function val = gauss(x, sigma, xc)

exponent = ((x-xc).^2)./(2\*sigma);

val = (exp(-exponent));

endfunction

% Shift a Signal in the Frequency Domain.

% Number of points in the signal:

N = 128;

% Create a sample point array. It must be 0 to N-1 in steps of 1.

n=linspace(0,N-1,128);

% Generate a signal from the sample points.

g=gauss(n,20,30);

% The Fourier transform simply maps whatever number of points you give it

% to 0 to 2\*pi. It does not take into consideration signal sampling rate or any other

% external parameters. Whatever set of points you give it, it maps them to the unit circle.

% Remember that a Fourier transform provides positive and negative frequency components centered

% about the zero frequency, called the DC offset. For a real signal the negative frequency components

% are the complex conjugate of the positive frequencies. This is called conjugate symmetric.

% Generate the frequency index.

m = linspace(-N/2, (N/2 – 1), N);

% For N = 128, m will be -64 to 63.

% In the following 2\*pi / N will appear everywhere and often that number will appear as a constant

% such as w = 2\*pi / N.

% Let S be the number of data points to shift the signal. Given the external parameters that were used

% to generate the signal you can back out what a point shift means, but the (Discrete) Fourier transform

% doesn’t care. It works on points, samples, or whatever you might call them.

S = 10;

% S = 10 means a 10 point shift.

% To introduce the shift in the frequency domain multiply the Fourier transform of the signal by

% exp(-i\*(2\*pi / N)\*S\* m) . It might help to remember that exp(i\*f) = cos(f) + i \* sin(f) .

% Fourier transform the signal.

fg = fft(g);

% Shift the signal in the frequency domain.

fh = fg .\* exp(-i\*(2\*pi / N) \* S \* m);

% cos(2\*pi / N \* m) – i sin(2\*pi/N \* m)

% Inverse transform to see the signal back in the special domain. Note that we have to take the real

% part because even though the imaginary part will be very very small, the result will still be

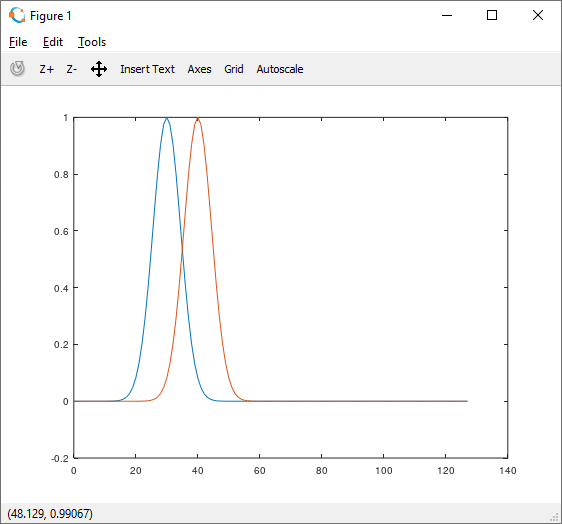
% a complex number and it will confuse the graphing package.

h=real( ifft(fh) );

plot(n,g)

hold on

plot(n,h)

[](https://i.stack.imgur.com/K0KVE.png)