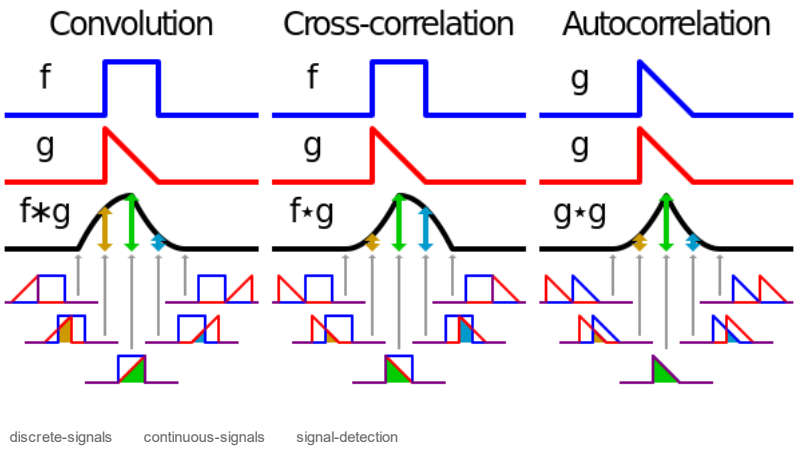
The difference between convolution and cross-correlation from a signal-analysis point of view

I am trying to understand the difference between convolution to cross-correlation. I have read an understood This answer. I also understand the picture below.

But, in terms of signal processing, (a field which I know little about..), Given two signals (or maybe a signal and a filter?), When will we use convolution and when will we prefer to use cross correlation, I mean, When in real life analysing will we prefer convolution, and when, cross-correlation.

It seems like these two terms has a lot of use, so, what is that use?



edited Jun 26 at 18:52 Community ♦ 1

asked Dec 2 '15 at 13:45 MathBgu 63 1 1 5

4 Answers

In signal processing, two problems are common:

- What is the output of this filter when its input is $x(t)$? The answer is given by $x(t) * h(t)$, where $h(t)$ is a signal called the "impulse response" of the filter, and $*$ is the convolution operation.
- Given a noisy signal $y(t)$, is the signal $x(t)$ somehow present in $y(t)$? In other words, is $y(t)$ of the form $x(t) + n(t)$, where $n(t)$ is noise? The answer can be found by the correlation of $y(t)$ and $x(t)$. If the correlation is large for a given time delay τ, then we may be confident in saying that the answer is yes.

Note that when the signals involved are symmetric, convolution and cross-correlation become the same operation; this case is also very common in some areas of DSP.

answered Dec 2 '15 at 14:13 MBaz 6,987 3 13 29

Got it. Thanks a lot for your clear and bright answer! – MathBgu Dec 2 '15 at 16:30

what I like about the impulse response explanation is you really get an intuition why convolution is "reversed". In discrete terms, the current output is the current input x impulse response at time 0 + residual output from previous inputs impulse responses (input a n-1 * impulse 1 + input n-2 * impulse 2 and so on). – Jean-Frederic PLANTE Aug 21 at 20:35

@Jean-FredericPLANTE yes, that's a good way to explain it. – MBaz Aug 21 at 21:44

The two terms **convolution** and **cross-correlation** are implemented in a very similar way in

DSP.

Which one you use depends on the application.

If you are performing a linear, time-invariant filtering operation, you *convolve* the signal with the system's impulse response.

If you are "measuring the similarity" between two signals, then you *cross-correlate* them.

The two terms come together when you try to produce a **matched filter**.

Here, you are trying to decide whether a given signal, $s[n]$ contains a known "pulse" (signal), $p[n]$. One way to do that is to convolve the given signal, s with the time-reversal of the known pulse, p: you are now using convolution to perform the cross-correlation of the given signal with the known pulse.

A Side Note

The term "cross-correlation" is (for some) misused in the field of DSP.

For statisticians, a correlation is a value that measures how close two variables are and should be between -1 and $+1$.

As you can see from the Wikipedia entry on cross-correlation, the DSP version is used and they state:

> cross-correlation is a measure of similarity of two series as a function of the lag of one relative to the other.

The problem with the DSP definition:

$$\sum_{\forall m} x[n]y[n+m]$$

is that this "similarity" measure depends upon the energy in each signal.

answered Dec 2 '15 at 14:11

Peter K. ♦
14.9k 8 27 51

1 This is extremely helpfull for me. Thank you! – MathBgu Dec 2 '15 at 16:30

In signal processing, the convolution is performed to obtain the output of an LTI system. The correlation (auto, or cross correlation) usually is calculated to be used later to do some other calculations

You have to be careful not to confuse correlation, covariance, and correlation coefficient. The correlation does not necessarily have to be between -1 and 1. The correlation coefficient (https://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient) falls between -1 and 1 because it is scaled by the two random variables variances. The thing we have to remember is that the real operation to be done in statistical signal processing to analize how related are two random variables is the "Covariance", not the correlation. But for most applications where a signal is captured by a sensor and transformed into a voltage and the digitized with an ADC, you can assume that the signal is zero mean, hence the correlation is equal to the covariance.

answered Dec 2 '15 at 14:56

bone
1,049 6 12

I will have a look in that link. Thank you! – MathBgu Dec 2 '15 at 16:31

@MathBgu I have read all above given answers, all are very informative one thing i want to add for your better understanding, by considering the formula of convolution as follows

$$f(x) * g(x) \quad = \quad \int_{-\infty}^{\infty} f(\tau) \cdot g(x-\tau)\, d\tau$$

and for the cross correlation

$$(f \star g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f^*(\tau)\ g(t+\tau)\, d\tau,$$

we comes to know that equation-wise the only difference is that, in convolution, before doing sliding dot product we flip the signal across y-axis i.e we changes (t) to (-t) , while the cross correlation is just the sliding dot product of two signals.

We use the convolution to get output/result of a system which have two blocks/signals and they are directly next to each other (in series) in the time domain.

edited Dec 27 '15 at 9:56

answered Dec 2 '15 at 22:54

RM Faheem
48 8

Thank you for mentioning thos additionsI clearifying point! – MathBgu Dec 16 '15 at 17:11