Note that the Hilbert transform of a constant is zero. If you compute the Hilbert transform in more than one dimension and one of the dimensions does not vary (is a constant), the transform will be zero (or at least numerically close to zero).

There are various definitions for the extension of the Hilbert transform to more than one dimension. In two dimensions this operation computes the transform by multiplying the 2D Fourier transform of the input by the factor (-i)sgn(x)(-i)sgn(y) and then computing the inverse Fourier Transform. A similar procedure is used when the input is 3D.

Examples

Extract the instantaneous amplitude and frequency of a narrow-band signal:

```
Make/O/N=1000 w0,amp,phase
SetScale/I x 0,50,"", w0,amp,phase
w0 = exp(-x/10)*cos(2*pi*x)
HilbertTransform /DEST=w0h w0 // w0+i*w0h is the "analytic signal", i=cmplx(0,1)
amp = sqrt(w0^2 + w0h^2)   // extract the envelope
phase = atan2(-w0h,w0)     // extract the phase [SIGN CONVENTION?]
Unwrap 2*pi, phase         // eliminate the 2*pi phase jumps
Differentiate phase /D=freq   // would have less noise if fit to a line
                              // over interior points
freq /= 2*pi               // phase = 2*pi*freq*time
Display w0,amp   // original waveform and its envelope; note boundary effects
Display freq     // instantaneous frequency estimate, with boundary effects
```

See Also

The **FFT** operation.

References

Bracewell, R., *The Fourier Transform and Its Applications*, McGraw-Hill, 1965.

Compute the envelope of a signal:

```
Function calcEnvelope(Wave ddd)
HilbertTransform/dest=ht ddd
Matrixop/o sEnv=abs(cmplx(ddd,ht))
CopyScales ddd,sEnv
KillWaves/z ht
End
```

Histogram

Histogram [***flags***] ***srcWaveName, destWaveName***

The Histogram operation generates a histogram of the data in *srcWaveName* and puts the result in *destWaveName* or in W_Histogram or in the wave specified by /DEST.

Parameters

srcWaveName specifies the wave containing the data to be histogrammed.

For historical reasons the meaning and use of *destWaveName* depend on the binning mode as specified by /B. See **Histogram Destination Wave** on page V-351 below for details.

Flags

/A — Accumulates the histogram result with the existing values in the destination wave instead of replacing the existing values with the result. Assumes /B=2 unless the /B flag is present.

Note: The result will be incorrect if you also use /P.

/B=mode Controls binning:

mode=1: Semi-automatic mode that sets the bin range based on the range of the Y values in *srcWaveName*. The number of bins is determined by the number of points in the destination wave.

mode=2: Uses the bin range and number of bins determined by the X scaling and number of points in the destination wave.

mode=3: Uses Sturges' method to determine optimal number of bins and redimensions the destination wave as necessary. By this method

```
numBins=1+log2(N)
```

where N is the number of data points in *srcWaveName*. The bins will be distributed so that they include the minimum and maximum values.

mode=4: Uses a method due to Scott, which determines the optimal bin width as

$$binWidth=3.49*\sigma*N^{-1/3}$$

where N is the number of data points in *srcWaveName* and σ is the standard deviation of the distribution. The bins will be distributed so that they include the minimum and maximum values.

method=5: Uses the Freedman-Diaconis method where

$$binWidth=2*IQR*N^{-1/3}$$

where IQR is the interquartile distance (see **StatsQuantiles**) and the bins are evenly distributed between the minimum and maximum values.

/B={*binStart*,*binWidth*,*numBins*}

Sets the histogram bins from these parameters rather than from *destWaveName*. Changes the X scaling and length of the destination wave.

/C Sets the X scaling so that X values are in the centers of the bins, which is required when you do a curve fit to the histogram output. Ordinarily, wave scaling of the output wave is set with X values at the left bin edges.

/CUM Requests a cumulative histogram in which each bin is the sum of bins to the left. The last bin will contain the total number of input data points, or, with /P, 1.0.

/CUM cannot be used with a weighted histogram (/W flag).

When used with /A, the destination wave must be the result of a histogram created with /CUM.

Note that if you use a binning mode (/B flag) that sets a bin range that does not include the entire range of the input data, then the output will not count all of input points and the last bin will not contain the total number of input points. Input points whose values fall below the left edge of the first bin or above the right edge of the last bin will not be counted.

/DEST=*destWave* Saves the histogram output in a wave specified by *destWave*. The destination wave is created or overwritten if it already exists.

Creates a wave reference for the destination wave in a user function. See **Automatic Creation of WAVE References** on page IV-72 for details.

See **Histogram Destination Wave** on page V-351 below for further discussion.

The /DEST flag was added in Igor Pro 7.00.

/DP Causes Histogram to create the destination wave as double-precision floating point instead of single-precision floating point. The /DP flag was added in Igor Pro 8.00.

Single-precision precisely represents integers up to 16,677,721 only. Double-precision precisely represents integers up to about 9 trillion.

/N — Creates a wave named W_SqrtN containing the square root of the number of counts in each bin. This is an appropriate wave to use as a weighting wave when doing a curve fit to the histogram results. /N cannot be used with a weighted histogram (/W flag).

/NLIN=*binsWave* — Computes a non-linear histogram using the bins specified in the wave *binsWave*. This option is not compatible with the flags /A, /B, /C, /CUM, /N, /P, /W.

The bins must be contiguous and non-overlapping so that *binsWave* contains monotonically increasing values with no NaNs and INFs. For example, if you want the 3 bins [1,10),[10,100),[100,1000), execute:

Make/O/N=4 *bins*={1,10,100,1000}

The upper end of each bin is open.

The /NLIN flag was added in Igor Pro 7.00.

/P — Normalizes the histogram as a probability distribution function, and shifts wave scaling so that data correspond to the bin centers.

When using the results with **Integrate**, you must use /METH=0 or /METH=2 to select rectangular integration methods.

/R=(*startX*,*endX*) — Specifies the range of X values of *srcWaveName* over which the histogram is to be computed.

/R=[*startP*,*endP*] — Specifies the range of points of *srcWaveName* over which the histogram is to be computed.

/RMD=[*firstRow*,*lastRow*][*firstColumn*,*lastColumn*][*firstLayer*,*lastlayer*][*firstChunk*,*lastChunk*]

Designates a contiguous range of data in the source wave to which the operation is to be applied. This flag was added in Igor Pro 7.00.

You can include all higher dimensions by leaving off the corresponding brackets. For example:

```
/RMD=[firstRow,lastRow]
```

includes all available columns, layers and chunks.

You can use empty brackets to include all of a given dimension. For example:

```
/RMD=[][firstColumn,lastColumn]
```

means "all rows from column A to column B".

You can use a * to specify the end of any dimension. For example:

```
/RMD=[firstRow,*]
```

means "from firstRow through the last row".

/W=*weightWave* — Creates a “weighted” histogram. In this case, instead of adding a single count to the appropriate bin, the corresponding value from *weightWave* is added to the bin. *weightWave* may be any number type, and it may be complex. If it is complex, then the destination wave will be complex.

/W cannot be used with a cumulative histogram (/CUM flag).

Histogram Destination Wave

For historical reasons there are multiple ways to specify the destination wave and the meaning and use of *destWaveName* depend on the binning mode as specified by /B. This section explains the details and then provides guidance and when to use which mode.

In binning modes 1 and 2 (/B=1 and /B=2, described above), the destination wave plays a role in determining the binning and *destWaveName* must be the name of an existing wave. If you omit /DEST then the output is written to *destWaveName*. If you provide /DEST then the output is written to the wave specified by /DEST.

In binning modes 3, 4 and 5 (/B=3, /B=4 and /B=5, described above), the destination wave plays no role in determining the binning. If you omit *destWaveName* and /DEST, Histogram stores its output in a wave named W_Histogram in the current data folder. If you omit /DEST and provide *destWaveName*, then

destWaveName must name an existing wave to which the output is written. If you provide /DEST, you can omit *destWaveName*. If you provide both /DEST and *destWaveName* then *destWaveName* must name an existing wave but the operation ignores it.

Here is the recommended usage:

If you want to use specific binning that you have determined, use /B={*binStart*,*binWidth*,*numBins*}, use /DEST to specify the destination wave, and omit *destWaveName*.

If you want Igor to determine the binning, use /B=3, /B=4 or /B=5, use /DEST to specify the destination wave, and omit *destWaveName*.

For backward compatibility with Igor Pro 6, use /B=1, /B=2, /B=3, /B=4 or /B={*binStart*,*binWidth*,*numBins*}, create a destination wave, use it as *destWaveName* and omit /DEST.

Details

If you use /B={*binStart*, *binWidth*, *numBins*}, then the initial number of data points in the wave is immaterial since the Histogram operation changes the number of points.

Only one /B and only one /R flag is allowed.

If both /A and /B flags are missing, the bin range and number of bins is calculated as if /B=1 had been specified.

When accumulating multiple histograms in one output wave, typically you will want to use /B={*binStart*,*binWidth*,*numBins*} for the first histogram, and /A for successive histograms.

The Histogram operation works on single precision floating point destination waves. If necessary, Histogram redimensions the destination wave to be single precision floating point. However, Histogram/A requires that the destination wave already be single precision floating point.

For a weighted histogram, the destination wave will be double-precision.

If you specify the range as `/R=(start)`, then the end of the range is taken as the end of *srcWaveName*.

In an ordinary histogram, input data is examined one data point at a time. The operation determines which bin a data value falls into and a single count is added to that bin. A weighted histogram works similarly, except that it adds to the bin a value from another wave in which each row corresponds to the same row in the input wave.