Transforms to both sides

$$\widehat{A_f}(\omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \overline{f(t')} f(t+t') e^{-i\omega t} dt' dt. \quad (54)$$

Let $\tau = t + t'$. Then if t' is held constant, $dt = d\tau$,

$$\widehat{A_f}(\omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \overline{f(t')} f(\tau) e^{-i\omega(\tau-t')} d\tau dt'. \quad (55)$$

which can be separated

$$\widehat{A_f}(\omega) = \int_{-\infty}^{\infty} \overline{f(t')} e^{i\omega t'} dt' \cdot \int_{-\infty}^{\infty} f(\tau) e^{-i\omega \tau} d\tau \quad (56)$$

which leads to

$$\widehat{A_f}(\omega) = \widehat{F}(\omega) \overline{\widehat{F}(\omega)} = |\widehat{F}(\omega)|^2. \quad (57)$$

Let's now look at an example of autocorrelation.

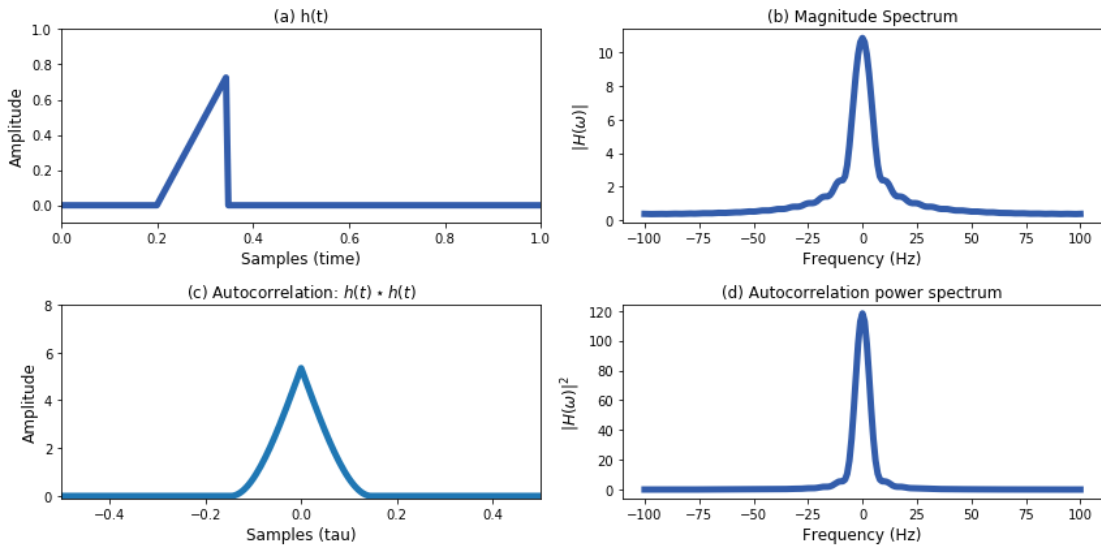


Figure 15. Illustration of the autocorrelation theorem. (a) Input signal $h(t)$. (b) Fourier magnitude spectrum of (a). (c) Autocorrelation of $h(t) \star h(t)$. (d) Autocorrelation power spectrum $|H(\omega)|^2$ of (c).

Deconvolution

I was also interested in the abilities of Scipy in performing deconvolution. Recall that if $y(t) = x(t) * h(t)$ that this operation performs the equivalent of

$$x(t) = \mathcal{F}^{-1} \left[\frac{\widehat{Y}(\omega)}{\widehat{H}(\omega)} \right]. \quad (58)$$

However, this does not compute it directly from the Fourier spectra. Rather, it aims to find a **polynomial expression** that adequately approximates $h(t)$ and then effectively uses polynomial division to remove this influence of $h(t)$ from $y(t)$.

This is somewhat analogous polynomial division that you may have done in earlier courses. For example, if $x(t) = t^2 + 1$ and $h(t) = 2t + 7$, then $y(t)$ is effectively the convolution of these two polynomials. Let these vectors represent the polynomial coefficients: $u = [1, 0, 1]$ and $v = [2, 7]$. The resulting **discrete convolution** of u and v is $[2, 7, 2, 7]$ or

$$y(t) = 2t^3 + 7t^2 + 2t + 7, \quad (59)$$

which is equivalent to the polynomial multiplication of the two functions $x(t)$ and $h(t)$. Let's verify this numerically:

```
u: [1, 0, 1]
v: [2, 7]
The convolution of u and v is: [2 7 2 7]
```

The **deconvolution** of $h(t)$ from $y(t)$ is then equivalent to **polynomial division**:

$$x(t) = \frac{2t^3 + 7t^2 + 2t + 7}{2t + 7} = \frac{(t^2 + 1)(2t + 7)}{2t + 7} = t^2 + 1. \quad (60)$$

This approach works great if you can find a **stable** polynomial approximation for $h(t)$.

```
The deconvolution of aa by u is: [ 2.  7.]
The remainder is: [ 0.  0.  0.  0.]
```

This doesn't always exist ... and we'll have to dig a bit deeper in a few weeks when discussing Z-transforms to figure out why. However, if we modify the above expression by changing the 7 into a 6 such that you can no longer factor out $2t + 7$

$$x(t) = \frac{2t^3 + 7t^2 + 2t + 6}{2t + 7}. \quad (61)$$

```
The deconvolution of anew by u is: [ 2.  7.]
The remainder is: [ 0.  0.  0. -1.]
```

Let's explore an example where convolution and then deconvolution is being applied using the following two function calls:

- `y = np.convolve(x,h,mode='same')`
- `x,remainder=scipy.signal.deconvolve(y,h)`

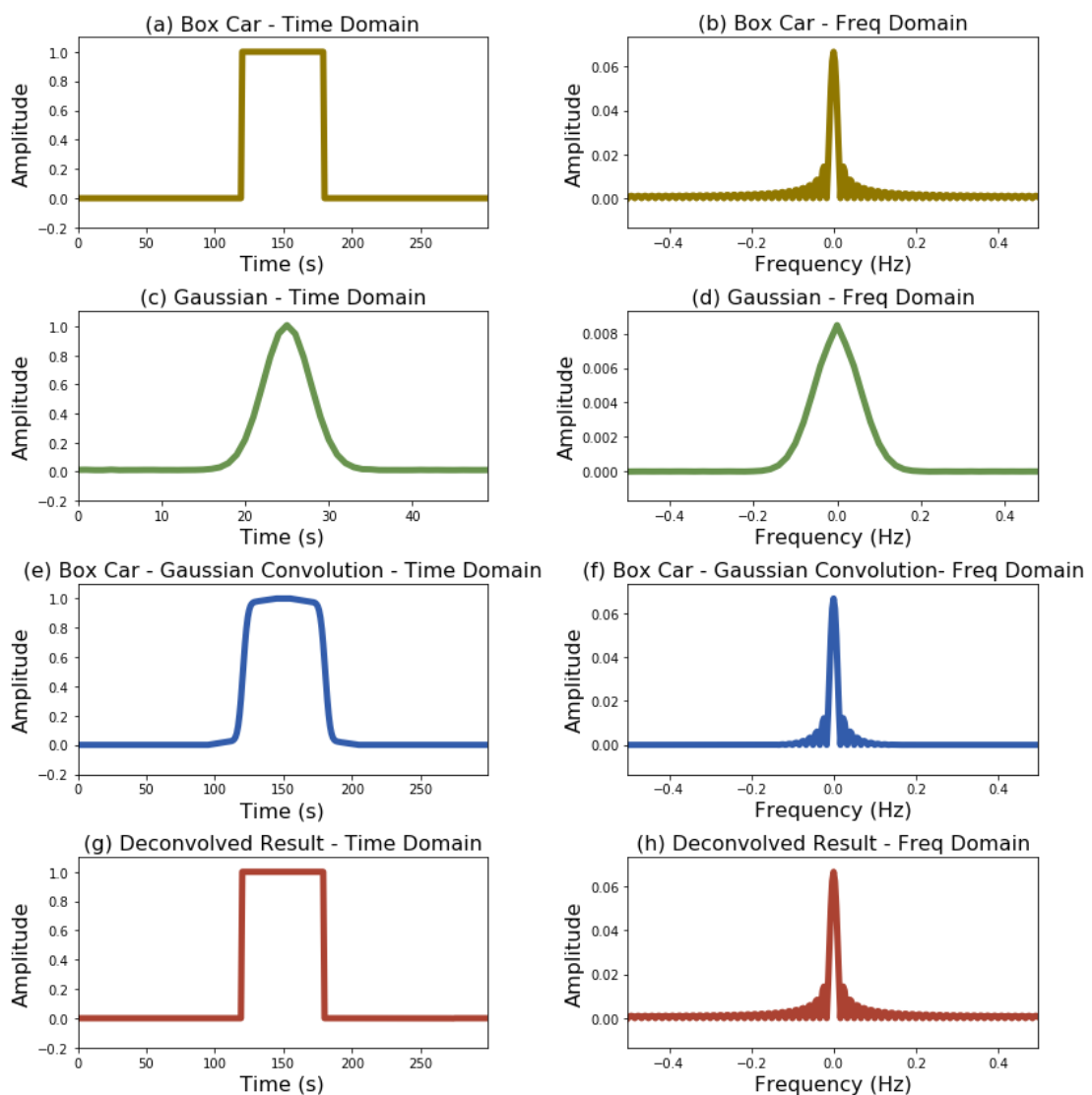


Figure 16. Illustration of the deconvolution process involving both time- (left panels) and frequency (right panels). (a) Original box car function and (b) associated Fourier magnitude spectrum. (c) Gaussian function and (d) the associated Fourier magnitude spectrum. (e) Convolution of box car in (a) and Gaussian in (c) along with (f) the Fourier magnitude spectrum of the convolved result. (g) The result of deconvolving the Gaussian in (c) from the convolved result in (e) along with (h) the associated deconvolved spectrum.

Overall, this process seems to have done a fairly stable result. However, if you play around with the parameters and, e.g., set $DC_term=0$ in the code section above, you'll see that things become unstable mighty quickly!

Deconvolution in practice

So far "decon" is an interesting theoretical operation ... but does it actually matter in practice? The answer is a resounding yes! Because we are using seismic waves to interrogate the Earth to learn more about its structure, it's very important that we attempt to remove as much of the wavelet effect as possible (which broadens the response and reduces the bandwidth (or difference between minimum and maximum amplitudes)). Let's look at two examples:

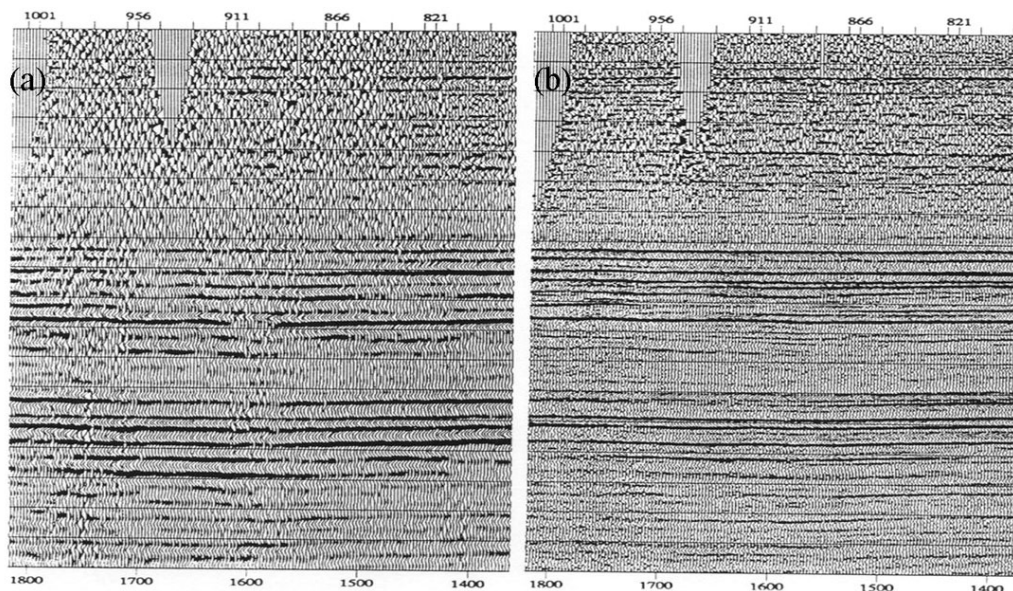


Figure 17. 2D seismic section showing the benefits of applying 2D deconvolution to land data. (a) Seismic image without applying deconvolution. (b) Same image as in (a) but after applying deconvolution. Note the improvement in resolution! (Credit: Cary, CSEG Recorder)

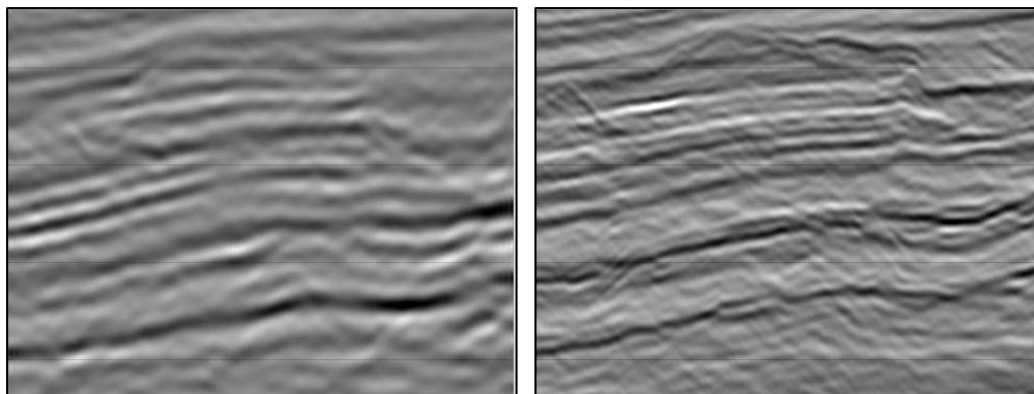


Figure 18. 2D seismic section showing the benefits of applying 2D deconvolution to marine data. (a) Seismic image without applying deconvolution. (b) Same image as in (a) but after applying deconvolution. Again, note the improvement in resolution! (Credit: Schlumberger)

References

1. Bracewell, R.N., 1965, The Fourier Transform and Applications, McGraw-Hill, New York.
2. James, J.F. 2011, A Student's Guide to Fourier Transforms, 3rd ed, Cambridge University Press.