Biosignals and Systems

Correlations and covariances

Advanced Measurements: Correlations and Covariances

- More complicated measurements can be made on a signal
  - by comparing it to other reference signals or mathematical functions.
- These comparisons are implemented through an operation known as
  - correlation.
- Correlation seeks to quantify how much one thing is like another.
- When comparing two mathematical functions (or signals),
  - the technique is to multiply one by the other,
  - then average the results.
  - This average is often scaled by some normalizing factor.

Correlation

The correlation between two signals, *x(t)* and *y(t)* over a time frame *T* is:

$$Corr = \frac{1}{T}\int_0^T x(t)\,y(t)\,dt$$

or in discrete form

$$Corr = \frac{1}{N}\sum_{k=1}^{N} x(k)\,y(k)$$

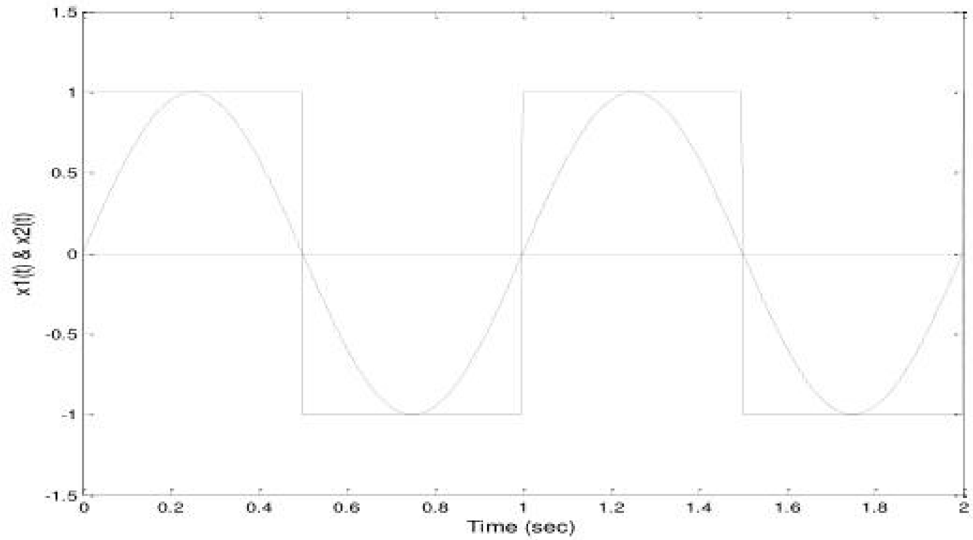
It is common to modify these equations by dividing by the square root of the product of the variances of the two signals.

This will make the correlation value equal to 1.0 when the two signals are identical and -1 if they are exact opposites

$$Corr_{normalized} = \frac{Corr}{\sqrt{\sigma_1^{\,2}\sigma_2^{\,2}}}$$

Example 2-6

Use the correlation equation to find the correlation (un-normalized) between the sine wave and the square wave shown.



$$Corr = \frac{1}{T}\int_0^T x(t)\,y(t)\,dt = \frac{1}{T/2}\int_0^{T/2} \sin\left(\frac{2\pi t}{T}\right)dt = \frac{2}{T}\frac{T}{2\pi}\left(-\cos\left(\frac{2\pi t}{T}\right)\right)\Bigg|_0^{T/2}$$

$$Corr = \frac{1}{\pi}\left(-\cos(\pi) - -\cos(0)\right) = \frac{2}{\pi}$$

Note that the correlation between a sine and cosine will be zero.

Covariance

- Covariance computes the variance that is shared between two (or more) signals.

- Specifically, the covariance is defied as:

$$\sigma_{xy} = \frac{1}{N-1}\sum_{k=1}^{k}(x_k - \overline{x})(y_k - \overline{y})$$

The equation for covariance is similar to the discrete form of correlation

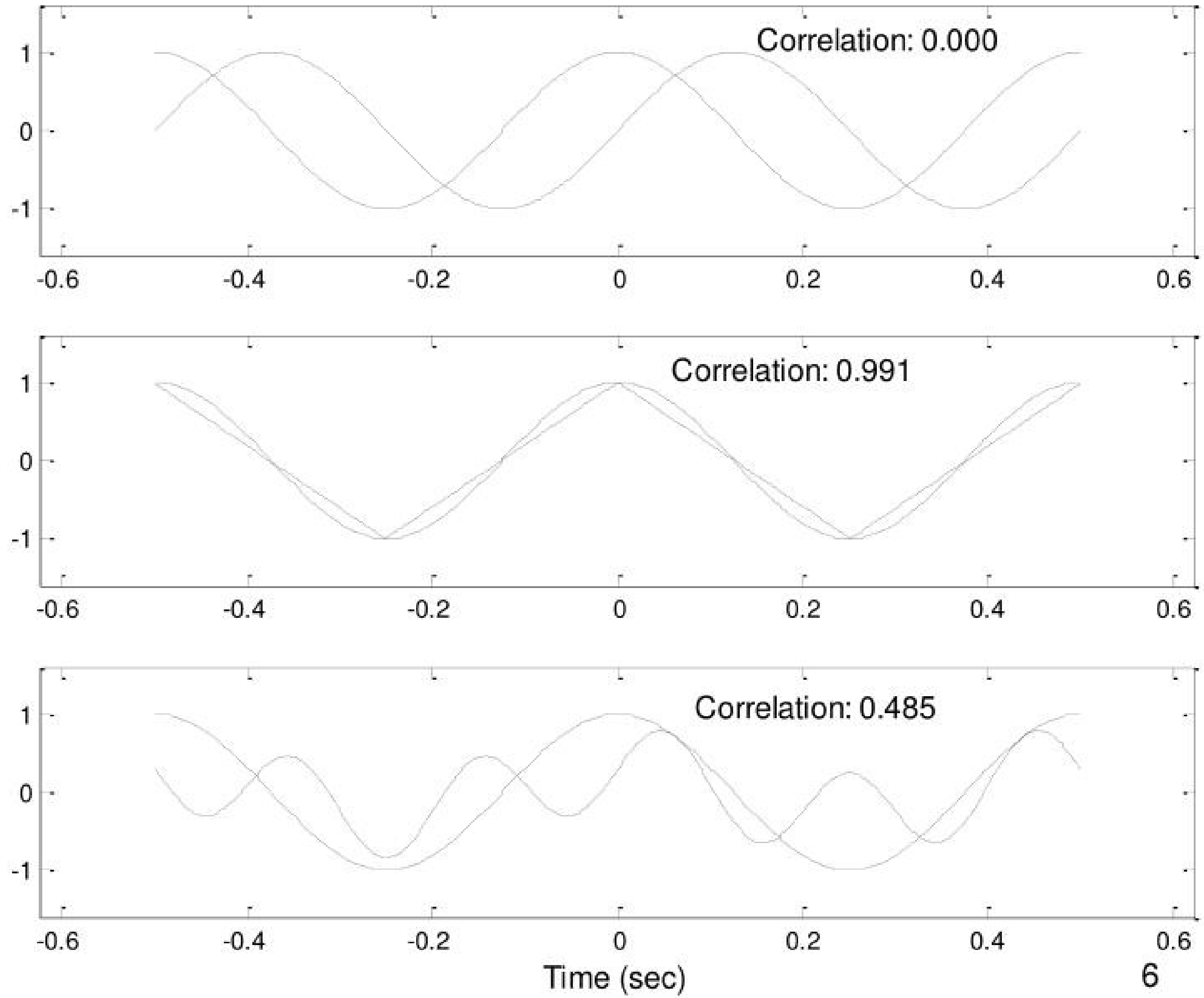
(except that the average values of the signals have been removed).

Correlation between different waveforms.

There is no correlation between a sine and cosine wave.

A high correlation is seen between a sine and a triangle.

A moderate correlation between a sine and a composite waveform.



Autocorrelation and Crosscorrelation

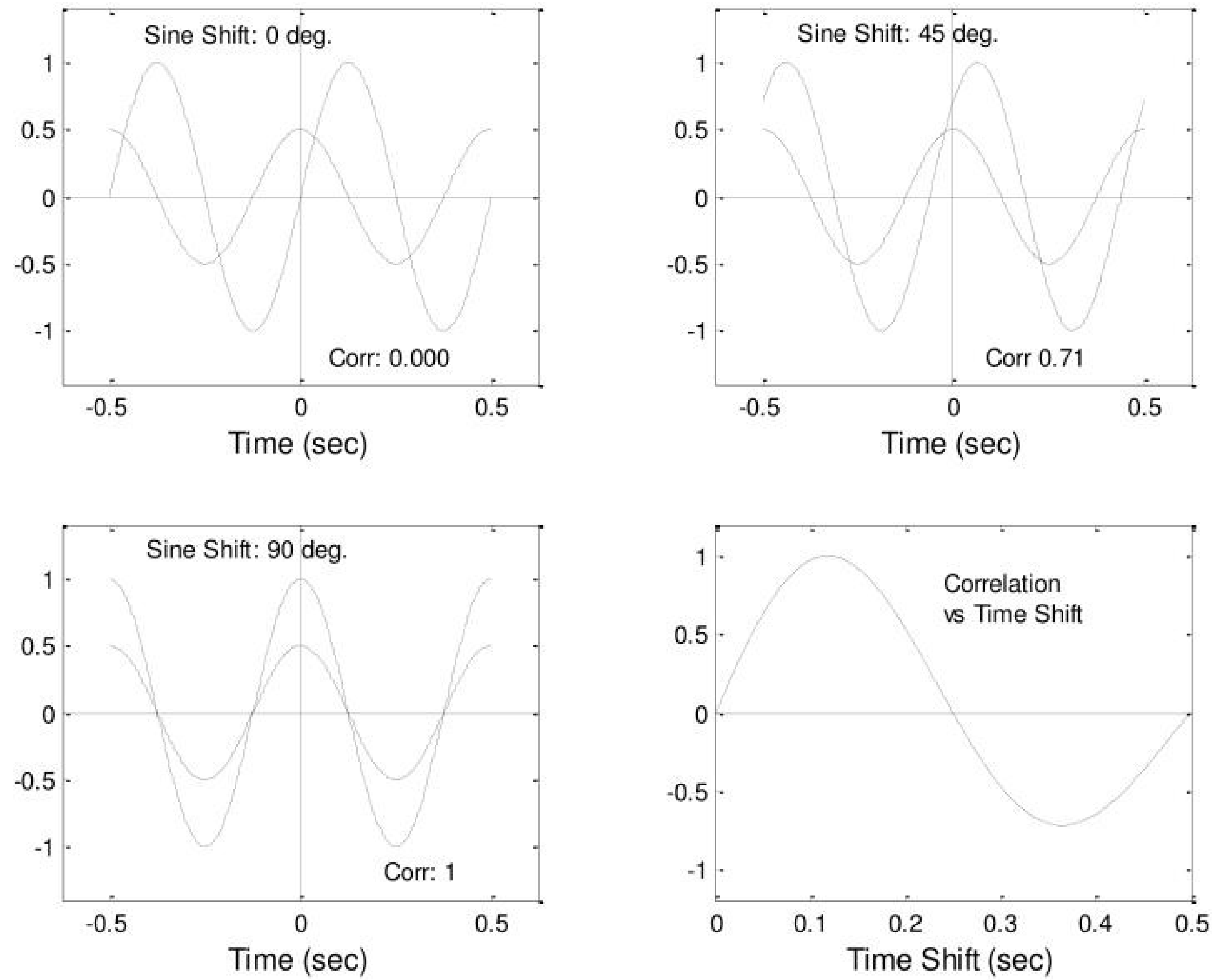
- The lack of mathematical correlation between a sine and a cosine can be a problem since they are intuitively similar even if they have zero correlation.
- A signal could be sinusoidal (e.g., a cosine), but if you are using a sine as a reference function the correlation would be small or negligible.
- To circumvent this problem, you could still use a sine reference for comparison, but shift this reference signal in time, performing the correlation for many different time shifts.
- Correlating over many different time shifts is called crosscorrelation.

Crosscorrelation

Shifting one sinusoid with respect to another probes all the possible relative positions.

The maximum correlation occurs when the two are in phase and is 1.0.

When the two are out of phase the correlation is —1.



Crosscorrelation

The shifting correlation in crosscorrelation can be achieved mathematically by introducing a variable time delay, or time lag, or simply "lag," into one of the two waveforms in the correlation equation.

(It does not matter which function is shifted)

$$Crosscorrelation \equiv r_{xy(\tau)} = \frac{1}{T}\int_0^T y(t)x(t+\tau)dt$$

where the variable τ is a continuous variable of time used to shift *x(t)* with respect to *y(t)*.

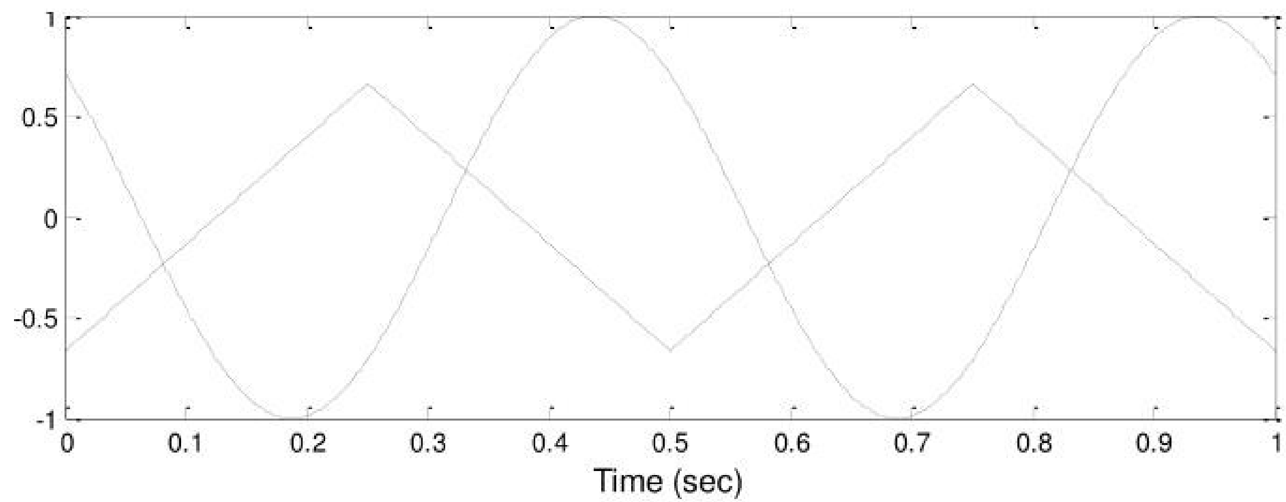
The variable τ is a variable of time, but not the time variable and is sometimes called a dummy time variable, although this is a bit misleading.

This variable is also called the "lag" or lag variable.

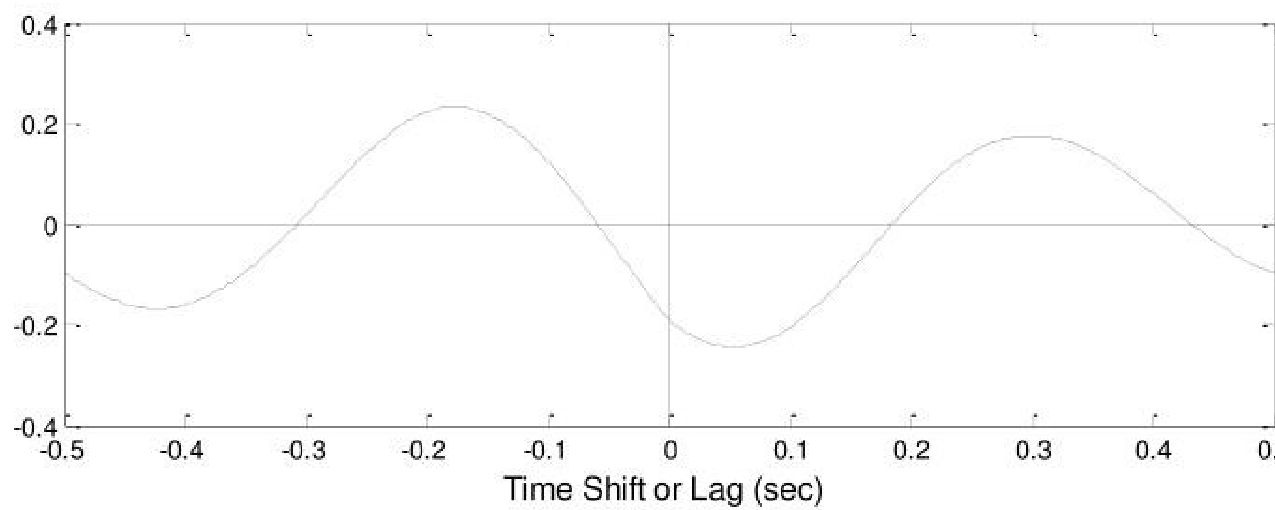
Note that the output of this equation is itself a waveform (i.e., function) of time, τ.

Crosscorrelation

Lower plot shows the cross-correlation function for the sinusoid and a triangle waveform given in the upper plot.



Note that they are most similar (i.e. have the highest correlation) when one signal is shifted 0.18 sec. with respect to the other.



Discrete Crosscorrelation

The discrete form of the crosscorrelation equation is constructed in the usual manner,

(replacing continuous variables with discrete variables and integration with summation.)

$$r_{xy}[n] = \frac{1}{N} \sum_{k=1}^{N} y(k)\,x(k+n)$$

Autocorrelation

- Autocorrelation is simply crosscorrelation of a waveform with itself.
- The autocorrelation equation in continuous and discrete forms becomes:

$$Autocorrelation \equiv r_{xx(\tau)} = \frac{1}{T}\int_0^T x(t)x(t+\tau)dt$$

$$r_{xx}[i] = \frac{1}{N}\sum_{k=1}^{N} x(k)x(k+i)$$

For the Direct Numerical Method via polynomial multiplication, use the following steps:

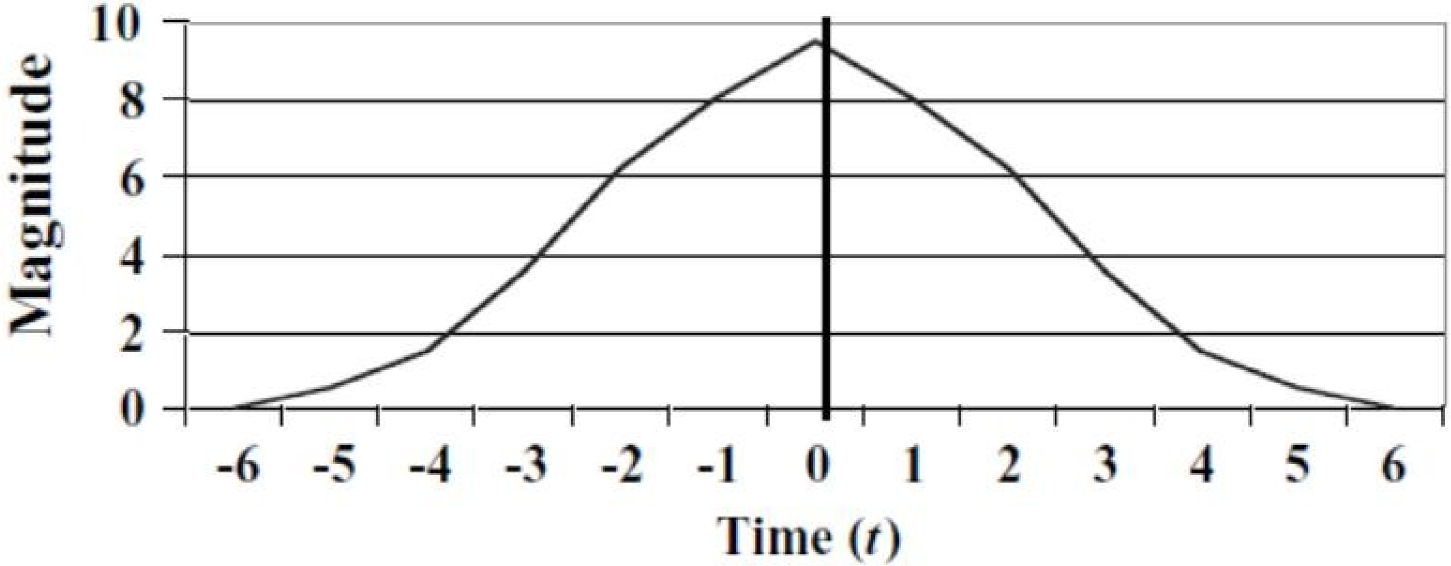
- directly tabulate values of the function in the second row with the increment of time above the function as in Table in the previous slide;
- reverse tabulate the same function in the third row;
- do normal polynomial multiplication (from right to left);
- write the products in columns following normal multiplication procedures;
- when there are no more products, sum the columns;

Autocorrelation by Polynomial Multiplication Method

t							0	1	2	3	4	5	6
x DT							1	1	2	1.5	1	0.5	0
x RT							0	0.5	1	1.5	2	1	1
							1	1	2	1.5	1	0.5	0
						1	1	2	1.5	1	0.5	0	
					2	2	4	3	2	1	0		
				1.5	1.5	3	2.3	1.5	0.8	0			
			1	1	2	1.5	1	0.5	0				
		0.5	0.5	1	0.8	0.5	0.3	0					
	0	0	0	0	0	0	0						
	0	0.5	1.5	3.5	6.3	8	9.5	8	6.3	3.5	1.5	0.5	0

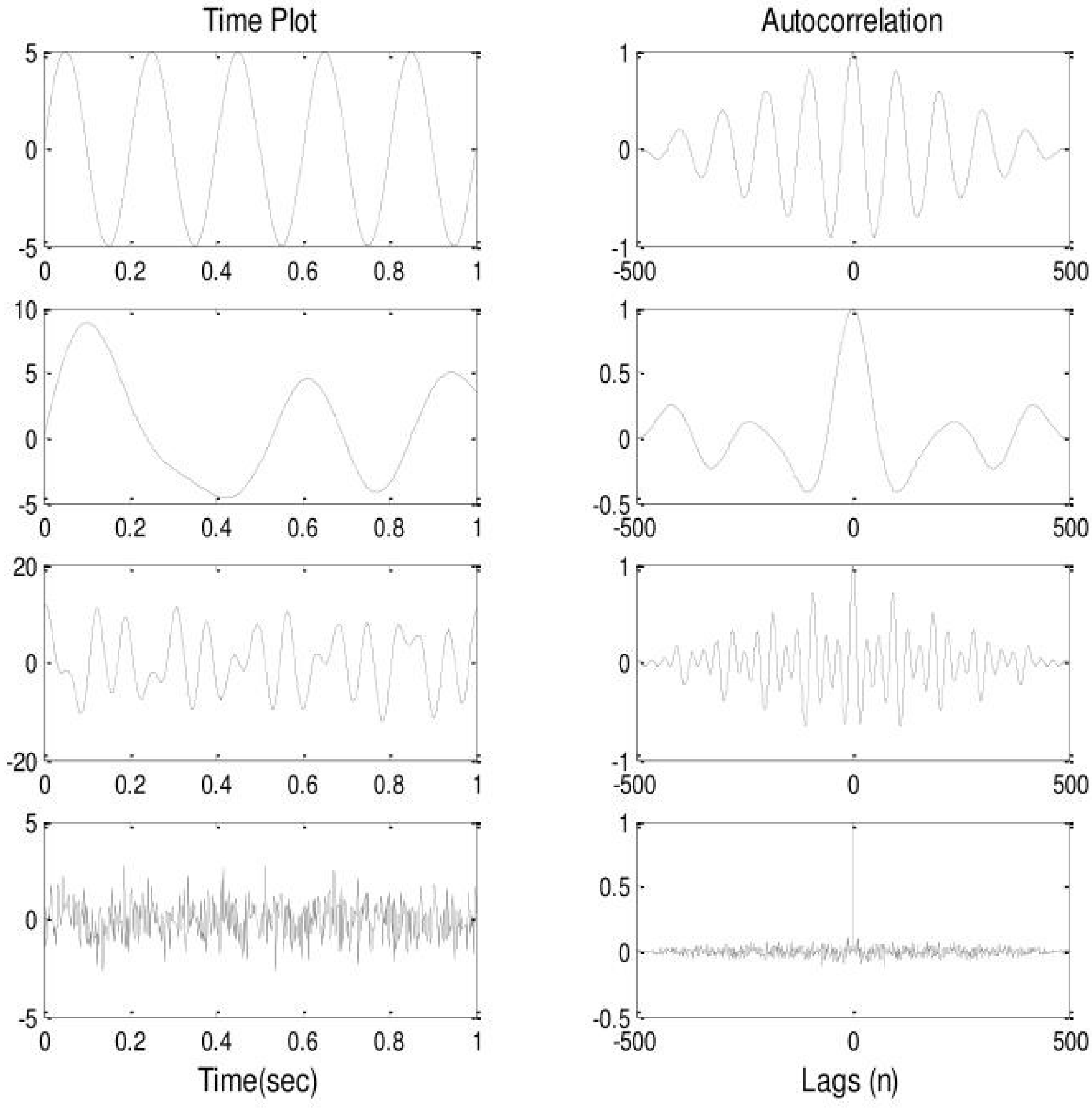
- Note: Original function (*x*); Time t ; DT is Direct Transcription;
- RT is Reverse Transcription.

Graph of the autocorrelation function



Note that the autocorrelation function is an even function

Four different signals (left side) and their autocorrelation functions (right side)



A) a sinusoid;

B) a slowly varying signal;

C) a rapidly varying signal;

D) a random signal.

MATLAB Implementation Covariance and Correlation

- Rxx = corrcoef(x); % Signal correlation
- S = cov(x); % Signal covariance

where x is a matrix that contains the various signals to be compared in columns.

The output, Rxx, of the 'corrcoef' routine will be a *n-by-n* matrix where *n* is the number of signals (ie., columns).
The diagonals of this matrix represent the correlation of the signals with themselves, Rxx (and, hence, will be 1), and the off diagonals represent the correlations of the various combinations.

The output of the 'cov' routine is similar except the entries contain variances

Correlation and Covariance

The output of the correlation routine 'corrcoef.'

$$R_{xx} = \begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,N} \\ r_{2,1} & r_{2,2} & \cdots & r_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ r_{N,1} & r_{N,2} & \cdots & r_{N,N} \end{bmatrix}$$

The output of the covaraince routine 'cov.'

$$S = \begin{bmatrix} \sigma_{1,1} & \sigma_{1,2} & \cdots & \sigma_{1,N} \\ \sigma_{2,1} & \sigma_{2,2} & \cdots & \sigma_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{N,1} & \sigma_{N,2} & \cdots & \sigma_{N,N} \end{bmatrix}$$

Example 2-9

Determine if a sinewave and cosinewave at the same frequency are orthogonal and if sinewaves at harmonically related frequencies are orthogonal.
Include one sinusoid at a non-harmonic frequency.

Solution:
If two signals are orthogonal they will be uncorrelated.
Generate a data matrix where the columns consist of a 1.0 Hz sine and cosine, a 2.0 Hz sine and cosine, and a 3.0 Hz sine and a 3.5 Hz cosine.
The six sinusoids should all be at different amplitudes.
Apply the covariance (cov) and correlation (corrcoef) MATLAB functions.
All of the sinusoids except the 3.5 cosine are orthogonal and should show negligible correlation and covariance.

```
% Example 2.9: Application of the correlation and
% covariance matrices to sinusoids that are orthogonal and
% nonorthogonal
% Generate the sinusoids as columns of the matrix
clear all; close all;
N = 256;
fs = 256;
n = (1:N)/fs;
% Number of points in waveform
% Assumed sample frequency
% Time vector: 1 sec of data
x(:,1) = sin(2*pi*n)';              % Generate a 1 Hz sin
x(:,2) = 2*cos(2*pi*n)';            % Generate a 1 Hx cos
x(:,3) = 1.5*sin(4*pi*n)';          % Generate a 2 Hz sin
x(:,4) = 3*cos(4*pi*n)';            % Generate a 2 Hx cos
x(:,5) = 2.5*sin(6*pi*n)';          % Generate a 3 Hx sin
x(:,6) = 1.75*cos(7*pi*n)';         % Generate a 3.5 Hz cos
S = cov(x) ;                        % Print covariance matrix
Rxx = corrcoef(x) ;                 %   and correlation matrix
```

Results:

The output from this program is a covariance and correlation matrix. The covariance matrix is:

S =

0.5020	0.0000	0.0000	0.0000	0.0000	-0.0497
0.0000	2.0078	-0.0000	-0.0000	-0.0000	-0.0137
0.0000	-0.0000	1.1294	0.0000	-0.0000	-0.2034
0.0000	-0.0000	0.0000	4.5176	-0.0000	-0.0206
0.0000	-0.0000	-0.0000	-0.0000	3.1373	-1.2907
-0.0497	-0.0137	-0.2034	-0.0206	-1.2907	1.5372

The diagonals of the covariance matrix give the variance of the six signals and these differ since the amplitudes of the signals are different. The correlation matrix shows similar results except that the diagonals are now 1.0 since these reflect the correlation of the signal with itself.

Rxx =

1.0000	0.0000	0.0000	0.0000	0.0000	-0.0566
0.0000	1.0000	-0.0000	-0.0000	-0.0000	-0.0078
0.0000	-0.0000	1.0000	0.0000	-0.0000	-0.1544
0.0000	-0.0000	0.0000	1.0000	-0.0000	-0.0078
0.0000	-0.0000	-0.0000	-0.0000	1.0000	-0.5878
-0.0566	-0.0078	-0.1544	-0.0078	-0.5878	1.0000

MATLAB Implementation
Autocorrelation and Crosscorrelation

- The crosscorrelation and autocorrelation operations are both performed with the same MATLAB routine, with autocorrelation being treated as a special case:

- `[r,lags] = xcorr(x,y,maxlags,'options');`

- Only the first input argument, x, is required.
- If no y variable is specified, autocorrelation is performed.
- The optional argument maxlags limits the shifting range.
- The shifted waveform is shifted between ±maxlags, or the default value which is *-N+1* to *N-1* where *N* is length of the input vector, x.

MATLAB Implementation
Auto- and Crosscovariance

- Autocovariance or crosscovariance is obtained using the ‘xcov’ function:
- [c,lags] = xcov(x,y,maxlags,’options’)
- The arguments are identical to those described above for the ‘xcorr’ function.
- Auto- and crosscovariance are the same as auto- and crosscorrelation if the data have zero means.

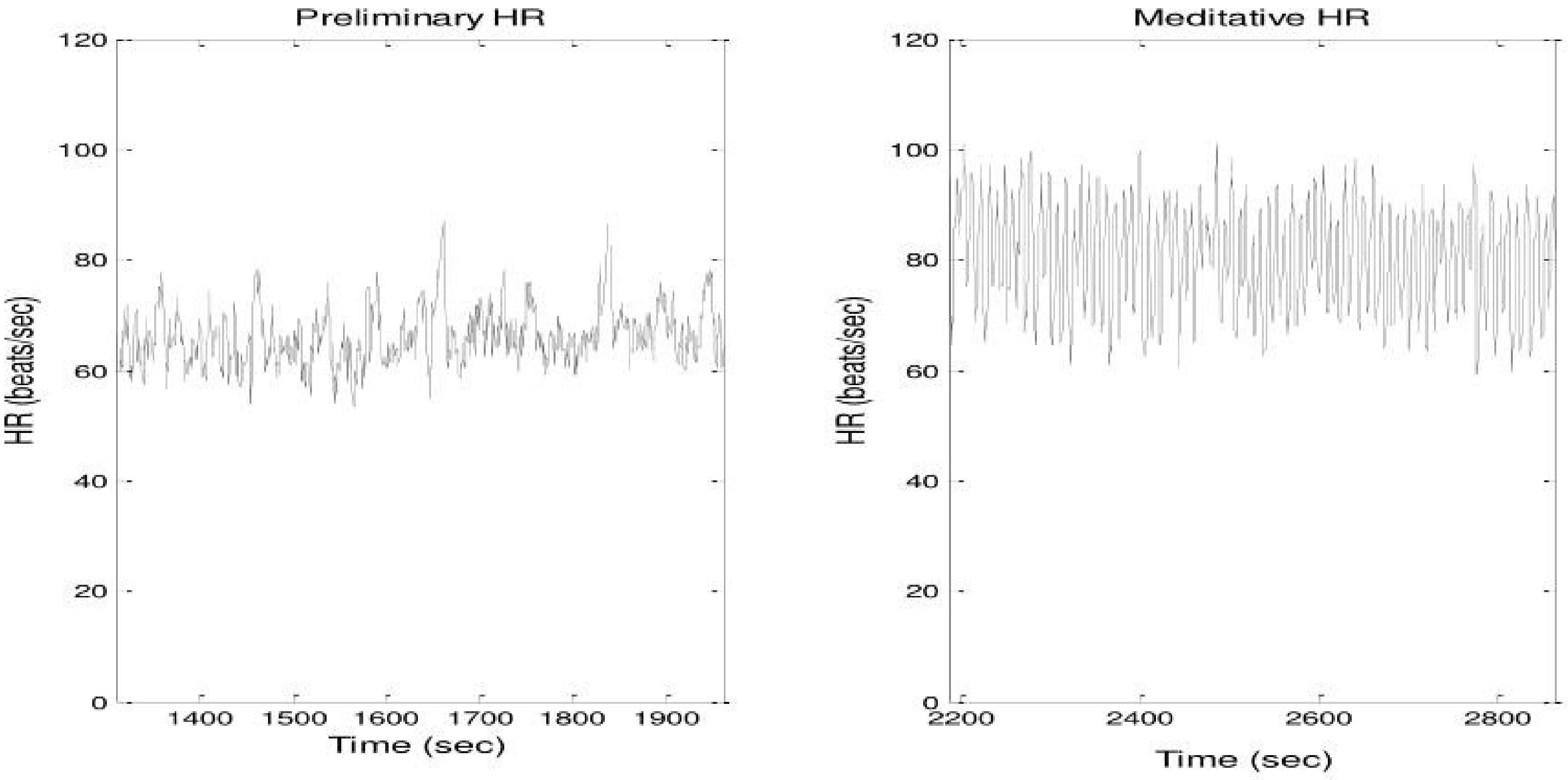
Example 2-10

Determine if there is any correlation in the variation between the timing of successive heart beats from the heart rate data shown below.

Solution

Load the heart rate data taken during normal resting conditions (file Hr_pre.txt).
Isolate the heart rate variable (the 2nd column) then take the autocovariance function.
The aotocovariance function will remove the mean HR giving only the change in HR.
Plot this function in such a way as to show potential correlation over approximately 30 successive beats

Example 2-10

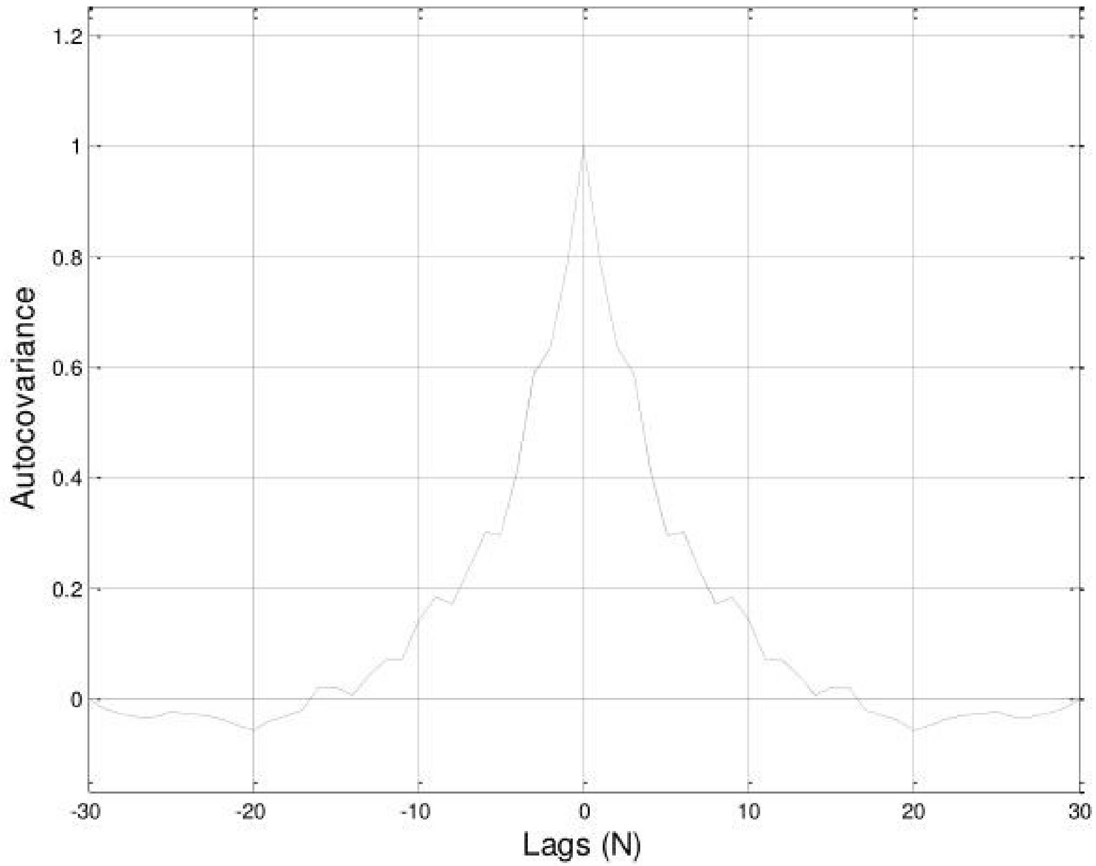


Heat rate data (beats/min against time) used in this example. Only the baseline, pre-meditative data (left plot) is used here.

Example 2-10

```
% EXAMPLE 2.10 and Figure 2.15
% Use of autocovariance to determine the correlation
% of heart rate variation between heart beats
%
clear all; close all;
figure ;
%load Hr_pre % Load data
load zf.mat % Load data
%[c,lags]=axcor(hr_pre-mean(hr_pre)); % Autocovariance (mean
subtracted)
[c,lags]=xcorr(zf-mean(zf)); % Autocovariance (mean subtracted)
plot(lags,c,'k'); hold on; % Plot autocovariance
%plot ( [lags (1) lags (end)], [0 0],'k') % Plot zero line for
% reference
xlabel('Lags (N) ' ) ; ylabel('Autocovariance') ; grid on;
%axis([-30 30-.21.2]); % Limit plot range
% to ± 30 beats
```

Example 2-10



Autocovariance function of the heart rate from one subject under normal resting conditions.
Some correlation is observed over approximately 10 successive heart beats.