5 Analysis of Coupled Rhythms in an Invasive Electrocorticogram

Synopsis

Data 1 s of ECoG data sampled at 500 Hz from two electrodes for 100 trials.

Goal Characterize the coupling of rhythmic activity between the two electrodes.

Tools Fourier transform, spectrum, amplitude, coherence, phase.

5.1 Introduction

5.1.1 Background

In chapters 3 and 4, we focused on field data recorded from a single electrode at the scalp (EEG) or cortical (ECoG) surface. However, typical brain voltage recordings consist of multiple electrodes. For example, the standard EEG recording consists of 21 electrodes spaced across the scalp surface, and sometimes many more [5]. The number of electrodes utilized in invasive ECoG recordings also range from a handful of contacts to over 100 implanted electrodes [10]. In this chapter, we continue our study of field data recorded from the cortical surface but now consider ECoG data recorded simultaneously from two electrodes during a task.

5.1.2 Case Study Data

We consider again the patient with epilepsy described in chapter 4. As part of her routine clinical workup before resective surgery, numerous electrodes were implanted directly on the cortical surface. The purpose of this invasive recording procedure was to monitor and localize her seizures for eventual surgical treatment. During this recording procedure, in which ECoG electrodes were implanted and recordings performed for one week, the patient volunteered to participate in an auditory task study administered by a collaborating researcher. The task required the patient to listen to individual phonemes through headphones and respond with a button click whenever she heard the phoneme "ba" (the other phonemes were different, e.g., "pa," "ma"). The tone presentation was repeated 100 times,

and her ECoG recorded (sampling rate 500 Hz) from two cortical electrodes over the auditory brain area for 1 s.

5.1.3 Goal

Our goal is to understand the coupling between the voltage activity recorded from two brain areas during the auditory task. To do so, we compute the *cross-covariance* and *coherence* between the two electrodes. These coupling measures build upon our previous development of the autocovariance, Fourier transform, and spectrum.

5.1.4 Tools

In this chapter, we develop the cross-covariance and coherence measures. For the latter, we continue to explore and understand the Fourier transform and examine in detail the notion of phase. We also briefly discuss strategies to assess the coherence for a single trial of data.

5.2 Data Analysis

5.2.1 Visual Inspection

To access the data for this chapter, visit

`http://github.com/Mark-Kramer/Case-Studies-Kramer-Eden`

and download the file `Ch5-ECoG-1.mat`. We begin our analysis by visualizing the ECoG data; load the ECoG data into MATLAB and plot the data from the first electrode (variable `E1`) and second electrode (variable `E2`) versus time (variable `t`) for the first trial:

```
load('Ch5-ECoG-1.mat')  %Load the ECoG data.
%... and plot one trial from each electrode.
plot(t,E1(1,:), 'b', 'LineWidth', 2)
hold on
plot(t,E2(1,:), 'r', 'LineWidth', 2)
hold off
```

The results for this trial and three others are plotted in figure 5.1. Visual inspection immediately suggests a dominant rhythmic activity in each trial.

Q: Approximate the dominant rhythmic activity in each electrode and trial by visual inspection of figure 5.1. A simple procedure is to count the number of peaks in each signal, then divide by the total length of the recording (in this case, 1 s). Does each electrode/trial exhibit approximately the same rhythms?

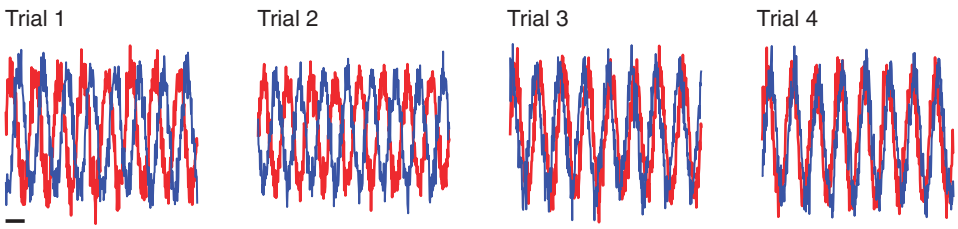

Figure 5.1
Traces of ECoG data recorded at two electrodes (*blue*, *red*) in four trials. Scale bar indicates 100 ms.

These techniques allow us to visualize the data one trial at a time. Doing so is often useful but can be time consuming, especially as the number of trials increases. Here we have 100 trials, and to visualize all of them in this way would require 100 plots. That's not so bad, but there's a better way. We can display the entire structure of the data across both time and trials as an *image*:

```
ntrials = size(E1,1);        %Define the number of trials,
imagesc(t,(1:ntrials),E1);   %... image all data,
xlabel('Time [s]')           %... and label the axes.
ylabel('Trial #')
```

The resulting image for the first electrode is shown in figure 5.2. Voltage is plotted as a function of time along the horizontal axis and trial number along the vertical axis. This allows us to visualize the voltage activity of the first electrode for all trials at once. We notice that each trial exhibits rhythmic structure, which manifests in figure 5.2 as repeating undulations of blue (low voltage), then red (high voltage) over time. We also observe variability in the alignment of these rhythms from trial to trial; from one trial to the next, the undulations appear not to align.

Q: Display an image of the activity for the second electrode and compare it to the image from the first electrode in figure 5.2. How do the two compare?

Visual inspection of the ECoG data allows us to draw some preliminary conclusions. First, the data appear to be rhythmic, with a particularly strong oscillation near 8 Hz. That's interesting but not the primary research objective. We would really like to understand whether the activity at the two electrodes is related. Many techniques exist to approach this problem [12], but let's begin with the most basic: visual inspection. By examining figure 5.1 we can attempt to deduce whether a consistent relation exists between the two ECoG signals

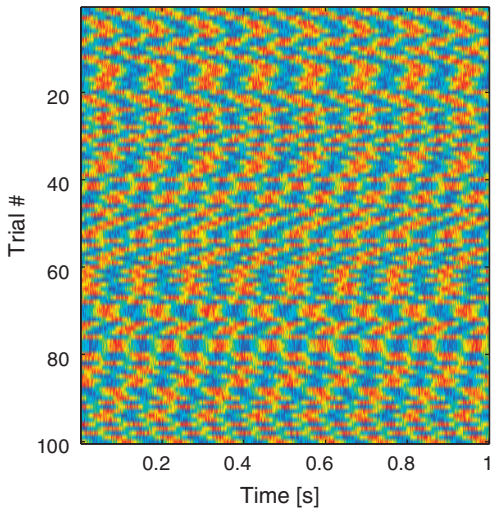

Figure 5.2
Image of ECoG data from first electrode.

across trials. We notice in the first two trials that the ECoG activity from the two electrodes appears nearly out of phase (i.e., when the blue curve is near a peak, the red curve is near a trough). However, for the next two trials, activity from the two electrodes nearly overlaps. From this initial visual inspection of four trials, it's difficult to conclude whether the ECoG activity at the two electrodes is interrelated; both electrodes display rhythmic activity across all trials, but the relation between these rhythms appears to change across trials: sometimes the activities overlap, and sometimes not.

Q: Repeat this analysis by examining additional trials, and by inspecting the activity images for each electrode. What conclusions can you make about the relations between the ECoG activity at the two electrodes? Are they related? Are they not related?

Although visual inspection is a useful initial tool for analyzing data, assessing the relations between two electrodes across multiple trials is a difficult task. To go further, we employ a new data analysis tool that builds from our experience with the Fourier transform: the coherence.

5.2.2 Autocovariance and Cross-covariance

In chapter 3, we defined and applied the autocovariance to a single time series, equation (3.3), and found that this measure helped reveal dependent structure in the data. We could,

of course, apply the autocovariance to each ECoG time series considered here. Let's do so, with a small update to the autocovariance formula that utilizes the trial structure of these data. We define the *trial-averaged autocovariance*[1] as

$$r_{xx}[L] = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{N}\sum_{n=1}^{N-L}(x_{n+L,k} - \bar{x}_k)(x_{n,k} - \bar{x}_k)\,, \tag{5.1}$$

where $x_{n,k}$ indicates the data at time index n and trial k, and $\bar{x}_k$ is the mean value of x for trial k. Notice that we include a new term, $\frac{1}{K}\sum_{k=1}^{K}$, which instructs us to sum over all trials the autocovariance computed for each trial and then divide by the total number of trials K. To compute and display the trial-averaged autocovariance for the first electrode in MATLAB,

```
load('Ch5-ECoG-1.mat')  %Load the ECoG data.
dt = t(2)-t(1);         %Define the sampling interval.
K = size(E1,1);         %Define the no. of trials.
nlags = 100;            %Define the max no. of +/- lags.
ac = zeros(1,2*nlags+1);%Declare empty vector for autocov.

for k=1:K               %For each trial,
 x = E1(k,:)-mean(E1(k,:));         %...subtract the mean,
 [ac0,lags]=xcorr(x,100,'biased');  %... compute autocovar,
 ac = ac + ac0/K;       %...and add to total, scaled by 1/K.
end
plot(lags*dt,ac)        %Plot autocovar vs lags in time.
xlabel('Lag [s]')       %Label the axes.
ylabel('Autocovariance');
```

Q: In using the function `xcorr`, we set the third input to `'biased'`. Why do we compute the biased autocovariance? *Hint:* See chapter 3 for a detailed discussion.

Q: Consider the results for the trial-averaged autocovariance plotted in figure 5.3. What do these results suggest about the rhythmic structure in these data?

A: The trial-averaged autocovariance in figure 5.3 helps characterize the rhythmic activity at this electrode. Visual inspection of this figure reveals three large positive peaks. The largest peak occurs at a lag of 0 s, as expected; the signal matches itself

1. We could instead write the trial-averaged *sample* autocovariance because this equation uses the observed data to estimate the theoretical covariance that we would see if we kept repeating this experiment. However, this distinction is not essential to the discussion here.

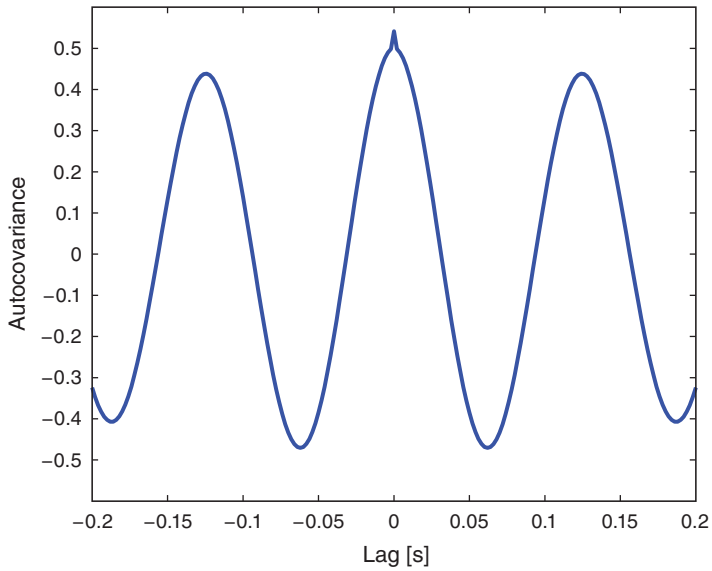

Figure 5.3
Trial-averaged autocovariance of ECoG data recorded at one electrode.

at zero lag. The two other peaks occur at lags of approximately ± 0.125 s. These peaks reveal that the data, and a version of the data shifted by $+0.125$ s or -0.125 s, are a good match. Notice that a shift of ± 0.125 s is consistent with periodic activity of approximate frequency $1/(0.125\,\text{s}) = 8$ Hz. For example, imagine a sinusoid of frequency 8 Hz; if we shift the sinusoid by its period (0.125 s) and compare it to the original (unshifted) sinusoid, the match will be excellent. Our data are more complicated than a simple sinusoid, but our visual inspection of the voltage traces (figure 5.1) did reveal a dominant 8 Hz rhythm consistent with these autocovariance results.

Q: Repeat the analysis to compute the trial-averaged autocovariance for the second electrode. What do you find? How do the trial-averaged autocovariances for the two electrodes compare?

The trial-averaged autocovariance results for each electrode are interesting, but our primary scientific question for these data is whether dependent structure exists *between* the ECoG activity recorded from the two electrodes. In other words, are the time series recorded from the two electrodes coupled? Many tools exist to characterize coupling between time series,

and in this chapter we focus on two such tools. The first is the cross-covariance, $r_{xy}[L]$, an extension of the autocovariance to include two time series, defined as,

$$r_{xy}[L] = \frac{1}{N}\sum_{n=1}^{N-L}(x_{n+L} - \bar{x})(y_n - \bar{y}), \tag{5.2}$$

where x and y are two time series with time index n. Notice what we've done; compared to the autocovariance, defined in (3.3), the cross-covariance formula simply replaces the x's in the second term in parentheses with y's.

The intuition for understanding the cross-covariance is similar to that for the autocovariance (see chapter 3). To calculate the cross-covariance, we multiply y with x shifted in time by lag L (figure 5.4). The cross-covariance is large at lag L if the two shifted time series x and y match. If we're interested in determining the coupling between x and y, finding these matches could be particularly useful. To illustrate an application of the cross-covariance, let's compute it between the two electrodes during the first trial of the ECoG data:

```
load('Ch5-ECoG-1.mat')                %Load the ECoG data.
dt = t(2)-t(1);                       %Define sampling interval.
x = E1(1,:) - mean(E1(1,:));          %Define one time series,
y = E2(1,:) - mean(E2(1,:));          %... and another.
[xc,lags]=xcorr(x,y,100,'biased');    %Compute trial 1 cross cov.
```

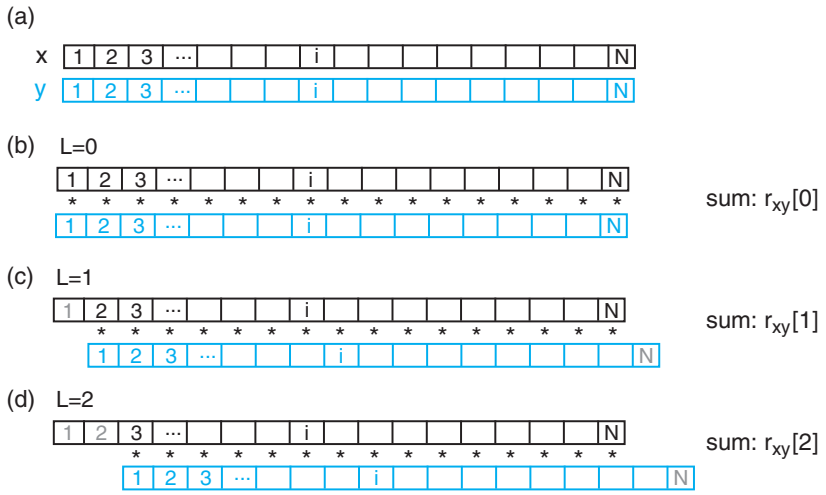

Figure 5.4

Cartoon representation of cross-covariance between two time series x and y. Data x and y are visualized as one-dimensional vectors, x in black and y in blue. The cross-covariance at (*b*) lag 0, (*c*) lag 1, and (*d*) lag 2 requires different alignments between the two vectors. To compute the cross-covariance at each lag, we multiply the overlapping elements of the two vectors, and sum the product. Non-overlapping elements are not included in the computation.

```
plot(lags*dt,xc)                    %Plot cov vs lags in time.
xlabel('Lag [s]')                   %... with axes labeled.
ylabel('Cross-covariance');
```

We subtract the mean from each electrode (the third and fourth lines) before computing the cross-covariance (in the fifth line) using the MATLAB function `xcorr`. In this case, we supply the `xcorr` function with four inputs, beginning with the two time series, `x` and `y`, and including the maximum number of lags to consider (`100`), and the keyword specifying calculation of the biased cross-covariance.

Q: Examine the cross-covariance between the ECoG data from the two electrodes in the first trial (figure 5.5a). What do you observe? At what lags are the largest and smallest values of the cross-covariance? How do these results compare to the trial-averaged autocovariance? How do these results compare to the voltage traces from each electrode in the first trial (figure 5.5b)?

Like the trial-averaged autocovariance for a single electrode (figure 5.3), the cross-covariance between the two ECoG electrodes in the first trial reveals periodic variations (figure 5.5a). To understand the structure of this cross-covariance, let's return to the voltage traces from the two electrodes in this trial (figure 5.5b). The largest peak in the cross-covariance occurs near a lag of 0.04 s. Now, imagine shifting the blue time series

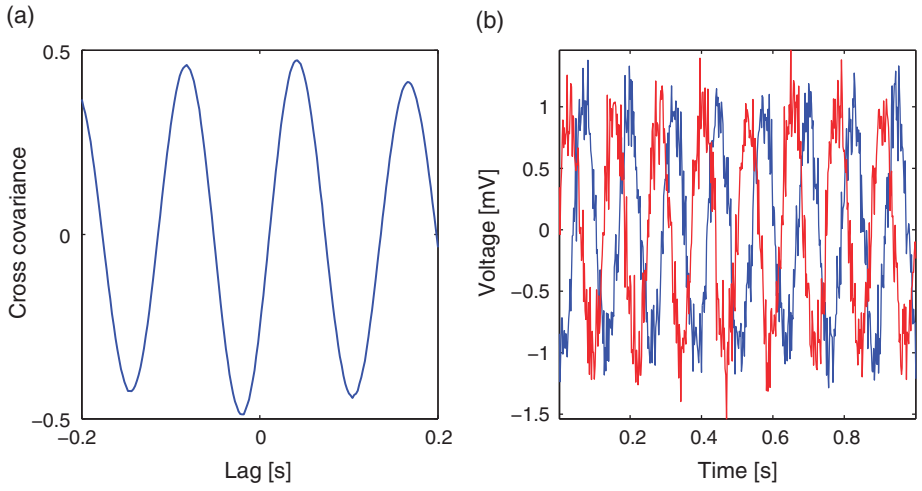

Figure 5.5

Cross-covariance between the two ECoG electrodes for the first trial. (*a*) Rhythmic structure is apparent. (*b*) Voltage data (with mean subtracted) for electrode 1 (*blue*) and electrode 2 (*red*).

(corresponding to electrode 1) in figure 5.5b by 0.04 s to the left. Doing so, we find that the red and blue traces approximately match; at this lag, when one time series is positive, so is the other, and when one time series is negative, so is the other. Because of this strong match, the cross-covariance is large; the sum in (5.2) at this lag involves many positive terms, so $r_{xy}[L]$ is a positive number. The largest *trough* in the cross-covariance occurs near a lag of -0.02 s. To understand this feature, imagine shifting the blue time series in figure 5.5b by 0.02 s to the right. After this shift, the red and blue time series match, but in a different way; when one voltage trace is positive, the other is negative, and vice versa. Therefore, the sum in (5.2) at this lag involves many negative terms, so $r_{xy}[L]$ is a negative number.

Q: Continue this exercise of comparing the cross-covariance with the voltage traces in figure 5.5. At what lags is the cross-covariance near zero? Can you explain these points in terms of shifted versions of the ECoG traces? Can you explain the repeated appearance of peaks (and troughs) at positive and negative lags in terms of shifted versions of the ECoG traces?

Let's also define the *trial-averaged cross-covariance*. The formula is similar to the trial-averaged autocovariance in (5.1):

$$r_{xy}[L] = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{N}\sum_{n=1}^{N-L}(x_{n+L,k}-\bar{x}_k)(y_{n,k}-\bar{y}_k)\,. \tag{5.3}$$

Notice that compared to the trial-averaged autocovariance in (5.1), we have replaced the x's in the last term with y's to compute the trial-averaged cross-covariance in (5.3). To implement the trial-averaged cross-covariance in MATLAB,

```
load('Ch5-ECoG-1.mat')      %Load the ECoG data.
K = size(E1,1);             %Define the number of trials.
dt = t(2)-t(1);             %Define the sampling interval.
maxlags = 100;              %Define variable with max lags.

XC = zeros(K,2*maxlags+1);  %Create variable to store cross cov.
for k=1:K                   %For each trial ...
    x=E1(k,:)-mean(E1(k,:));%...get data from one electrode,
    y=E2(k,:)-mean(E2(k,:));%...and the other electrode,
    [xc0]=xcorr(x,y,maxlags,'biased');%...compute cross cov,
    XC(k,:) = xc0;          %...and store result.
end
XC = mean(XC,1);            %Average cross cov over trials.
```

```
%Plot trial-averaged cross cov vs lags in units of time,
plot((-maxlags:maxlags)*dt,XC)
xlabel('Lag [s]')              %... with axes labeled.
ylabel('Trial-Averaged Cross-Covariance');
```

The implementation of the trial-averaged cross-covariance is similar to the implementation of the single-trial cross-covariance. The main difference is the inclusion here of the `for` statement, which we use to compute and store the cross-covariance of each trial. We then average these results across trials using the `mean` command. The trial-averaged cross-covariance for the ECoG data is shown in figure 5.6a.

Q: Compare the trial-averaged cross-covariance in figure 5.6a to the example single-trial cross-covariances in figure 5.6b. What differences and similarities do you notice between the two cross-covariances?

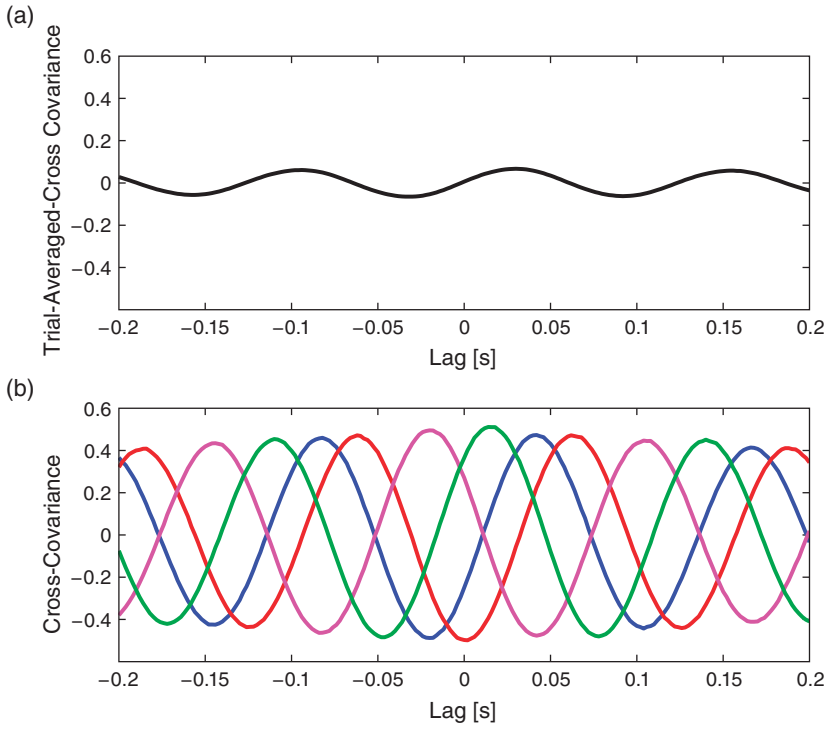

Figure 5.6

(*a*) Trial-averaged cross-covariance, and (*b*) four example single-trial cross-covariances between the two electrodes.

A: Perhaps the most striking difference between the two cross-covariances is their magnitude; the single-trial cross-covariances are much larger—approximately an order of magnitude—than the trial-averaged cross-covariance. To understand why this difference occurs, consider the impact of averaging the four example single-trial cross-covariances in figure 5.6b. At each lag, we find both positive and negative cross-covariance values. We therefore expect that, upon averaging these values across trials, we will obtain a value near zero at each lag. In fact, that's just what we find in the trial-averaged cross-covariance. Because the single-trial cross-covariance functions lack alignment across trials, the averaging procedure acts to cancel out the individual (large) fluctuations of each single-trial cross-covariance.

We may therefore conclude the following. At the single-trial level we find strong cross-covariance that is periodic with period near 0.125 s (examples in figure 5.6b). However, we find much weaker trial-averaged cross-covariance; the cross-covariance structure that exists at the single-trial level does not persist when averaged across trials.

Why are the prominent cross-covariance features in the single-trial analysis lost in the trial-averaged cross-covariance? We discuss this issue in more detail in the chapter summary.

5.2.3 Trial-Averaged Spectrum

One goal of this chapter is to characterize the relations (if any) between the data recorded at the two ECoG electrodes. To do so, let's review a vital tool in this characterization, the Fourier transform. We defined in chapter 3 the Fourier transform of a signal x; we repeat that definition here:

$$X_j = \sum_{n=1}^{N} x_n \exp(-2\pi i f_j t_n)\,. \tag{5.4}$$

Recall that x_n are the data evaluated at time index n. For the ECoG data of interest here, we have 1 s of data sampled at 500 Hz; therefore n ranges from 1 to $N = 500$, and $t_n = \Delta n$ denotes the discrete time steps, where Δ is the sampling interval. The discrete frequencies are $f_j = j/T$, where $j = \{-N/2+1, -N/2+2, \ldots, N/2-1, N/2\}$. Replacing the expressions for f_j and t_n with their definitions and simplifying, we can rewrite (5.4) as

$$X_j = \sum_{n=1}^{N} x_n \exp(\frac{-2\pi i}{N} j n)\,. \tag{5.5}$$

In general, X_j can be a complex quantity (i.e., the Fourier transform of x_n can have both real and imaginary parts). We can therefore think of X_j as residing in the two-dimensional

complex plane (figure 5.7). As you may remember from a geometry or calculus class, we can represent a point in the plane using another coordinate system: polar coordinates. In polar coordinates, we imagine connecting each point to the origin. The resulting line has a length, called the radius or *amplitude*, and forms an angle with the real axis, called the *phase*. Like the real and complex parts, the amplitude and phase uniquely specify each point in the complex plane.[2] These two coordinate systems are shown for an example point in the complex plane in figure 5.7.

Using polar coordinates, we can then express the complex quantity X_j as

$$X_j = A_j \exp(i\phi_j)\,, \tag{5.6}$$

where A_j is the amplitude and ϕ_j is the phase at frequency index j. Notice that both the amplitude and phase are functions of frequency. Recall that to compute the spectrum, we multiply the Fourier transform of the data by its complex conjugate and scale the result (see chapter 3). The spectrum of x_n then becomes

$$S_{xx,j} = \frac{2\Delta^2}{T} X_j X_j^* \tag{5.7}$$

$$= \frac{2\Delta^2}{T} \big(A_j \exp(i\phi_j)\big)\big(A_j \exp(-i\phi_j)\big)\,, \tag{5.8}$$

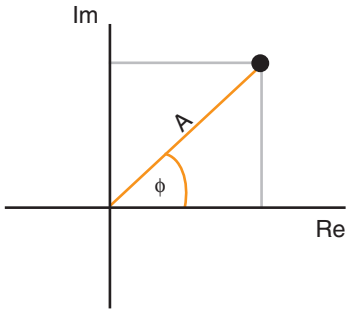

Figure 5.7

Points in the complex plane can be specified in two coordinate systems: Cartesian coordinates (*gray*) or polar coordinates (*orange*). The complex plane contains the real part (horizontal axis) and imaginary part (vertical axis) of every point.

2. This statement is mostly correct. Can you think of the exception? *Hint:* Think small.

where, to compute the complex conjugate in the second term, we replace i with $-i$. The last expression simplifies rather nicely:

$$\begin{aligned} S_{xx,j} &= \frac{2\Delta^2}{T} A_j^2 \exp(i\phi_j - i\phi_j) \\ &= \frac{2\Delta^2}{T} A_j^2 \exp(0) \\ &= \frac{2\Delta^2}{T} A_j^2 \,. \end{aligned} \tag{5.9}$$

This expression provides a new, and perhaps more direct, interpretation of the spectrum as proportional to the squared amplitude of the point X_j in the complex plane. We can extend this simplified expression in one additional way to make explicit the trial structure of the ECoG data analyzed here. Because we possess multiple trials, and we assume that each trial represents an instantiation of the same underlying process, we average the spectra across trials to compute the *trial-averaged spectrum*,

$$< S_{xx,j} >= \frac{2\Delta^2}{T} \frac{1}{K} \sum_{k=1}^{K} A_{j,k}^2 \,, \tag{5.10}$$

where k indicates the trial number, K the total number of trials, and $A_{j,k}$ the amplitude of the signal at frequency index j and trial index k. Notice how we implement the trial averaging: we simply average the squared amplitude at frequency index j across the K trials. We use the angular brackets ($< >$) to denote that the spectrum ($S_{xx,j}$) has been averaged across trials. We can compute the trial-averaged spectrum in MATLAB:

```
load('Ch5-ECoG-1.mat')      %Load the ECoG data.
K = size(E1,1);             %Define the number of trials.
N = size(E1,2);             %Define the number of time indices.
dt = t(2)-t(1);             %Define the sampling interval.
T  = t(end);                %Define the duration of data.

Sxx = zeros(K,N);       %Create variable to store each spectrum.
for k=1:K                   %For each trial,
    x = E1(k,:);            %... get the data,
    xf  = fft(x-mean(x));   %... compute Fourier transform,
    Sxx(k,:) = 2*dt^2/T *(xf.*conj(xf));%... compute spectrum.
end
Sxx = Sxx(:,1:N/2+1);       %Ignore negative frequencies.
Sxx = mean(Sxx,1);          %Average spectra over trials.
```

```
df = 1/max(T);                  %Define frequency resolution,
fNQ = 1/dt/2;                   %... and Nyquist frequency.
faxis = (0:df:fNQ);             %... to construct frequency axis.

plot(faxis, 10*log10(Sxx))      %Plot spectrum in decibels vs
xlim([0 100]);                  %... frequency, in select frequency
ylim([-50 0])                   %... range, in select decibel
xlabel('Frequency [Hz]')        %... range, with axes labeled.
ylabel('Power [ mV^2/Hz]')
```

Q: Are the terms *frequency resolution, Nyquist frequency*, and *decibel* familiar to you? Can you define each in words and equations?

A: If not, we recommend reviewing the case study in chapter 3.

The resulting trial-averaged spectrum is shown in figure 5.8. Compared to the example spectrum from a single trial, the variability is greatly reduced. By reducing the variability in this way, interesting structure in the data may become more apparent.

Q: Upon examining the trial-averaged spectrum from one electrode, what additional conclusions can you now make about the data beyond visual inspection of the voltage traces? Repeat this computation of the trial-averaged spectrum for the second electrode. What do you find? *Hint:* The 8 Hz peak is obvious and consistent with our visual inspection of the data. Do you notice any other (smaller) peaks?

5.2.4 Introduction to the Coherence

Coherence is a measure of association between two time series. Briefly, two signals are coherent at some frequency if there exists a constant phase relation between them at this frequency. To compute the coherence, we use the simplified expression for the spectrum (5.9) and an additional term, the *cross-spectrum*. Consider two signals $x_{n,k}$ and $y_{n,k}$, with time index n and trial index k. These signals have corresponding Fourier transforms $X_{j,k}$ and $Y_{j,k}$. Then the trial-averaged cross-spectrum between these two signals is

$$< S_{xy,j} >= \frac{2\Delta^2}{T}\frac{1}{K}\sum_{k=1}^{K} X_{j,k}Y_{j,k}^{*}, \tag{5.11}$$

where compared to (5.7) we replace X_j^* with Y_j^* and include the average over the trial index k. Let's modify and clean up this expression by using polar coordinates. We first define

$$Y_{j,k} = B_{j,k}\exp(i\theta_{j,k}), \tag{5.12}$$

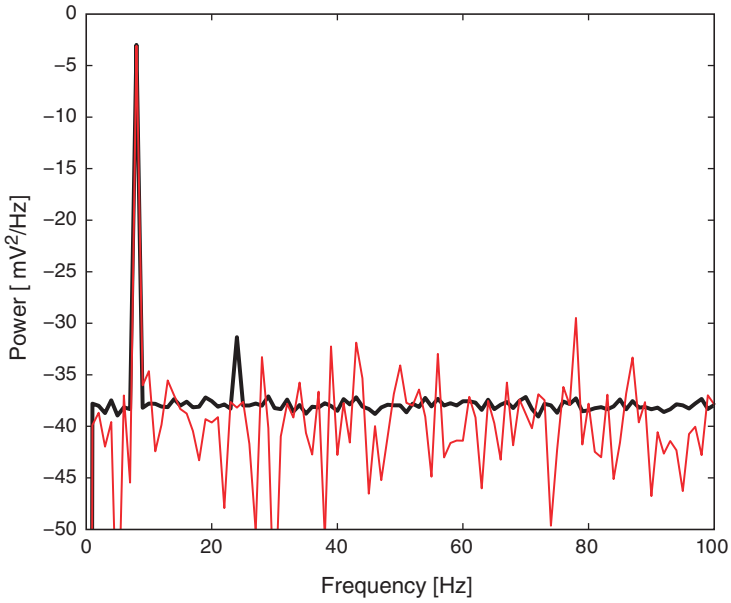

Figure 5.8

Trial-averaged spectrum reduces the variation of the power. Compare the trial-averaged spectrum (*black*) to an example spectrum from an individual trial (*red*).

where $B_{j,k}$ is the amplitude and $\theta_{j,k}$ is the phase at frequency index j and trial index k for the signal $y_{n,k}$. A similar expression exists for $X_{j,k}$, with amplitude $A_{j,k}$ and phase $\phi_{j,k}$. Then replacing $X_{j,k}$ and $Y^*_{j,k}$ in (5.11) with their polar coordinate expressions, we find

$$< S_{xy,j} >= \frac{2\Delta^2}{T}\frac{1}{K}\sum_{k=1}^{K} A_{j,k}B_{j,k}\exp\left(i\Phi_{j,k}\right), \tag{5.13}$$

where we have defined the *phase difference* between the two signals as $\Phi_{j,k} = \phi_{j,k} - \theta_{j,k}$. Equation (5.13) is the trial-averaged cross-spectrum of the two signals $x_{n,k}$ and $y_{n,k}$. We note that the trial-averaged cross-spectrum ($< S_{xy,j} >$) can be complex (i.e., may have nonzero real and imaginary parts).

At last we can define the coherence,

$$\kappa_{xy,j} = \frac{|< S_{xy,j} >|}{\sqrt{< S_{xx,j} >}\sqrt{< S_{yy,j} >}}, \tag{5.14}$$

where $|< S_{xy,j} >|$ indicates the magnitude of the trial-averaged cross-spectrum. In words, the coherence is the magnitude of the trial-averaged cross-spectrum between the two signals at frequency index j divided by the magnitude of the trial-averaged spectrum of each signal at frequency index j.

To further our understanding of the mathematical expression of the coherence in (5.14), let's replace the trial-averaged spectra in the numerator and denominator with their corresponding expressions in polar coordinates:

$$\kappa_{xy,j} = \frac{\left| \sum_{k=1}^{K} A_{j,k} B_{j,k} \exp\left(i\Phi_{j,k}\right) \right|}{\sqrt{\sum_{k=1}^{K} A_{j,k}^2} \sqrt{\sum_{m=1}^{K} B_{j,m}^2}}. \tag{5.15}$$

The expression in (5.15) is complicated. So, to gain some intuition for the behavior of $\kappa_{xy,j}$, let's make the simplifying assumption that at each frequency the amplitude is identical for both signals and all trials, that is, $A_{j,k} = B_{j,k} = C_j$. In using only the expression C_j for the amplitude, we've eliminated the trial index k and only preserved the frequency index j. With this simplifying assumption, the expression for the coherence (5.15) becomes,

$$\begin{aligned}
\kappa_{xy,j} &= \frac{\left| \sum_{k=1}^{K} C_j C_j \exp\left(i\Phi_{j,k}\right) \right|}{\sqrt{\sum_{k=1}^{K} C_j^2} \sqrt{\sum_{m=1}^{K} C_j^2}} \\
&= \frac{C_j^2}{C_j^2} \frac{\left| \sum_{k=1}^{K} \exp\left(i\Phi_{j,k}\right) \right|}{\sqrt{\sum_{k=1}^{K} 1} \sqrt{\sum_{m=1}^{K} 1}} \\
&= \frac{1}{K} \left| \sum_{k=1}^{K} \exp\left(i\Phi_{j,k}\right) \right|.
\end{aligned} \tag{5.16}$$

Under the simplifying assumption that the amplitude is identical at each frequency for both signals and all trials, the coherence simplifies to (5.16). In this special case, the expression for the coherence only involves the phase difference between the two signals averaged across trials; the amplitudes in the numerator and denominator have canceled out.

Now, let's interpret the simplified expression in (5.16). To do so, we consider two scenarios.

Scenario 1: Phases Align across Trials. We assume that at a specific frequency index j, the two signals possess a *constant* phase difference across trials. Under this assumption, the

phase difference in the first trial ($\Phi_{j,1}$) equals the phase difference in the second trial ($\Phi_{j,2}$), which equals the phase difference in the third trial ($\Phi_{j,3}$), and so on for all trials. To denote this equivalence in the phase difference across trials, let's replace the symbol for the phase difference $\Phi_{j,k}$ with $\Phi_{j,0}$; here, we have replaced the subscript k with the subscript 0 to remind ourselves that the phase difference does not depend upon the choice of trial. Now consider the expression

$$\exp\left(i\Phi_{j,0}\right).$$

This term defines a point in the complex plane with amplitude 1, which we can visualize as a vector leaving the origin at angle $\Phi_{j,0}$ to the real axis (figure 5.9a). The summation of these terms across trials then becomes

$$\sum_{k=1}^{K} \exp\left(i\Phi_{j,0}\right).$$

This expression defines a sum of vectors in the complex plane, each of radius 1. Because the phase difference is the same for each trial, these vectors point in the same direction for each trial (figure 5.9a). Then by summing up these vectors end to end across trials, we produce a long vector in the complex plane that terminates far from the origin (figure 5.9a).

Q: How long is the summed vector in this case?

A: We add K vectors (one for each trial) each of length 1, and each pointing in the same direction ($\Phi_{j,0}$). So the total length of the vector (i.e., the total distance from the origin to the termination point of the summed vector) is K.

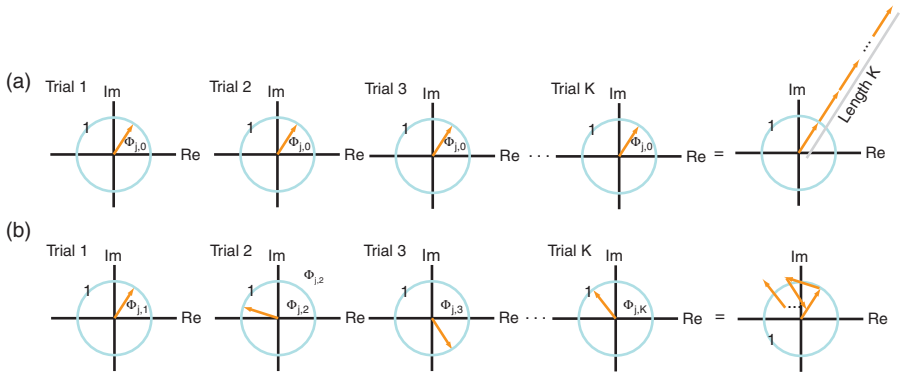

Figure 5.9

Cartoon illustration of the complex plane for two coherence scenarios. (*a*) For each trial, phase difference is the same, and summed vector (last column) terminates far from the origin. (*b*) For each trial, phase difference is random, and summed vector terminates near origin. Blue circles indicate radius 1.

The coherence (5.16) is this vector length, divided by K, so we conclude in this case that

$\kappa_{xy,j} = 1,$

which indicates strong coherence between the two signals. The strong coherence in this case results from the constant phase relation between the two signals across trials at frequency index j.

Q: Does the conclusion $\kappa_{xy,j} = 1$ depend upon the value of the phase difference $\Phi_{j,0}$? For example, does this result require that the phase difference between the two signals in each trial ($\Phi_{j,0}$) equal 0, or $\pi/4$, or π?

Scenario 2: Phases Are Random across Trials. As a second scenario, consider another specific frequency j in which the two signals have a *random* phase difference in each trial. In this case, the phase difference can assume any value between 0 and 2π for each trial. To visualize this, let's imagine the phase differences in the complex plane (figure 5.9b); in this scenario, the vectors point in different (random) directions from trial to trial.

Q: Consider the sum of these vectors end to end in the complex plane. What is the approximate length of this summed vector across trials?

A: We expect the length of this vector to be small. Because the angles lack organization from trial to trial, the vectors are equally likely to point in any direction. Therefore, when we sum these vectors across trials, the length fails to accumulate in any particular direction (figure 5.9b).

Under the simplifying assumption that the amplitude is identical at this frequency for both signals and all trials, the coherence (5.16) is this summed vector length, divided by K. Our visual inspection of figure 5.9b suggests that this summed vector length will be small. Therefore, for this scenario we conclude that

$\kappa_{xy,j} \approx 0,$

which indicates weak coherence between the two signals. The weak coherence in this case results from the random phase relation over trials between the two signals.

Summary of Coherence. These two examples illustrate in simplified scenarios the behavior of the coherence. To summarize, the coherence (5.14) is a measure of the relation between x and y at the same frequency. The coherence ranges between 0 and 1:

$0 \leq \kappa_{xy,j} \leq 1,$

in which 0 indicates no coherence between signals x and y at frequency index j, and 1 indicates strong coherence between signals x and y at frequency index j.

> The coherence is a measure of the phase consistency between two signals at frequency index j across trials.

We note that because computing the coherence requires the Fourier transform, the notions of frequency resolution and Nyquist frequency are identical to those described for the spectrum. In other words, the frequency resolution of the coherence is $1/T$, and the Nyquist frequency is half of the sampling frequency; see chapter 3 for details.

Q: What are the units of the coherence? *Hint:* Consider (5.14) and the units of the terms in the numerator and denominator. You should find that the coherence is unitless.

Cross-Covariance and Cross-Spectrum. Although we defined the cross-spectrum in (5.11) and used it to define the coherence in (5.14), the cross-spectrum may appear somewhat unmotivated. Fortunately, there is additional insight to be gained. We showed in appendix A of chapter 3 that the spectrum is the Fourier transform of the autocovariance. Conceptually, the spectrum and autocovariance provide a frequency domain and time domain measure of a signal's rhythms, respectively. In the same way, the cross-spectrum and cross-covariance are partners.

> The cross-spectrum is the Fourier transform of the cross-covariance.

The cross-spectrum and cross-covariance form a Fourier transform pair. The cross-spectrum is a frequency domain measure of coupling, while the cross-covariance is a time domain measure of coupling. To move back and forth between these two measures, we use the Fourier transform. In practice, we rarely examine the cross-spectrum directly; it's a complex quantity and so requires two dimensions (i.e., the complex plane) to visualize. However, the cross-spectrum is fundamental to the coherence, so in that sense it's an important actor in the analysis.

Computing the Coherence. With that introduction, we are now equipped to compute the coherence. We expect the coherence to reveal the frequencies at which the two ECoG signals exhibit a constant phase relation across trials.

Q: Before we compute the coherence, hypothesize whether you expect to observe coherence between the two ECoG signals. If so, at what frequencies? Your hypothesis should be based on the previous visual analysis and spectral analysis of these data (see, for example, figures 5.1 and 5.8).

Q: To plot the coherence versus frequency, we must identify the frequency resolution and Nyquist frequency appropriate for the analysis of the ECoG data. What are they?

There are a variety of alternatives to compute the coherence. To start, let's compute the coherence by hand. The reason for doing so is that we can implement the preceding mathematical expressions and in that way gain more understanding of their features. Here's the MATLAB code:

```
load('Ch5-ECoG-1.mat')  %Load the ECoG data.
K = size(E1,1);         %Define the number of trials.
N = size(E1,2);         %Define the number of indices per trial.
dt = t(2)-t(1);         %Define the sampling interval.
T  = t(end);            %Define the duration of data.

Sxx = zeros(K,N);       %Create variables to save the spectra,
Syy = zeros(K,N);
Sxy = zeros(K,N);
for k=1:K               %... and compute spectra for each trial.
    x=E1(k,:)-mean(E1(k,:));
    y=E2(k,:)-mean(E2(k,:));
    Sxx(k,:) = 2*dt^2/T * (fft(x) .* conj(fft(x)));
    Syy(k,:) = 2*dt^2/T * (fft(y) .* conj(fft(y)));
    Sxy(k,:) = 2*dt^2/T * (fft(x) .* conj(fft(y)));
end

Sxx = Sxx(:,1:N/2+1);   %Ignore negative frequencies.
Syy = Syy(:,1:N/2+1);
Sxy = Sxy(:,1:N/2+1);

Sxx = mean(Sxx,1);      %Average the spectra across trials.
Syy = mean(Syy,1);
Sxy = mean(Sxy,1);      %... and compute the coherence.
cohr = abs(Sxy) ./ (sqrt(Sxx) .* sqrt(Syy));
```

```
df = 1/max(T);          %Determine the frequency resolution.
fNQ = 1/dt/2;           %Determine the Nyquist frequency,
faxis = (0:df:fNQ);     %... and construct frequency axis.

plot(faxis, cohr);      %Plot coherence vs frequency,
xlim([0 50])            %... in chosen frequency range,
ylim([0 1])
xlabel('Frequency [Hz]')%... with axes labeled.
ylabel('Coherence')
```

Q: That's quite a bit of code. Look through it line by line, and confirm that each step makes sense. Can you identify the calculation of the cross-spectrum? of the trial averaging?

Q: Consider the coherence between the two ECoG electrodes in figure 5.10. At what frequencies do strong coherences appear? How do these frequencies compare to the trial-averaged spectra, shown for one electrode in figure 5.8?

A: The coherence measures the phase consistency at a chosen frequency between two signals across trials. For the ECoG data, both electrodes possess trial-averaged spectra with similar features: a large peak near 8 Hz and a smaller peak near 24 Hz (see the trial-averaged spectrum for one electrode in figure 5.8). However, the coherence between the two ECoG signals reveals a peak only at 24 Hz (figure 5.10). We conclude that the two ECoG signals both exhibit a dominant oscillation at 8 Hz, yet this rhythm is not coherent across trials; only the smaller-amplitude rhythm at 24 Hz is coherent between the two electrodes.

5.2.5 Visualizing the Phase Difference across Trials

The coherence results suggest for the two ECoG recordings a constant phase relation across trials at 24 Hz and a random phase relation across trials at 8 Hz. To further explore these relations, let's visualize the distribution of phase differences at the two frequencies, as implemented in the following MATLAB code:

```
load('Ch5-ECoG-1.mat')  %Load the ECoG data.
K = size(E1,1);         %Define the number of trials.
N = size(E1,2);         %Define the number of indices per trial.
dt = t(2)-t(1);         %Define the sampling interval.
```

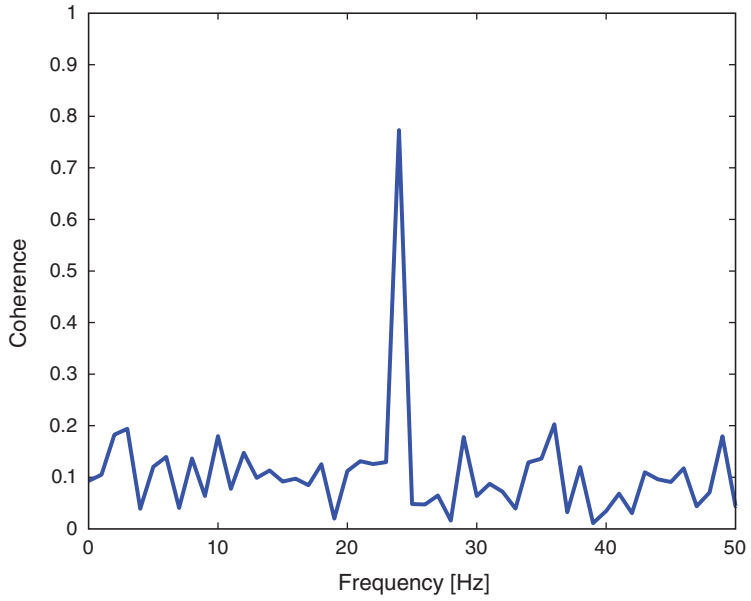

Figure 5.10
Coherence between the two ECoG signals.

```
T  = t(end);              %Define the duration of data.
df = 1/max(T);            %Determine the frequency resolution.
fNQ = 1/dt/2;             %Determine the Nyquist frequency,
faxis = (0:df:fNQ);       %... and construct frequency axis.

j8 = find(faxis == 8);  %Determine index j for frequency 8 Hz.
j24= find(faxis == 24); %Determine index j for frequency 24 Hz.

phi8=zeros(K,1);          %Variables to hold phase differences.
phi24=zeros(K,1);

for k=1:K                 %For each trial, compute cross spectrum,
    Sxy = fft(E1(k,:)).*conj(fft(E2(k,:)));
    phi8(k)  = angle(Sxy(j8));  %... and the phases.
    phi24(k) = angle(Sxy(j24));
end

subplot(1,2,1)            %Display the phase differences.
rose(phi8);   title('\Phi at 8 Hz')
```

```
subplot(1,2,2)
rose(phi24);  title('\Phi at 24 Hz')
```

Again, we're encountering quite a bit of MATLAB code. Fortunately, large chunks of this code are familiar. We begin by defining useful quantities, like the number of trials (`K`), the number of indices per trial (`N`), and the frequency axis (`faxis`). Then, within the frequency axis variable (`faxis`), we use the `find` function to identify the indices corresponding to a frequency of 8 Hz and a frequency of 24 Hz. For each trial, we then compute the cross-spectrum (`Sxy`). The cross-spectrum is a complex quantity at each frequency, and we identify the angle in the complex plane corresponding to the frequencies 8 Hz and 24 Hz using the MATLAB function `angle`. We store these results in two vectors, `phi8` and `phi24`.

The function `rose` displays a histogram of the phase differences (figure 5.11). By default, the phase axis is divided into 20 bins of equal size from 0 to 2π radians, or equivalently, 0 to 360 degrees. At 8 Hz, we observe that phase differences appear in all angular intervals; notice that the number of phase differences located in each angular interval remains small, typically less than 10. At 24 Hz, the angular differences concentrate near 0 degrees; all of the angles lie between -60 and 60 degrees. This visualization is consistent with the strong coherence at 24 Hz, indicative of a consistent phase difference across trials between the two electrodes.

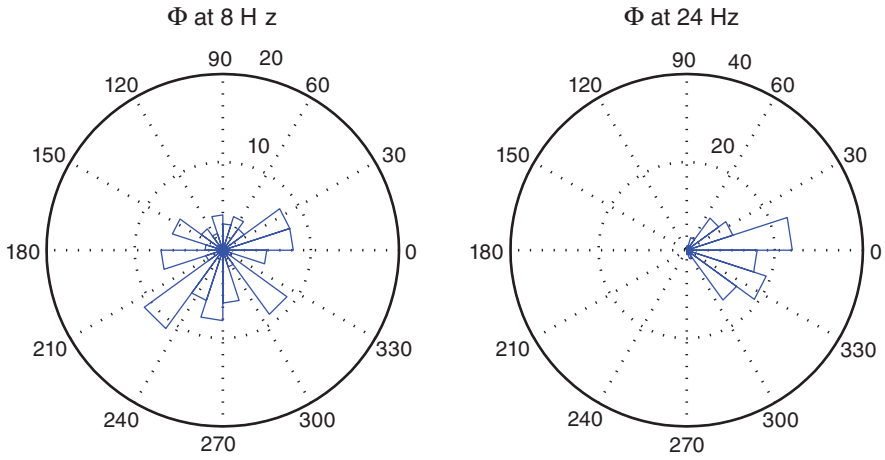

Figure 5.11

Distribution of phase differences between the two ECoG signals depends on the frequency. Angular histograms of phase differences at 8 Hz (*left*) and 24 Hz (*right*). Number of counts in each phase bin is indicated by labeled circles.

Q: Compute and display the distribution of phase differences at other frequencies. What do you find? Are these results consistent with the coherence plotted in figure 5.10?

5.2.6 Single-Trial Coherence

We have emphasized that coherence is a measure of phase consistency between two signals at some frequency *across trials*. This type of analysis is appropriate in many instances in which data are collected in a trial structure. However, we might also be interested in computing the coherence between two signals recorded in a single observation or trial.

Q: Is it possible? Can we compute the coherence between two signals for a single trial?

To address this question, consider the equation for the coherence written in polar coordinates (5.15). Remember that in writing this equation, we have made no assumptions about the data; instead, all we have done is express the complex quantities in polar coordinates. Now consider (5.15) for the case in which we possess only one trial, so that $K = 1$. Then

$$\kappa_{xy,j} = \frac{\left| A_{j,1} B_{j,1} \exp\left(i\Phi_{j,k}\right) \right|}{\sqrt{A_{j,1}^2}\sqrt{B_{j,1}^2}} = \left| \exp\left(i\Phi_{j,k}\right) \right| = 1 \,. \tag{5.17}$$

So, we find here perfect coherence ($\kappa_{xy,j} = 1$) for any choice of signals x and y and for any frequency (index j). For example, we could choose x to be the price of a publicly traded stock (e.g., GE) and y to be an ECoG recording, both sampled at 500 Hz for 1 s. Even in this case, we will find perfect coherence between the two signals.

Q: Can we use an ECoG signal to predict the stock price of GE? If so, then we're rich! How can any two arbitrary signals be perfectly coherent at all frequencies?

The answer is that the coherence measure requires a trial structure. Recall that the coherence measures the phase consistency between two signals *across trials*. If only one trial is observed, then the two signals are trivially coherent; the two signals have some phase difference between 0 and 2π and because we have no other trials with which to compare this difference, the two signals are "coherent."

But what if we only collect one trial of data? We can still attempt to compute the coherence in (at least) two ways. First, we could divide the single trial of data into smaller intervals and then treat each interval as a trial. This approach can be effective if we believe

the phase relation persists in time, and if we possess a long enough recording. Note that by dividing the data into smaller intervals, we impact the frequency resolution.

Q: Imagine we collect 10 s of ECoG data (sampling frequency 500 Hz) from two electrodes and would like to compute the coherence. To do so, we divide the data into ten nonoverlapping 1 s intervals, and treat each interval as a trial to compute the coherence. What is the frequency resolution of the coherence? If instead we divide the data into 100 nonoverlapping frequency intervals, what is the frequency resolution? In both cases, what is the Nyquist frequency?

A second approach to compute the coherence from a single trial of data is to use the multitaper method. In this case, each taper acts like a trial. Therefore, to acquire more trials for an accurate estimate of the coherence, we include more tapers. But remember that to increase the number of tapers we worsen the frequency resolution (see chapter 4). Computing the coherence using a multitaper method is made relatively easy by software packages like Chronux [2]. Because the ECoG data of interest here consist of multiple trials, we do not focus on measures of single-trial coherence. An example of using the multitaper method to compute the coherence is described in the problems section at the end of this chapter.

Relation between Statistical Modeling and Coherence. Before concluding the discussion of coherence, let's briefly consider a complementary statistical modeling approach. In developing this statistical model, our goal is to capture the (linear) relation between two signals x and y observed simultaneously for multiple trials. We begin by proposing a statistical model that predicts one signal (y) as a linear function of the other (x):

$$\begin{aligned} y_n &= \sum_{m=-\infty}^{\infty} \beta_m x_{n-m} + \epsilon_n \\ &= (\beta \star x)\big[n\big] + \epsilon_n \,, \end{aligned}$$

where we express the predicted signal (y_n) as a function of x_n, coefficients β_m, and a Gaussian noise term ϵ_n, and where n is a discrete-time index. In the first equation, the summation limits indicate that the predicted signal at time index n may depend on x at any past or future time. The second equality expresses the summed product of β and x as their convolution. Taking the Fourier transform of both sides of this equation, and remembering that convolution in the time domain is equivalent to multiplication in the frequency domain, we find

$$Y_j = \gamma_j X_j + \Upsilon_j \,,$$

where Y_j is the Fourier transform of y_n, γ_j is the Fourier transform of β_n, X_j is the Fourier transform of x_n, Υ_j is the Fourier transform of ϵ_n, and j indicates a discrete frequency index. Multiplying both sides of this equation by the complex conjugate of the Fourier transform of x,

$$Y_j X_j^* = \gamma_j X_j X_j^* + \Upsilon_j X_j^* ,$$

and averaging this result across the trials of data, we find

$$< Y_j X_j^* > = \gamma_j < X_j X_j^* > + < \Upsilon_j X_j^* > ,$$

where we use the notation $< >$ to indicate the trial average. Assuming that the noise term and signal x are unrelated, their trial average is zero (i.e., $< \Upsilon_j X_j^* >= 0$). Solving for γ_j, we find

$$\begin{aligned} \gamma_j &= \frac{< Y_j X_j^* >}{< X_j X_j^* >} \\ &= \frac{< S_{xy,j} >}{< S_{xx,j} >} . \end{aligned} \tag{5.18}$$

Then, comparing (5.18) to the equation for coherence (5.14), we find

$$\kappa_{xy,j} =\mid \gamma_j \mid \frac{\sqrt{< S_{xx,j} >}}{\sqrt{< S_{yy,j} >}} . \tag{5.19}$$

We conclude that the coherence ($\kappa_{xy,j}$) is a scaled version of the frequency domain representation of the statistical model coefficients (γ_j) for predicting y from x. We note that γ_j is a complex quantity that allows us to model both the magnitude and phase of the relation between x and y. The phase difference computed from the model and the coherence is the same as well.

Summary

In this chapter, we analyzed ECoG data recorded from two electrodes during an auditory task. The task involved the repeated presentation of auditory stimuli, resulting in 100 trials of 1 s duration recorded simultaneously from the two electrodes. We began the analysis with visual inspection of individual trials and of all trials at once. Then, to assess the relations between the two recordings, we computed the cross-covariance. We discussed how the cross-covariance is an extension of the autocovariance, and found that the single-trial cross-covariance between the ECoG signals exhibited periodic structure, consistent with rhythmic coupling of period 0.125 s. However, the trial-averaged cross-covariance provided less evidence for consistent rhythmic coupling across trials. We then computed the trial-averaged spectrum and found a large peak near 8 Hz and a much smaller peak near 24 Hz.

To further assess the relation between the two electrodes, we computed the coherence. The coherence is strong (approaches 1) at a chosen frequency f_0 when there exists a constant phase relation at frequency f_0 between two electrodes over trials. We found a strong coherence between the two ECoG electrodes only at 24 Hz. We concluded that although both ECoG signals possessed dominant rhythms at 8 Hz, these rhythms were not coherent between the two electrodes. The strong coherence appeared only at the small-amplitude 24 Hz rhythm. Finally, we implemented a technique to visualize the distribution of phase differences between the two electrodes across trials, and provided some suggestions for how to compute the coherence for a single trial of data.

Caution! Large amplitude does not imply large coherence.

In this example, only the coherence revealed the low-amplitude coupling at 24 Hz between the two ECoG electrodes. This coupling was not obvious in the single-trial or trial-averaged cross-covariance. In fact, the single-trial cross-covariance was deceiving; we found strong single-trial cross-covariance with period 0.125 s, or 8 Hz (figure 5.6b), yet no coherence at 8 Hz.

To understand this discrepancy, consider two unrelated signals, each dominated by the same rhythm. By *unrelated* we mean that the signals do not communicate in any way. Yet both are rhythmic and happen to oscillate at the same frequency. If we compute the cross-covariance between these two unrelated signals, we will find periodic lags at which two signals nearly overlap and the cross-covariance is large. The period of these cross-covariance peaks corresponds to the period of the common rhythm shared by the two signals. Here the periodic, large cross-covariance values occur because the two signals happen to both exhibit a similar rhythm, not because one signal influences the other.

This example illustrates a point of caution in the interpretation of cross-covariance results. Unrelated signals that happen to share a similar dominant rhythm will exhibit large periodic structure in the cross-covariance. One approach to defend against such cross-covariance results is to compute the trial-averaged cross-covariance. If two signals are unrelated—to one another and to the trial structure—then we do not expect similar cross-covariance functions across trials. Therefore, although each single-trial cross-covariance may have large values at some lags, their average across trials will be small. This is just what we found for the ECoG data examined here (figure 5.6). We note that the unrelated 8 Hz signals, which dominate the ECoG activity at each electrode, mask the much smaller amplitude 24 Hz activity that is coupled between the two electrodes. The coupling at 24 Hz is not apparent in the trial-averaged cross-covariance (figure 5.6). The coherence, which normalizes by the power at each frequency, uncovers this relation.

As is true for the Fourier transform and spectrum, there exists a vast literature on computing and interpreting the coherence. Some references for further reading include [8, 9, 11].

Problems

5.1. Consider two signals x and y, where x is a cosine function and y is a sine function. Both signals are of duration 2 s and of frequency 10 Hz. Simulate both signals (each with a sampling interval of 0.001 s) and compute their cross-covariance. What do you find, and how do you interpret the results? Imagine that the signal x was collected from the scalp EEG of a human subject two years ago, while signal y was collected from a voltage recording made in rat hippocampus yesterday. Would you expect these two signals—collected from very diverse preparations—to be related? How does this knowledge impact your interpretation of the cross-covariance results? Consider your answer in terms of the cautions issued in the chapter summary.

5.2. Generate synthetic data consisting of Gaussian noise. More specifically, generate 100 trials of 1 s data sampled at 500 Hz. Do this twice to generate two synthetic datasets, and the compute the following:

a. The trial-averaged spectrum of each synthetic dataset.

b. The trial-averaged cross-covariance between the two synthetic datasets.

c. The coherence between the two synthetic datasets.

Describe your results for each analysis. What cross-covariance and coherence results do you expect to find between these noisy, unrelated sets of data? Do your results match your expectations?

5.3. Generate synthetic data consisting of a sinusoid oscillating at frequency f plus additive Gaussian noise. More specifically, generate 100 trials of 1 s data sampled at 500 Hz. For each trial, set the initial phase of the sinusoid to a random value between 0 and 2π. Repeat this procedure to create a second dataset, but in this case fix the initial phase of the sinusoid to π. Then compute the coherence between these two synthetic datasets. What do you expect to find (i.e., do these two signals possess a constant phase relation across trials at any frequency)? Do your coherence results match your expectations?

5.4. Generate synthetic data consisting of Gaussian noise. More specifically, generate 1 s of data sampled at 500 Hz. Do this twice, and then compute the coherence between these two synthetic signals. Notice, in this case, the data consist of *single trials*. What do you expect to find (i.e., are these noisy signals coherent at any frequency)? Do your coherence results match your expectations?

5.5. In chapter 4, we discussed the notion of tapering in the context of computing the spectrum. Tapers are also applicable in computing the coherence. To analyze the ECoG data in this chapter, we applied the (default) rectangular taper. To (briefly) investigate the application of an alternative tapering procedure, let's consider the Hanning taper and use it to compute the coherence. The Hanning taper is discussed in detail

in chapter 4. To apply the Hanning taper, update the MATLAB code provided in section "Computing the coherence." You will need to apply the Hanning taper to the data from each trial before computing the Fourier transforms and the spectra. Compare the results of your coherence analysis using the Hanning taper to the coherence analysis using the (default) rectangular taper shown in figure 5.10. How does the coherence change?

5.6. Load the file `Ch5-ECoG-2.mat`, available at

```
http://github.com/Mark-Kramer/Case-Studies-Kramer-Eden
```

into MATLAB. You will find three variables in your workspace. The variables `x` and `y` correspond to two simultaneous recordings of ECoG activity from two electrodes. Both of these variables are organized so that the rows correspond to trials and the columns to time. You should find 100 trials, with $1,000$ time points per trial. The variable `t` corresponds to the time axis for these data, in units of seconds. Use these data to answer the following questions.

a. Visualize the data from each electrode. What rhythms do you observe?

b. Plot the trial-averaged spectrum versus frequency for each electrode. Are the dominant rhythms in the spectrum consistent with your visual inspection of the data?

c. Plot the trial-averaged cross-covariance between the two datasets. What features do you observe?

d. Plot the coherence between the two datasets. At what rhythms, if any, is the coherence large?

e. Summarize the results of your data analysis. What are the important features of these data that you would communicate to a colleague?

5.7. Load the file `Ch5-ECoG-3.mat`, available at

```
http://github.com/Mark-Kramer/Case-Studies-Kramer-Eden
```

into MATLAB. You will find three variables in your workspace. The variables `x` and `y` correspond to two simultaneous recordings of ECoG activity from two electrodes. Both of these variables are organized so that the rows correspond to trials and the columns to time. You should find 100 trials, with $1,000$ time points per trial. The variable `t` corresponds to the time axis for these data, in units of seconds. Use these data to answer the following questions.

a. Visualize the data from each electrode. What rhythms do you observe?

b. Plot the trial-averaged spectrum versus frequency for each electrode. Are the dominant rhythms in the spectrum consistent with your visual inspection of the data?

c. Plot the trial-averaged cross-covariance between the two datasets. What features do you observe?

d. Plot the coherence between the two datasets. At what rhythms, if any, is the coherence large?

e. Summarize the results of your data analysis. What are the important features of these data you would communicate to a colleague?

5.8. (*Advanced*) As an illustration of using the multitaper method to compute the coherence, we consider the following synthetic data: two time series each consisting of 10 s of Gaussian noise, generated with a sampling interval of 0.001 s. We generate these data in MATLAB as follows:

```
T = 10;              %Define total duration of data,
dt = 0.001;          %... sampling interval,
N = T/dt;            %... and no. of pts in data.
x = randn(N,1);      %Generate Gaussian noise data,
y = randn(N,1);      %... and additional Gaussian noise data.
```

Using the approach described in this chapter to compute the coherence between these two signals, we find a value of 1 at all frequencies; see section "Single-trial Coherence." Of course, that's not the answer we want. The two simulated time series are Gaussian noise, and we expect no coherence between them at any frequency. Let's instead use the multitaper method to compute the coherence between these two synthetic signals. First download and install the software package Chronux; see section 1.24 of chapter 1. We use the function `coherencyc` from this software package. When using the multitaper method, we need to choose the time-bandwidth product. Let's assume we demand a resolution bandwidth of 4 Hz. Then the time-bandwidth product is $(10\,\mathrm{s}) \times (2\,\mathrm{Hz}) = 20$, and we choose $2 * 20 - 1 = 39$ tapers. Conceptually, we can think of each taper as acting like a trial; each of these trials selects a different chunk of the synthetic data. The coherence then evaluates the phase consistency of these data across the trials. In MATLAB,

```
%Set the parameters of the MTM.
TW = 20;               %Choose time-bandwidth product of 20.
ntapers = 2*TW-1;      %...which sets the no. of tapers.
params.Fs = 1/dt;      %Define sampling frequency,
params.tapers = [TW,ntapers];%... time-band product,
                             %... no. of tapers.
params.pad = -1;       %Specify no zero padding,
                       %... and compute the coherence.
[C,phi,S12,S1,S2,f]=coherencyc(x, y, params);
```

```
plot(f,C)                %Plot the coherence vs frequency,
ylim([0 1])              %... set the vertical axis,
xlabel('Frequency [Hz]')%... and label the axes.
ylabel('Coherence')
```

We set the parameters used by the multitaper method with the variable `params`. This variable is a *structure array*, with data containers called fields; see MATLAB Help for more details about structure arrays. We use the Chronux function `coherencyc` to compute the coherence using the multitaper method. This function returns a variety of outputs, including the cross-spectrum (`S12`) and the spectra (`S1` and `S2`). We focus here only on the coherence, and plot the results in figure 5.12. We find that using the multitaper method, the coherence is not 1 for all frequencies.[3] Instead, the coherence remain small and fluctuates between 0.0 and approximately 0.4. That's closer to the answer we expect for two unrelated noisy time series. This example illustrates that the multitaper method can serve to evaluate the coherence for a single trial of data and does not simply produce a trivial result. To compute this coherence, we accept

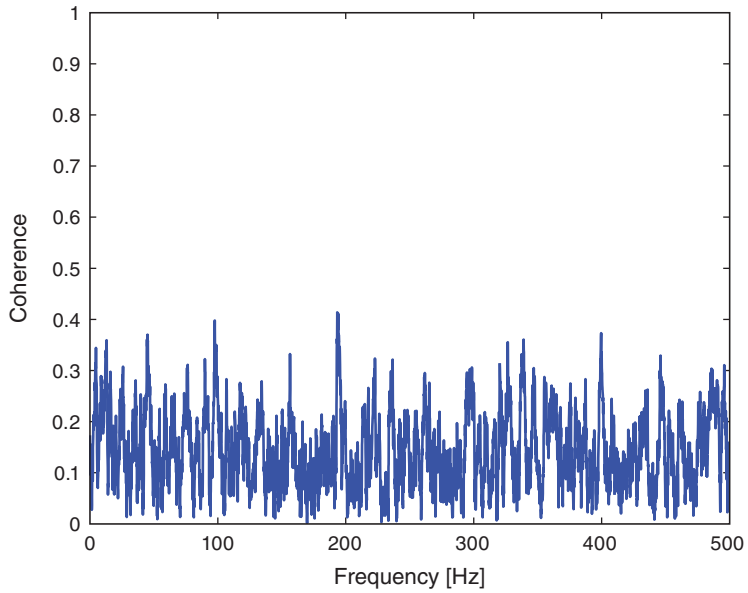

Figure 5.12

Coherence between two Gaussian noise time series.

3. The plot you generate will differ slightly in detail from the plot in figure 5.12 because the synthetic noise data will differ.

worse frequency resolution to acquire the tapers over which to detect constant phase relations between the two signals.

To further examine this assessment of single-trial coherence, consider the following synthetic signals: to the variables `x` and `y` defined in preceding code, add a sine function of amplitude 1 and frequency 10 Hz. Recompute the single-trial coherence using the multitaper method for these new data and examine the results. Does the computed coherence match your expectations for these new data?