4 Analysis of Rhythmic Activity in an Invasive Electrocorticogram

Synopsis

Data Field data: 1 s of ECoG data sampled at 500 Hz.

Goal Characterize the observed rhythms in these data.

Tools Fourier transform, spectrum, tapers, multitaper method.

4.1 Introduction

4.1.1 Background

In chapter 3, we considered noninvasive recordings of brain electrical activity from the scalp surface. Although the scalp EEG provides fine temporal resolution of brain activity, the spatial resolution is poor because of the low conductivity of the skull [5]. An alternative, invasive approach to improve the spatial resolution of the scalp EEG is to record directly from the brain's surface [10]. This technique, known as electrocorticogram (ECoG), eliminates the distorting spatial blurring effect of the skull, at the cost of an invasive surgical procedure of implantation.

4.1.2 Case Study Data

A patient with epilepsy is scheduled to undergo resective surgery to remove the portion of her brain causing recurrent, unprovoked seizures. As part of her clinical workup, electrodes are implanted beneath the skull, directly on the brain's surface. We assume that our skilled neurosurgeon collaborator expertly implants the ECoG electrode, and that the ECoG data are collected with no artifacts. We receive from our clinical collaborator a 1 s segment of ECoG data recorded from a single electrode and sampled at 500 Hz.

4.1.3 Goal

Our collaborator (and we) would like to know what rhythms appear in these invasive brain voltage recordings. Our goal is to analyze the 1 s of ECoG data by characterizing the rhythmic attributes of the activity. To do so, we build upon the spectral analysis techniques developed in chapter 3 and focus on the impact of two important elements in computing the spectrum: windowing and zero padding.

4.1.4 Tools

In this chapter, we continue to develop understanding of the Fourier transform and spectrum. We apply the techniques introduced in chapter 3 to compute the spectrum. We also investigate the impact of windowing and zero padding on the spectrum, and explain how to apply and interpret the multitaper method.

4.2 Data Analysis

4.2.1 Visual Inspection

To access the data for this chapter, visit

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

and download the file `Ch4-ECoG-1.mat`. We begin as always by looking at the data. In almost all cases, visual inspection is the first step in data analysis. We load the ECoG data into MATLAB and plot them by issuing the following commands:

```
load('Ch4-ECoG-1.mat')        %Load the ECoG data.
plot(t,ECoG)                  %Plot voltage vs time.
xlabel('Time [s]')            %Label time axis.
ylabel('Voltage [mV]')        %Label voltage axis.
```

Q: The ECoG data are plotted in figure 4.1. What do you see?

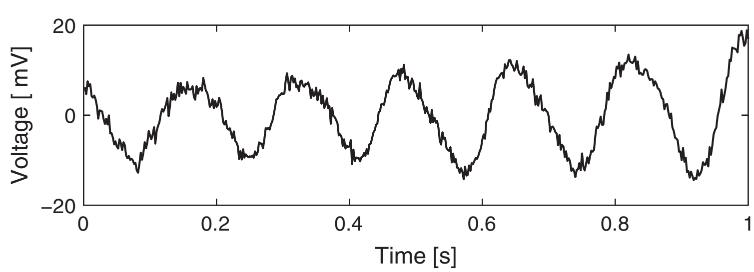

Figure 4.1
ECoG data provided by our collaborator.

You might notice a dominant rhythmic activity. We can approximate the frequency of this rhythm by counting the number of oscillations that occur in the 1 s interval. To do so, we may approximate the total number of large-amplitude cycles that we observe in the data. Through visual inspection of figure 4.1, we find that the first large-amplitude cycle occurs between ≈ 0 s and ≈ 0.175 s, the next between ≈ 0.175 s and ≈ 0.3 s, and so on. Counting this way, we approximate 6 full cycles per second, or a dominant 6 Hz rhythm.

4.2.2 Spectral Analysis: The Rectangular Taper and Zero Padding

Visual inspection, although essential to data analysis, is usually not enough. Visual inspection often guides intuition and reveals major features of the data, but it may also lead us astray; initial looks can be deceiving. To further explore the rhythmic activity of the ECoG data, we compute the spectrum.[1] We do so using the same approach implemented in chapter 3. The MATLAB code is nearly the same:

```
load('Ch4-ECoG-1.mat')          %Load the ECoG data.
x  = ECoG;                      %Relabel the data variable.
dt = t(2)-t(1);                 %Define the sampling interval,
T  = t(end);                    %... and duration of data.

xf = fft(x-mean(x));            %Compute Fourier transform of x,
Sxx = 2*dt^2/T*(xf.*conj(xf));  %... and the spectrum.
Sxx = Sxx(1:length(x)/2+1);     %Ignore negative frequencies.

df = 1/T;                       %Define frequency resolution.
fNQ = 1/dt/2;                   %Define Nyquist frequency.
faxis = (0:df:fNQ);             %Construct frequency axis.

plot(faxis, Sxx)                %Plot spectrum vs frequency,
xlim([0 100])                   %... in select frequency range,
xlabel('Frequency [Hz]')        %... with axes labeled.
ylabel('Power [mV^2/Hz]')
```

Q: For the ECoG data considered here, what is the frequency resolution df, and what is the Nyquist frequency (f_{NQ})? Compare your answers to the variables `df` and `fNQ` defined in this MATLAB code.

---

1. We could instead write the *sample* spectrum because we use the observed data to estimate the theoretical spectrum that we would see if we kept repeating this experiment. However, this distinction is not essential to the discussion here.

Q: Interpret the spectrum of the ECoG data plotted in figure 4.2a. What rhythms appear?

The visualization in figure 4.2a suggests a single dominant frequency near 6 Hz, consistent with the visual inspection of the ECoG trace (see figure 4.1). Other interesting structure may also appear, perhaps at frequencies near 10 Hz; note the tiny peak barely visible in the spectrum shown in figure 4.2a. These initial observations suggest we can more appropriately scale the spectrum to emphasize both the low-frequency bands and weaker signals. Let's utilize a logarithmic scale for both the power spectral density (decibels) and the frequency. In MATLAB,

```
semilogx(faxis, 10*log10(Sxx))  %Plot spectrum vs frequency,
xlim([0 100])                   %... in select frequency range,
xlabel('Frequency [Hz]')        %... with axes labeled.
ylabel('Power [dB]')
```

The first line of code is updated to plot the *x*-axis (frequency) on a logarithmic scale and to scale the spectrum to decibels. The resulting spectrum (figure 4.2b) confirms the initial observation of a strong rhythm at 6 Hz (the dominant peak in figure 4.2b) and also suggests interesting structure near 10 Hz. But what is really going on near 10 Hz? The spectra in figure 4.2 are hardly conclusive. Is the bump near 10 Hz an interesting feature or an artifact of the analysis? To address this question, we next consider the issue of windowing and computing the spectrum.

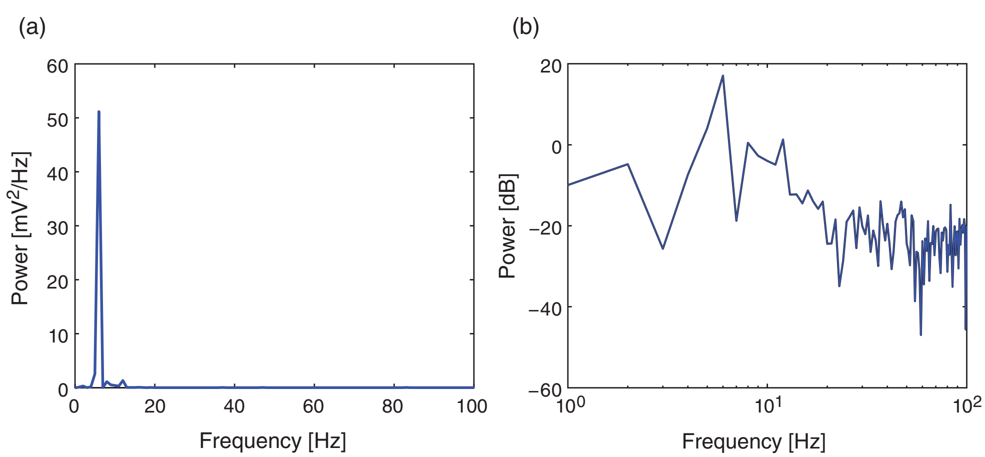

Figure 4.2
Power spectrum of ECoG data on (*a*) linear scale and (*b*) logarithmic scale.

10234_004 2016/7/12 12:11 Page 97 #5

Q: Are the terms *frequency resolution, Nyquist frequency, Fourier transform, decibel*, and *spectrum* familiar? Can you define or explain each term?

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

By Doing Nothing, We're Doing Something: The Rectangular Taper. ECoG time series continue for long durations. For example, an individual's brain voltage activity may persist for over 90 years, from birth until death. However, ECoG recordings are finite, limited by convenience, technology, or other factors. In the example here, we consider 1 s of ECoG data. Performing this finite observation (lasting 1 s) on a long duration (i.e., 90-year) time series can be understood as a *rectangular taper*. A rectangular taper multiplies the observed data by 1 and the unobserved data by 0 (figure 4.3). We can think of the value 1 as representing the time period when our recording device is operational; activating the ECoG recording device opens the rectangular taper (value 1), and deactivating the ECoG recording device closes the rectangular taper (value 0). The rectangular taper makes explicit our knowledge about the observed data (in this case, the 1 s interval of ECoG) and our ignorance about the unobserved data, which are assigned the value zero. Notice that the rectangular taper looks like a rectangle (figure 4.3).

So, by computing the spectrum of 1 s of observed ECoG data, we're actually computing the spectrum of the product of two functions: the many years of mostly unobserved ECoG data and the rectangular taper. We note that by "doing nothing" we have implicitly made the choice to use the rectangular taper. We have already plotted the resulting spectrum of the observed ECoG data (again, using the default rectangular taper) in figure 4.2.

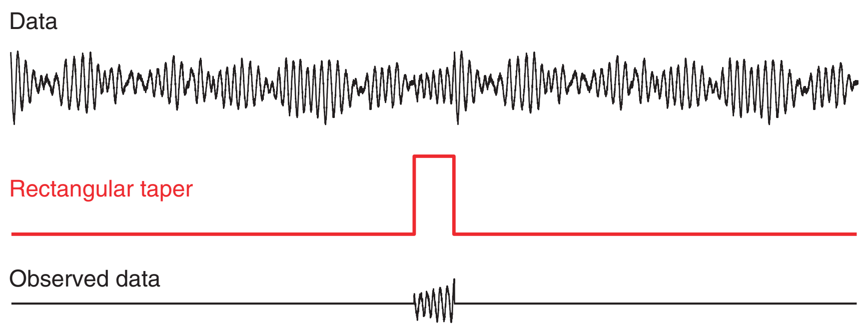

Figure 4.3

Example of rectangular taper application. Raw data continue for a long period of time (*top*). Most of these data are unobserved. Rectangular taper (*red*) specifies the interval of observation. Multiplying the raw data by the rectangular taper determines the observed ECoG data (*bottom*).

10234_004 2016/7/12 12:11 Page 98 #6

Exploring the Impact of the Rectangular Taper. Figure 4.3 illustrates how the rectangular taper impacts the observed data in the time domain, namely, the taper selects a region of observation. The rectangular taper also impacts the spectrum in the frequency domain. To see this, consider a perfect sinusoid at frequency 10 Hz that in theory continues forever. In this case, the energy concentrates at a single frequency—the frequency of the sinusoid (10 Hz)—and for the (theoretical) case of an infinite sinusoid, the power spectral density is infinite at that frequency (figure 4.4a). In mathematical language, we say that the spectrum of the infinite sinusoid is a *delta function*. However, we never observe an infinite sinusoid; to do so would require unlimited resources and unlimited time. Instead, let's assume we observe only 1 s of the sinusoid's activity; we imagine multiplying the infinite duration sinusoid by a rectangular taper and observing only a finite interval of time (figure 4.4b). The corresponding spectrum of the (now finite) sinusoid is shown in Figure 4.4b.

Q: Examine the spectrum of the finite sinusoid plotted in figure 4.4b. What do you see? Is the spectrum concentrated at one frequency (near 10 Hz), as we expected for an infinite sinusoid?

A: Visual inspection of the spectrum suggests an unexpected result: the spectrum is *not* concentrated at a single frequency. Instead, many peaks appear, centered at the expected frequency (10 Hz) but also spreading into neighboring frequency bands. The culprit responsible for this leakage outside of the 10 Hz peak is the rectangular taper applied to the true infinite-duration sinusoidal signal.

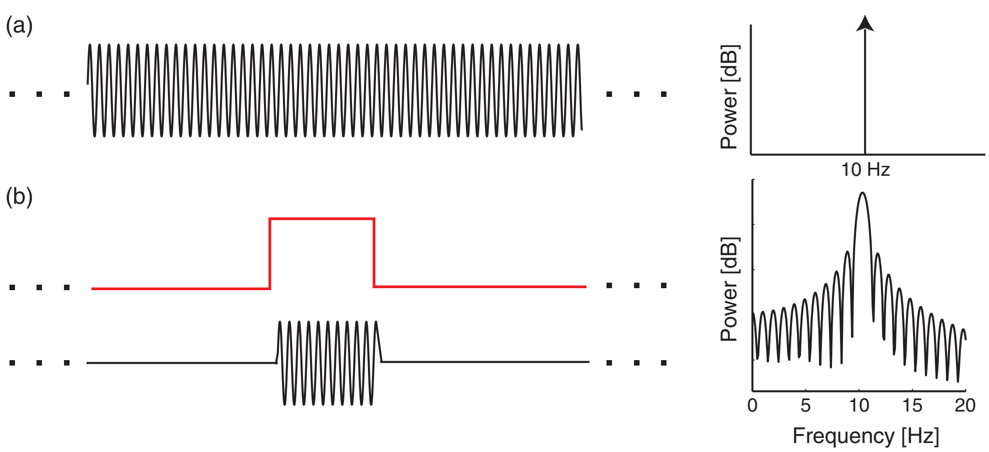

Figure 4.4

Example of rectangular taper application to a sinusoid. (*a*) Infinite sinusoid continues forever in time; energy concentrates at sinusoid's frequency of 10 Hz. (*b*) Multiplying the infinite duration sinusoid by a rectangular taper (*red*) yields a sinusoid of finite duration. Spectrum of the resulting signal (*black*) exhibits features at many frequencies, with peak at 10 Hz.

To understand further the impact of the rectangular taper, let's consider the spectrum of the rectangular taper itself.

Q: Examine the spectrum of the rectangular window plotted in figure 4.5b. What do you see? At what frequency is the spectrum concentrated?

To compute the spectrum of the rectangular taper, we consider the time series shown in figure 4.5a. In theory, the rectangular taper is infinite, and is preceded and followed by infinite intervals of zeros (i.e., there's an infinite period in which we do not observe the ECoG data). To represent the infinite extent of the rectangular taper, we add 10 s of zeros to the beginning and end of a 1 s interval of ones. Of course, 10 s of zeros is a poor representation of an infinite interval of time, but it's sufficient for our purposes here. We note that the rectangular taper consists of two sharp edges, when the observation interval opens and closes (i.e., transitions from 0 to 1, and then back from 1 to 0).

The spectrum of the rectangular taper is shown in figure 4.5b. Visual inspection suggests that most of the power spectral density is concentrated at a single frequency, 0 Hz. Notice that in this case we've plotted the spectrum at positive and negative frequencies. The negative frequencies are redundant (because the signal is real; see chapter 3), but the symmetry helps visualize the results. To understand why the spectrum concentrates at 0 Hz, consider the rectangular taper over the 1 s duration for which the data collection window is open (figure 4.5a). Within this window, the value of the taper is the constant 1. The only frequency present in this signal is 0 Hz (i.e., no oscillations occur). Therefore, the spectrum is concentrated at 0 Hz.

Although the spectrum is concentrated at 0 Hz, it exhibits regions of increased spectral density at nonzero frequencies, more specifically, the repeated peaks (or side lobes) in the

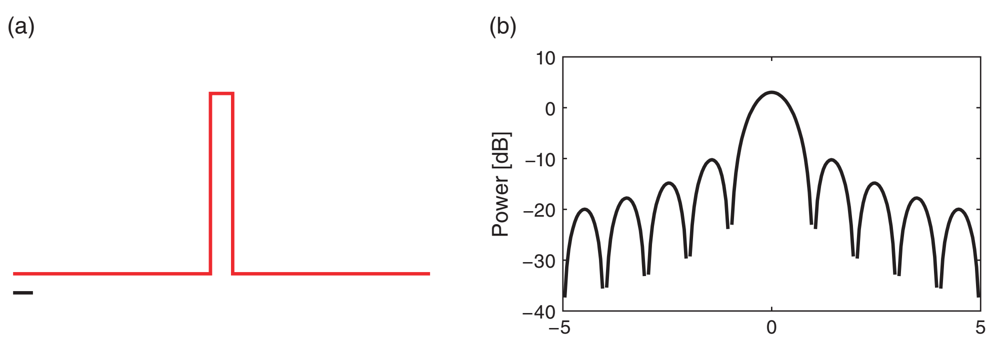

Figure 4.5

The rectangular taper possesses its own spectrum. (*a*) The rectangular taper; here, we set the taper equal to zero for 10 s, then equal to one for 1 s, then equal to zero for 10 s. The black line indicates a 1 s scale bar. (*b*) The spectrum of the rectangular taper; note in this case we plot both positive and negative frequencies.

spectrum near ±1.5 Hz, ±2.5 Hz, ±3.5 Hz, and so on. In fact, it's possible to work out exactly the functional form of the Fourier transform of the rectangular taper; it's the sinc function [8]. However, that's not particularly important for our purposes. The important result here is that the rectangular taper itself has a complicated spectrum, with features appearing across a range of frequencies.

Q: Why does the spectrum of the rectangular taper contain features at many frequencies?

A: The spectrum consists of features at many frequencies to represent the rapid increase and decrease of the rectangular taper (i.e., the rapid transition from 0 to 1, and then from 1 to 0). The intuition for this is that many sinusoids, aligned in a particular way, are required to represent a sharp transition in a time series. As an example, consider the square wave function in figure 4.6. This square wave consists of many repeated sharp transitions. Visual inspection suggests that the square wave is rhythmic, with a period of 2 Hz. Notice that the square wave begins at a sustained value of 1, then transitions to 0 and remains there for an interval of time, and then transitions back to a value of 1 in 0.5 s; the square wave therefore completes two cycles in 1 second.

So, the square wave is rhythmic, and we may try to represent this square wave with rhythmic sinusoids. In figure 4.6a, we plot a 2 Hz sinusoid. It's an okay match to the square wave but certainly not perfect; the 2 Hz sinusoid fails to capture the sharp transitions of the square wave. In figure 4.6b, we plot the sum of a 2 Hz sinusoid and a 6 Hz sinusoid. The combination of these two sinusoids better matches the square wave, although again the sharp transitions are not accurately captured. In figure 4.6c, we plot the sum of 2 Hz, 6 Hz, 10 Hz, 14 Hz, and 18 Hz sinusoid. These five sinusoids better match the square wave and begin to more accurately capture the sharp transitions. As more sinusoids are used to represent the square wave, the approximation of the sharp edges improves.

So, sharp transitions in data require many sinusoids to be accurately approximated. The rectangular taper consists of sharp transitions and therefore requires sinusoids at many frequencies for an accurate representation.

To understand why the spectrum of the rectangular taper is important, let's return to the example of an infinite sinusoid. The (theoretical) spectrum for this sinusoid concentrates all the spectral density at a single frequency (i.e., the frequency of the sinusoid), as illustrated in figure 4.4a. However, we never observe an infinite sinusoid; instead, we observe this sinusoid multiplied by the rectangular taper (figure 4.4b). This multiplication of the two functions impacts the spectrum we observe. To see why, we state an important property of the Fourier transform.

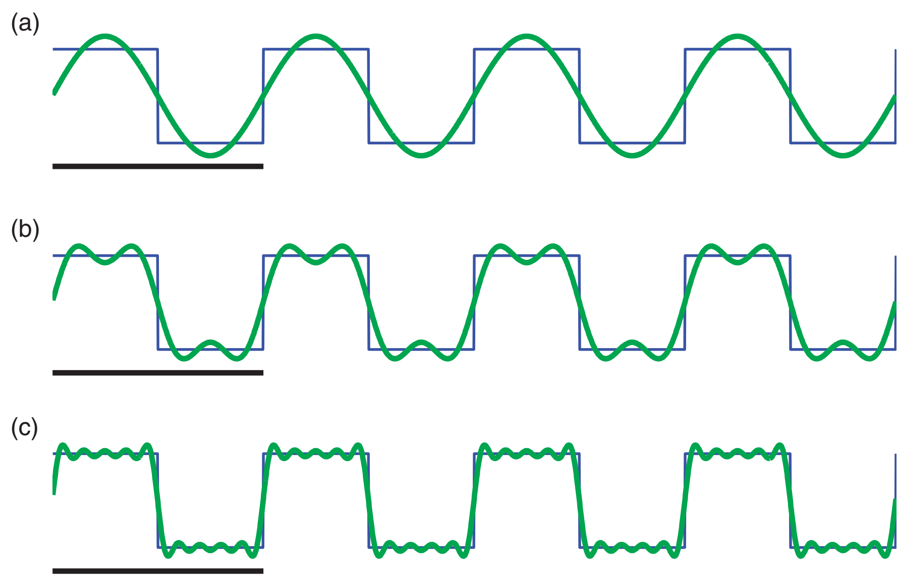

Figure 4.6

Approximation of a square wave with sinusoids. Square wave (*blue*) has a period of 2 Hz. As the number of sinusoids used to approximate the square wave increases from (*a*) one sinusoid, to (*b*) two sinusoids, to (*c*) five sinusoids, approximation improves. Scale bar indicates 0.5 s.

> Multiplication in the time domain is equivalent to *convolution* in the frequency domain.

Because we perform multiplication of two functions in the time domain (i.e., we multiply the rectangular taper and the infinite sinusoid element by element), we produce a convolution of the Fourier transforms of these functions in the frequency domain. Although the mathematical expression of convolution is somewhat complicated, we can understand the impact of convolution by examining a few plots. Let's start with the Fourier transform of the infinite sinusoid and the infinite rectangular taper (where we've appended zeros to the rectangular taper to make it infinite). We know (or can look up) the results of each Fourier transform. The Fourier transform of the infinite 10 Hz sinusoid, which we assume here is a cosine function, consists of two delta functions at ± 10 Hz (figure 4.7a). The Fourier transform of the rectangular taper is the sinc function (figure 4.7b). Now, let's imagine shifting in frequency the Fourier transform of the rectangular taper (i.e., shifting in frequency the sinc function). At each shift, we multiply element by element the two Fourier transforms (i.e., the unshifted Fourier transform of the sinusoid and the shifted Fourier transform of the rectangular taper) and sum the product. The result of this shifting, multiplying, and summing of one Fourier transform by the other is the convolution.

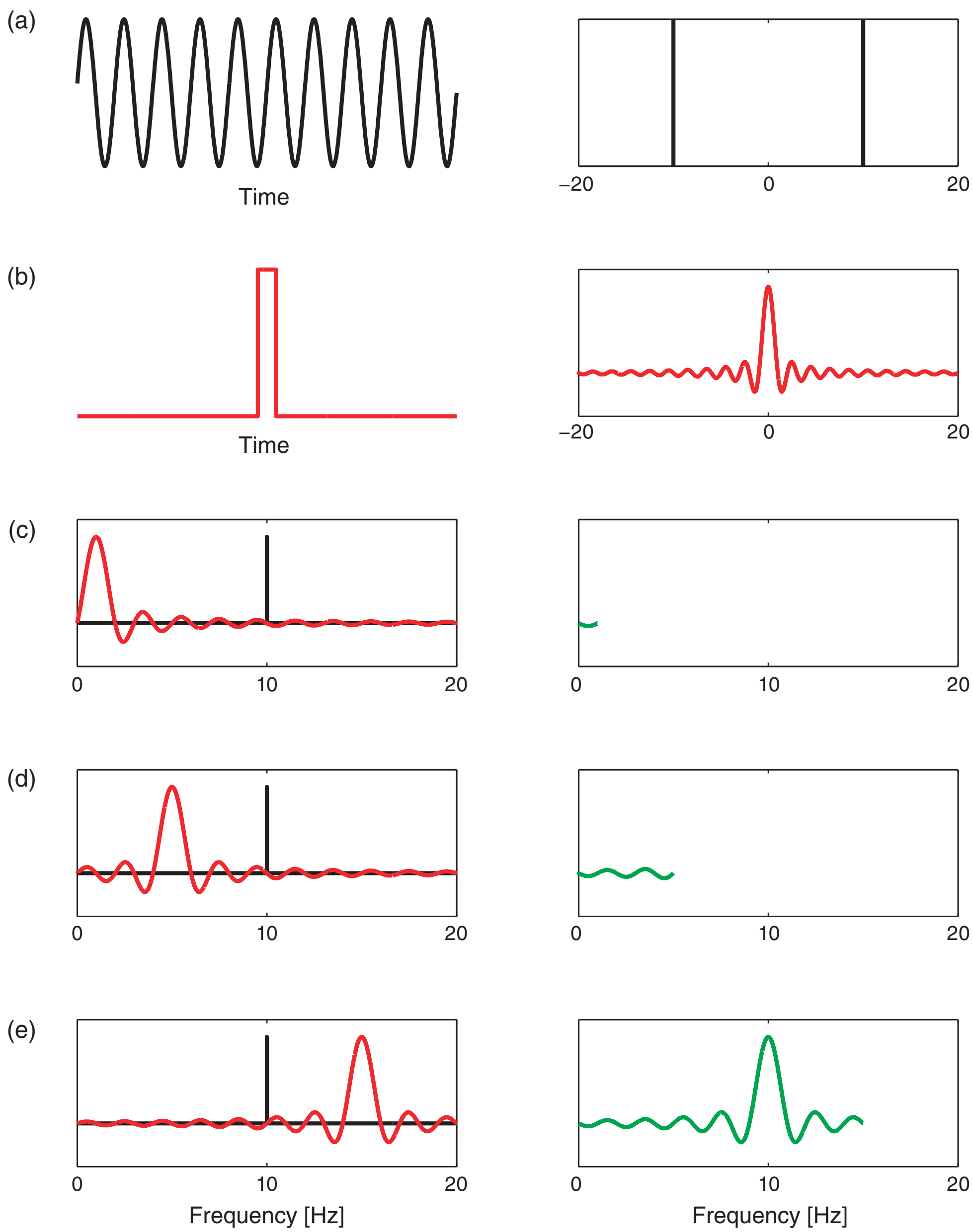

Figure 4.7

The impact of the rectangular taper on the Fourier transform of a sinusoid can be understood by considering the convolution between the two functions. (*a,b*) The sinusoid (*black*) and rectangular taper (*red*) time series (*left*) and Fourier transforms (*right*). (*c–e*) Illustration of the Fourier transform of each function (*left*) and their convolution evaluated at different shifts (*right in green*). Shifts up to 1 Hz (*c*), 5 Hz (*d*), and 15 Hz (*e*) are shown.

Q: What is the result of this convolution? How is the expected Fourier transform of the infinite sinusoid affected?

To illustrate this convolution procedure, we plot in figures 4.7c–e examples of shifting and multiplying the two Fourier transforms. Figure 4.7c shows the case in which the rectangular taper's Fourier transform is shifted by 1 Hz. Because the sharp peak of the sinusoid's Fourier transform does not overlap the large central peak of the rectangular taper's Fourier transform, the product of the two functions remains small.

As we continue to shift the rectangular taper's Fourier transform, the product of the two functions at first remains small (for shifts up to 5 Hz in figure 4.7d). However, once we shift the rectangular taper's Fourier transform up to and through the 10 Hz peak of the sinusoid, we begin to find larger deviations in the convolution. As the shifted rectangular taper's Fourier transform passes through the 10 Hz peak of the sinusoid, we multiply the very thin and very tall peak of the sinusoid (figure 4.7e) with the center portion of the shifted rectangular taper's Fourier transform. The result of this multiplication is to "smear" the sharp peak of the sinusoid in the frequency domain. Instead of a sharp peak at 10 Hz, the convolution produces a broad, wiggly peak in the frequency domain, with deviations at neighboring frequencies (i.e., in the side lobes around 10 Hz). In other words, because we observe the sinusoid over a finite interval of time (determined by the extent of the rectangular taper) the Fourier transform of the sinusoid becomes "smeared."

This description and the illustrations in Figure 4.7 provide some intuition for the relation between the multiplication of two signals in the time domain (here, a sinusoid and the rectangular taper) and their Fourier transform. For mathematical details and corresponding MATLAB code, see the appendix at the end of this chapter.

Zero Padding. An interesting issue to consider is how appending zeros impacts the spectrum of the rectangular taper. We know that increasing the signal length (T) improves the frequency resolution; recall the equation $df = 1/T$ from chapter 3. We therefore expect that adding more points to the signal (even noninformative points, such as zeros) will increase the number of points along the frequency axis. However, appending zeros to a time series is *not* equivalent to observing more data (and thereby increasing T). By appending zeros, we of course do not gain any additional information about the signal. Therefore appending zeros to a signal *cannot* improve the frequency resolution. Instead, the impact of appending zeros is to increase the number of points along the frequency axis in the spectrum. This can be useful in visualizing the spectrum; for example, by appending more and more zeros to the rectangular taper, we produce a less jagged spectrum (figure 4.8).

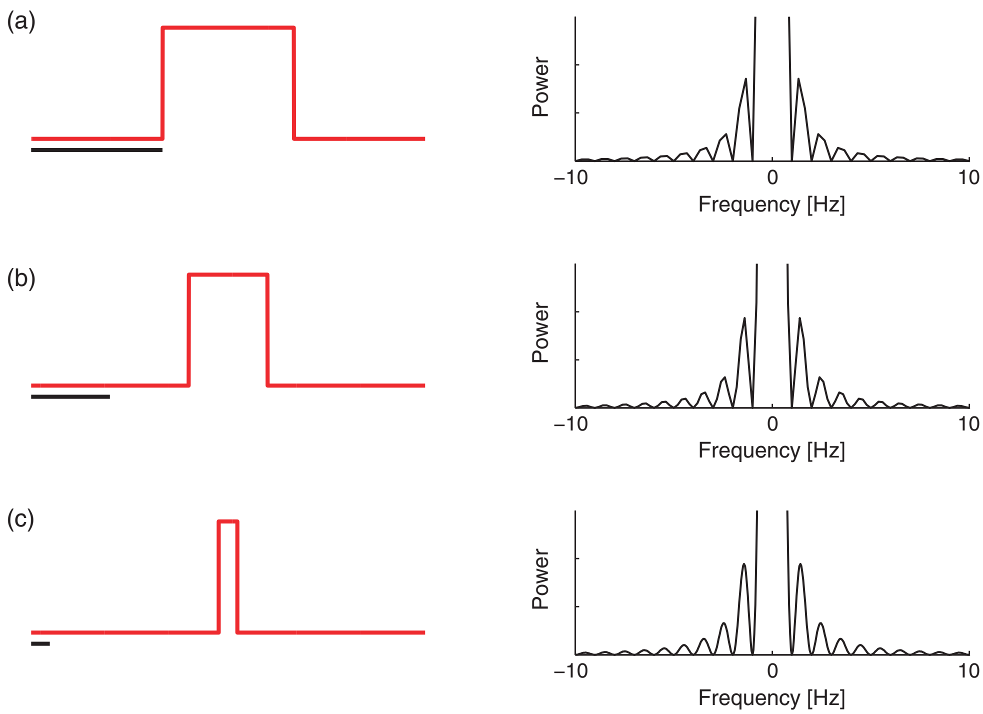

Figure 4.8

Impact of appending zeros on the spectrum of the rectangular taper. *Left*, Rectangular tapers vs. time; scale bars indicate 1 s. *Right*, Spectra of the rectangular tapers. (*a*) One second of zeros surrounding 1 s interval of ones; (*b*) two seconds of zeros surrounding 1 s interval of ones; (*c*) ten seconds of zeros surrounding 1 s interval of ones. As the number of zeros increases, the spectrum becomes smoother.

This procedure of appending zeros to a time series is called *zero padding*. It can be useful for visualizing a spectrum. But we must be careful to remember the following important fact.

> The frequency resolution of the spectrum is fixed by the amount of data recorded. The number of points along the frequency axis in the spectrum is adjustable and can be increased by zero padding.

With this understanding of the impact of the rectangular taper and zero padding, let's now return to the spectrum of the 1 s of sinusoidal activity (see figure 4.4b). We now expect that because we observe the infinite sinusoid for only a short duration (1 s), the spectral power at 10 Hz will leak into neighboring frequency bands. And as plotted in figure 4.4b, that's indeed what we find. In figure 4.4b the side lobe structure is clearly visible; in this example, we computed the spectrum with zero padding to evaluate the spectrum at many points along the frequency axis.

To explore further the impact of this zero padding, let's now consider an example in MATLAB. We define a 10 Hz sinusoid with duration 1 s, apply 10 s of zero padding and examine the impact on the spectrum:

```
Fs = 500;                        %Define sampling frequency.
dt = 1/Fs;                       %Define sampling interval.
t = (dt:dt:1);                   %Define time axis.
T = t(end);                      %Define total time.

d = sin(2.0*pi*t*10);            %Make a 10 Hz sinusoid,
d = [d, zeros(1,10*Fs)];         %... with 10 s of zero padding.

df = 1/(length(d)*dt);           %Define the frequency step size,
fNQ = Fs/2;                      %... and Nyquist frequency,
faxis = (0:df:fNQ);              %... to create frequency axis.

pow = 2*dt^2/T*(fft(d).*conj(fft(d)));%Compute spectrum.
pow = pow(1:length(d)/2+1);      %Ignore negative frequencies,
plot(faxis, 10*log10(pow))       %... and plot it,
xlim([0 20])                     %... in selected frequency range,
ylim([-60 10])                   %... and selected decibel range,
xlabel('Frequency [Hz]')         %... with axes labeled.
ylabel('Power [dB]')
```

In the first four lines of this code, we define the sampling frequency (`Fs`), sampling interval (`dt`), time axis (`t`), and duration of data (`T`) to simulate. We then define the 10 Hz sine function and include zero padding. The remaining lines evaluate the spectrum and display the results (as done for the preceding ECoG data). Notice that we plot the spectrum on a decibel scale, and focus on the frequency interval 0–20 Hz.

Q: We usually subtract the mean of a signal before computing the Fourier transform, but we did not do so here when defining the variable `pow`. Is that a problem?

A: In this case, the mean of `d` is zero, so subtracting the mean does not impact the results.

Q: How much zero padding do we add? How does this change the spacing on the frequency axis?

A: We append `10*Fs` zeros to the end of the simulated sinusoidal data. This corresponds to 10 s (or 5,000 points). The original spacing on the frequency axis for the data was $1/T = 1/1$ s = 1 Hz. After zero-padding, the spacing on the frequency axis becomes $1/(1 + 10\text{s}) = 1/(11\text{s}) \approx 0.091$ Hz.

The results for this case (using 10 s of zero padding) are shown in figure 4.9a. We find a dominant peak at 10 Hz, as expected, and large side lobes that extend throughout the 0–20 Hz frequency range. Choices of smaller-duration zero padding (5 s in figure 4.9c) and longer-duration zero padding (100 s in Figure 4.9d) produce similar results. With 5 s of zero-padding, the plotted spectrum appears less smooth; in this case, we evaluate the

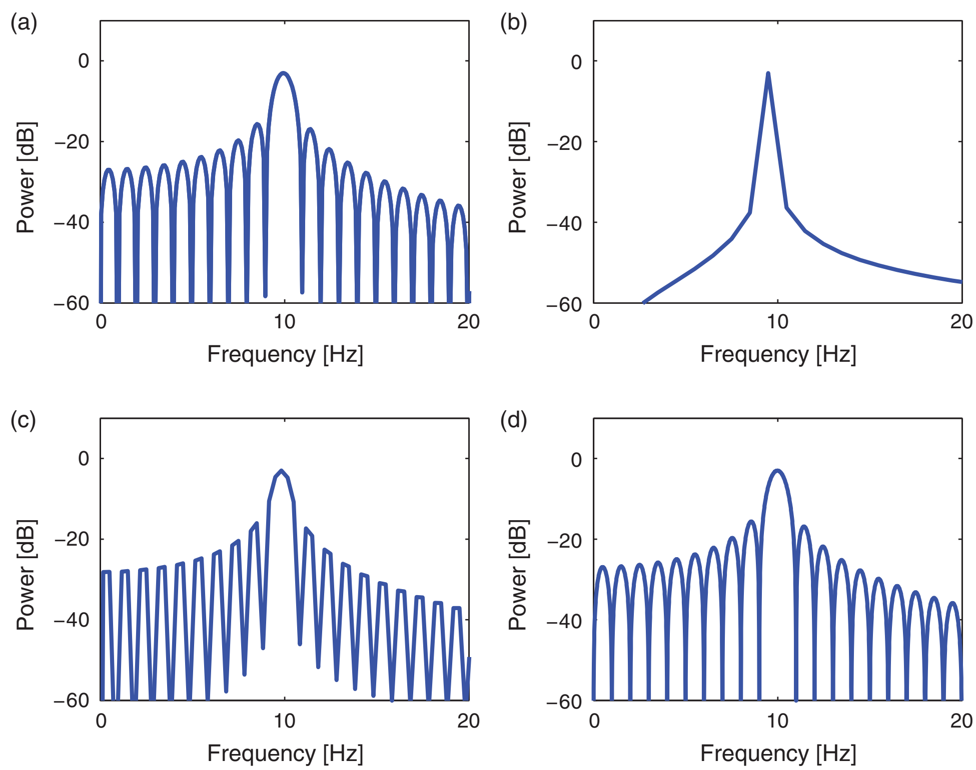

Figure 4.9
Impact of zero padding duration on the spectrum of a 1 s, 10 Hz sinusoid. Duration of zero paddings: (*a*) 10 s, (*b*) 0 s, (*c*) 5 s, (*d*) 100 s.

spectrum at fewer points on the frequency axis, and the side lobe peaks become more jagged. The choice of 100 s of zero padding does not produce a qualitative change compared to the 10 s of zero padding; in this case, evaluating the spectrum at 10 times as many points along the frequency axis is not particularly useful.

We also show in figure 4.9b the impact of no zero padding. To do so, we omit the sixth line in the preceding MATLAB code (i.e., `d = [d, zeros(1,10*Fs)];`). The spectrum is quite different from the original example. Without zero padding, the falloff from the 10 Hz peak appears rather smooth and gradual, and does not exhibit obvious side lobe structure, as expected after the discussion of the rectangular taper and example in figure 4.7. Leakage from the 10 Hz peak into neighboring frequency bands still occurs, but without zero padding we're sampling the frequency axis too coarsely to accurately make out the side lobes.

Zero padding and Frequency Resolution: An Example. We stated in the previous section that zero padding does not improve the frequency resolution of the spectrum. As an example of this, consider a simple signal of duration 1 s that consists of two sinusoids: a 10 Hz sinusoid and a 10.5 Hz sinusoid.

Q: Given this 1 s time series, can we distinguish the two rhythms in the spectrum?

A: No. The frequency resolution is $df = 1/T = 1/(1\text{s}) = 1$ Hz. Because the two sinusoids are separated by less than 1 Hz, we cannot distinguish these two rhythms.

But, perhaps by zero padding the data, we can distinguish these two rhythms. After all, zero padding acts to increase T, right? We're appending lots of zeros to the time series, so the data become longer .... Let's create these synthetic data in MATLAB and investigate.

```
Fs = 500;                    %Define sampling frequency.
dt = 1/Fs;                   %Define sampling interval.
t = (dt:dt:1);               %Define time axis.
T = t(end);                  %Define total time.

d1 = sin(2.0*pi*t*10);       %Make a 10 Hz sinusoid.
d2 = sin(2.0*pi*t*10.5);     %Make a 10.5 Hz sinusoid.
d  = d1+d2;                  %Make the summed signal,
d = [d, zeros(1,10*Fs)];     %... with 10 s of zero padding.

df = 1/(length(d)*dt);       %Define the frequency step size,
fNQ = Fs/2;                  %... and Nyquist frequency,
faxis = (0:df:fNQ);          %... to create frequency axis.
```

```
pow = 2*dt^2/T*(fft(d).*conj(fft(d)));%Compute spectrum.
pow = pow(1:length(d)/2+1);     %Ignore negative frequencies,
plot(faxis, 10*log10(pow))      %... and plot it,
xlim([0 20])                    %... in selected frequency range,
ylim([-40 10])                  %... and selected decibel range,
xlabel('Frequency [Hz]')        %... with axes labeled.
ylabel('Power [dB]')
```

In the first four lines, we define some useful parameters, including the sampling frequency and a time axis. We then create the two sinusoids (`d1` and `d2`), sum these sinusoids to create a composite signal (`d`), and zero-pad this composite signal with 10 s of zeros. The rest of the code computes and plots the spectrum, which is shown in figure 4.10a.

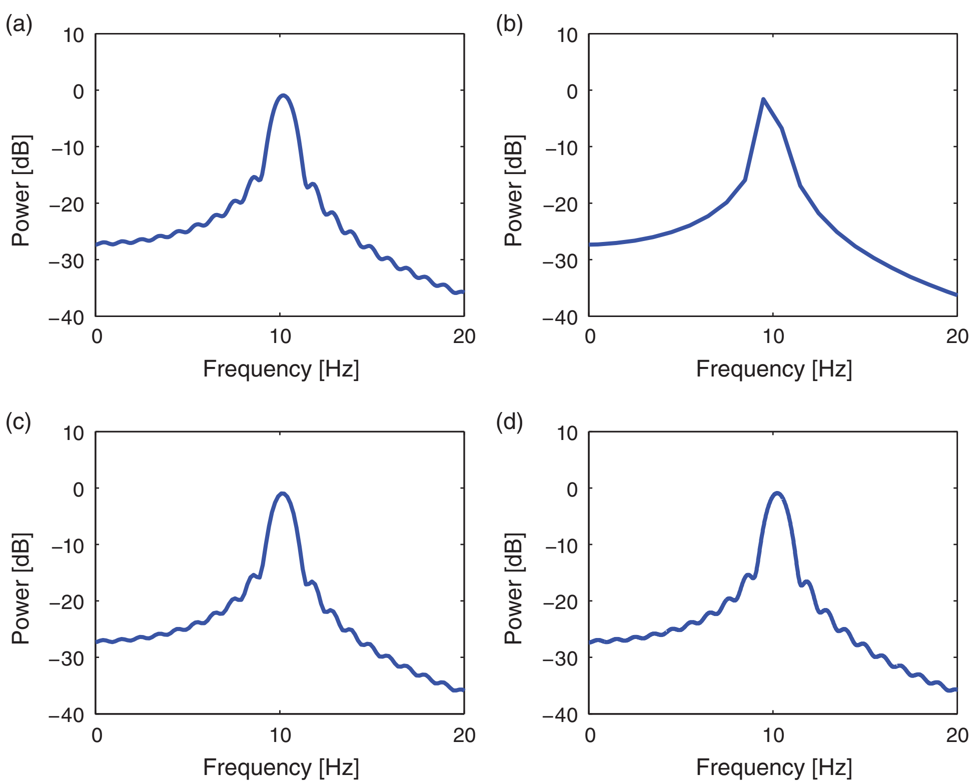

Figure 4.10

Impact of zero padding duration on the frequency resolution of the spectrum. The data consist of the sum of two sinusoids at frequencies 10 Hz and 10.5 Hz, and last for 1 s. Duration of zero padding is: (*a*) 10 s, (*b*) 0 s, (*c*) 5 s, (*d*) 100 s.

Q: Consider the spectrum in figure 4.10a. Can you identify the two rhythms present in the simulated time series? The other graphs show the spectrum with different durations of zero padding. Do any of these figures reveal the two separate rhythms?

A: No. No choice of zero padding resolves the two spectral peaks. In this case, the two rhythms are separated by less than the frequency resolution df. Zero padding the data (even with 100 s of zeros) does not resolve the two peaks. We only simulated 1 s of data, and therefore set $df = 1$ Hz; zero padding does not change this fact.

4.2.3 Beyond the Rectangular Taper—The Hanning Taper

We have considered so far a single type of taper (the default, a rectangular taper) and its impact on the spectrum. In particular, we noted the "smearing" of spectral peaks (i.e., side lobes) that impact neighboring frequency bands. At best these side lobes are distracting, and at worst they may contaminate our conclusions. For example, consider the spectrum for the 1 s of ECoG data plotted in figure 4.2. Is the small peak near 10 Hz representative of a true rhythm in these data, or is it a side lobe of the large peak near 6 Hz? Many different taper shapes have been developed with the goal of reducing the side lobes that contaminate the signals. Here we consider one of these tapers, the *Hanning taper*.

The problem with the rectangular taper is its sharp edges (i.e., the rapid transitions from 0 to 1, and from 1 back to 0). To represent these sharp edges in the frequency domain requires many sinusoids, oscillating at different frequencies (e.g., figure 4.6), which manifest as side lobes in the spectrum. The Hanning taper acts to smooth the sharp edges of the rectangular taper. To see this, we plot in figure 4.11a both the Hanning taper and the rectangular taper. Notice that the Hanning taper gradually increases from zero, reaches a maximum of 1 at the center of the interval, then gradually decreases to zero. The corresponding spectra of the Hanning taper and the rectangular taper are plotted in figure 4.11b. These spectra reveal two main differences between these tapers: (1) the central lobe of the Hanning taper is wider than the rectangular taper, and (2) the side lobes in the Hanning taper are reduced compared to the rectangular taper. These two features illustrate the trade-off between the two window choices. By accepting a wider central peak in the Hanning taper, we acquire side lobes with lower power.

The Hanning taper is applied to time series data in the same way as the rectangular taper. The data are multiplied element by element by the taper. Let's compute and apply the Hanning taper in MATLAB for the ECoG data:

```
load('Ch4-ECoG-1.mat')        %Load the ECoG data.
x  = ECoG;                    %Relabel the data variable.
N = length(x);                %Define no. of points in data.
x = hann(N).*x;               %Multiply data by Hanning taper.
```

The function `hann` returns the Hanning taper, which we multiply element by element with the ECoG data in variable `x`. The results of this multiplication are plotted in figure 4.11c. Notice that the slow increase in the Hanning taper reduces the amplitude of the ECoG activity near the taper edges and emphasizes the ECoG activity in the center of the interval.

Q: Under what conditions would reducing the activity near the taper edges be a bad idea? *Hint:* What if the signal features of interest occur at the very beginning or very end of the observed data?

Q: The spectra for the Hanning-tapered ECoG data and the rectangular tapered ECoG data are plotted in figure 4.11d. What conclusions do you now draw regarding the rhythms present in the ECoG data? Consider, in particular, the activity near 10–15 Hz.

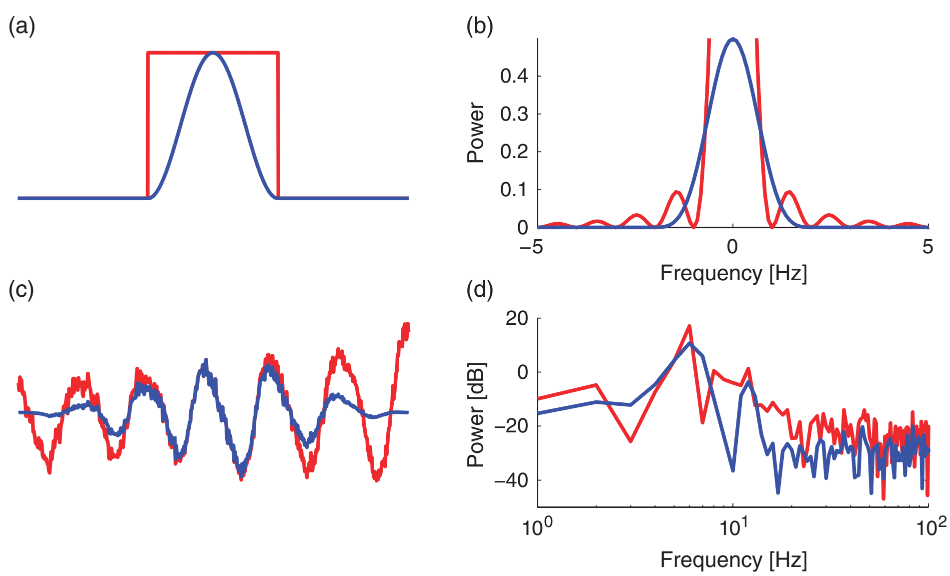

Figure 4.11

Hanning taper (*blue*) and the rectangular taper (*red*) have different trade-offs in the time and frequency domains. (*a*) The two tapers and (*b*) their corresponding spectra. (*c*) The tapers applied to the ECoG data, and (*d*) the resulting spectra.

A: The spectrum of the Hanning-tapered ECoG data reveals a peak at 10–15 Hz. This peak was hidden by the side lobes of the 6 Hz peak in the (default) rectangular-tapered ECoG data. The Hanning taper reduces the side lobes of the 6 Hz peak, and allows us to uncover the smaller 10-15 Hz peak that was originally obscured by these side lobes. This observation dramatically changes our interpretation of the ECoG data. We now propose that the ECoG activity consists of two rhythms: a rhythm near 6 Hz, and a second band of rhythms near 10–15 Hz. Without application of the Hanning taper, we might have missed the second rhythm by attributing it to side lobes of the 6 Hz peak.

4.2.4 Beyond the Hanning Taper—The Multitaper Method

The Hanning taper provides a nice alternative to the rectangular taper. If we're willing to allow slightly broader spectral peaks in the frequency domain, and lose some data near the taper edges in the time domain, then the Hanning taper helps reduce the impact of side lobes. We now consider a brief introduction to a more advanced approach to tapering, the multitaper method. The idea of the multitaper method is to apply multiple tapers to the data, each with a different shape. The spectrum is then computed for each taper, and the results averaged over the tapers. Each taper, which is given by portions of the discrete prolate spheroidal sequences, provides an independent estimate of the (theoretical) spectrum. Therefore, the variance of the average estimate over the tapers is $1/K$ times the variance of the estimated spectrum from a single taper, where K is the number of tapers used. The multitaper method has a number of additional advantages, such as minimizing the bias due to other spectral peaks outside of the frequency band being considered. We do not discuss these properties in detail, but more information can be found in [8, 9].

In the preceding examples, we estimated the spectrum from an observed time series. Often from these estimates we have difficulty identifying features that are significant (or not). For example, consider the small peak at 40–50 Hz in figure 4.11d. Is that peak significant or a random fluctuation we expect in an estimate from real-world ECoG data? To address this question, we might consider collecting more data and using these additional data to improve our estimate of power. However, as we mentioned in chapter 3, collecting more data does not automatically improve the spectral estimate. Instead, collecting more data (i.e., increasing the duration of data recorded, T) produces more spectral estimates at additional frequencies, yet the spectrum at each frequency remains just as variable. We would like a way to improve the spectral estimate—a way to reduce the variability of the estimate—so that we could more confidently identify interesting features. The multitaper method offers a procedure to do so. However, this comes at a cost. If we desire reduced variance in the spectral estimate, we must accept worse frequency resolution. Let's explore these issues in more detail.

To get a sense for the multitaper method, let's examine some of the tapers, as plotted in figure 4.12a. The first taper looks familiar—it's similar to the Hanning taper. Notice that the first taper starts at zero, increases throughout the interval, and then decreases back to zero. As we add more tapers, each contains more "wiggles." In particular, as the number of tapers increases, the edges of the data become better represented. Unlike the Hanning taper, which slowly approaches zero at the taper edges, some of these tapers increase near the window edges (see taper 5 in figure 4.12a). This can be useful, especially if we are interested in the spectral features near the beginning and end of the data. Just as with the Hanning taper, we multiply the time series data by each taper, which emphasizes different intervals of the data. We show examples of this multiplication for the ECoG data in figure 4.12b. The first taper emphasizes the middle of the ECoG data, and the fifth taper emphasizes the beginning and end of the ECoG data.

We only show the first five tapers in figure 4.12. There are an infinite number more. How do we choose the number of tapers to use in the multitaper method? To answer this, we utilize the equation

$$TW = X, \tag{4.1}$$

where T is the duration of the recording, $2W$ is the desired frequency resolution (or resolution bandwidth), and we're free to choose X, the aptly named *time-bandwidth product*. For concreteness, let's consider the 1 s of ECoG data.

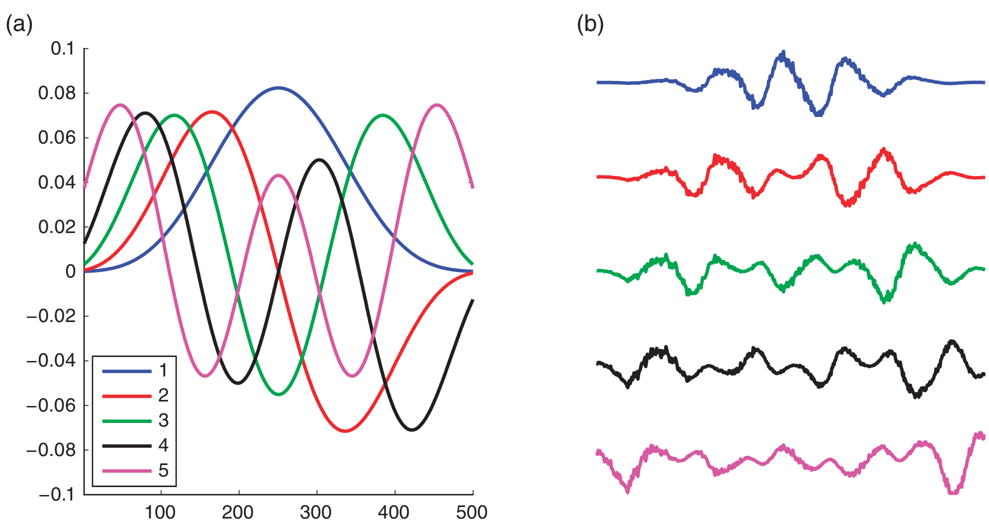

Figure 4.12
(*a*) Traces of the first five tapers, plotted as a function of indices, and (*b*) each taper applied to 1 s of ECoG data.

Q: Given 1 s of data, what is the frequency resolution?

A: Using the equation for the frequency resolution $df = 1/T$, we find $df = 1/(1\text{s}) = 1$ Hz. Without applying the multitaper method, we begin with a frequency resolution of 1 Hz. We now concede some frequency resolution to apply the multitaper method and reduce the variability in the spectral estimate.

Remember that a frequency resolution of 1 Hz indicates that we can resolve features in the spectrum separated in frequency by 1 Hz or more. For example, at a frequency resolution of 1 Hz, we can distinguish 10 Hz activity from 9 Hz or 11 Hz activity. However, we are unable to distinguish 10 Hz activity from 10.5 Hz or 9.5 Hz; those frequencies lie within the frequency resolution.

Let's assume we do not require a frequency resolution of 1 Hz; instead, we are satisfied with a frequency resolution of 6 Hz. Using the multitaper method, we trade off a worse frequency resolution to improve the estimate of the spectrum. In this case, $T = 1$ s, and we are willing to accept a resolution bandwidth of $2W = 6$ Hz. So, from (4.1), we compute the time-bandwidth product and find $TW = 3$. Now, with this value, we select the number of tapers following this rule of thumb:

$$\text{No. of tapers} = 2TW - 1. \tag{4.2}$$

We choose the first $2TW - 1$ tapers because doing so allows us to preserve most of the information present in the original data. We could choose fewer tapers, but in most cases we follow this rule of thumb and pick as many tapers as we can. Choosing more tapers does not improve the multitaper estimate of the spectrum and may lead to spurious results; for more details, see [8]. So, for the ECoG data of interest here, we select the number of tapers to be $2 \times 3 - 1 = 5$. These five tapers are plotted in figure 4.12a. Applying these tapers to the ECoG data, we create the five time series shown in figure 4.12b. We then compute the spectrum of each tapered ECoG time series, and average the resulting spectra across the five tapers. Through this averaging procedure across tapers, we reduce the variability of the spectral estimate.

Multitaper method tradeoff: The multitaper method permits a trade-off between frequency resolution and variance of the spectrum. If we can tolerate worse frequency resolution, we can include more tapers and reduce the spectrum variance.

Q: Given 10 s of data, we demand a frequency resolution of 2 Hz or better. Using the multitaper method, what is the maximum number of tapers we could select and still maintain the desired frequency resolution?

10234_004 2016/7/12 12:11 Page 114 #22

The multitaper method is a sophisticated approach, and we have only touched on the surface in this brief discussion. There are many detailed references and important applications [8, 9]. Fortunately, there are also MATLAB software packages and functions to implement and apply the multitaper method. In subsequent chapters, we utilize the software package Chronux (chronux.org) [2]. But for now, let's use the MATLAB function `pmtm` to compute the multitaper spectrum of the ECoG data:

```
load('Ch4-ECoG-1.mat')      %Load the ECoG data.
x  = ECoG;                  %Relabel the data variable.
x  = x - mean(x);           %Set mean of x to zero.
dt = t(2)-t(1);             %Define the sampling interval.
Fs = 1/dt;                  %Define the sampling frequency.
TW = 3;                     %Choose time-bandwidth product of 3.

[Sxx,f]=pmtm(x,TW,length(x),Fs); %Compute MTM spectrum.

semilogx(f, 10*log10(Sxx))  %Plot spectrum vs. frequency,
xlim([0 100])               %... in selected frequency range,
xlabel('Frequency [Hz]')    %... with axes labeled.
ylabel('Power [dB]')
```

We define the time-bandwidth product `TW`. By default, MATLAB sets the number of tapers to $2TW - 1$. We include as inputs to the function `pmtm` the data (`x`), the time-bandwidth product (`TW`), the amount of zero padding (here, set to none), and the sampling frequency (`Fs`). The function `pmtm` returns the multitaper estimate of the spectrum (`Sxx`) and the frequency axis (`f`). In figure 4.13 the spectrum of the ECoG data is computed in three ways, using the rectangular taper, the Hanning taper, and the multitaper method.

Q: Compare the spectra of the ECoG data in figure 4.13. How do the multitaper method results differ from the results computed with the other tapers? Do any new features appear in the spectrum using the multitaper method?

4.2.5 Confidence Intervals of the Spectrum

Another useful feature of the `pmtm` function is the ability to compute confidence intervals for the spectrum. The confidence interval for a multitaper spectral estimator with K tapers can be computed using a chi-square distribution with $2K$ degrees of freedom [8]. To return the confidence intervals, we recompute the spectrum with an additional returned output (`Serr`), as follows,

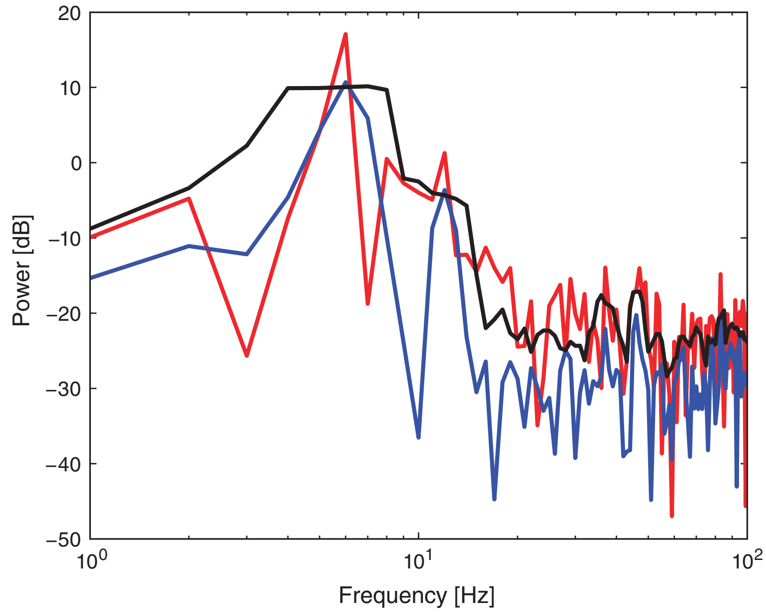

Figure 4.13
Spectrum of ECoG data computed using the rectangular taper (*red*), Hanning taper (*blue*), and the multitaper method (*black*).

```
%Compute MTM spectrum, and return CIs.
[Sxx,Serr,f]=pmtm(x,TW,length(x),Fs);

semilogx(f,10*log10(Sxx))        %Plot spectrum vs frequency,
hold on                          %... freeze the graphics window,
semilogx(f,10*log10(Serr))       %... plot the CI,
hold off                         %... release graphics window.
xlim([0 100])                    %Select frequency range,
xlabel('Frequency [Hz]')         %... and label the axes.
ylabel('Power [dB]')
```

The resulting spectra with confidence intervals are shown for different choices of time-bandwidth product in figure 4.14.

Q: Consider the multitaper spectra estimates of the ECoG data in figure 4.14. For each case determine the frequency resolution, and compare your answer to the plots in figure 4.14. How does changing the time-bandwidth product impact the plotted spectra?

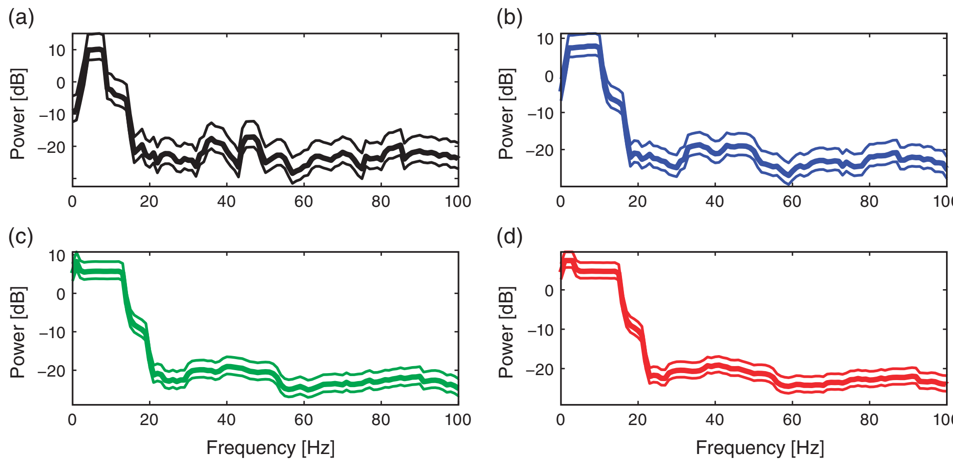

Figure 4.14

The multitaper spectral estimate of the ECoG data with different choices of the time-bandwidth product. The time-bandwidth products are (*a*) 3, (*b*) 5, (*c*) 8, (*d*) 10. The means are indicated with thick lines, and the 95% confidence intervals with thin lines.

Q: Use the spectral analysis results to draw some conclusions about the data. What rhythms do you think are present?

A: Our previous analysis using the Hanning taper suggested two peaks in the spectrum: a large peak near 6 Hz, and a second smaller peak near 10–15 Hz. Both peaks are still visible, at least partially, in the multitaper spectra estimates. We notice that as the time-bandwidth product increases, the frequency resolution becomes worse, and it becomes more difficult to resolve these two peaks; these neighboring peaks start to smear together. This is expected and the trade-off we accept in the multitaper method. The big payoff of the multitaper method occurs in the higher-frequency bands. As the time-bandwidth product increases, we observe a small elevation across a broad frequency range, at approximately 30–50 Hz. The spectral density in this interval is small and was hidden by the noise in the previous spectral estimates using the rectangular taper or the Hanning taper. However, by using many tapers in the multitaper method, we reduce this noise and reveal the broad elevation of power. Physiologically, the impact of this observation is enormous; the data exhibit a broad gamma band peak—one of the best studied and understood frequency bands in the brain—with implications for cognitive function and dysfunction [4].

Summary

In this chapter, we analyzed the rhythmic activity present in 1 s of ECoG data. We computed the spectrum and considered two issues: zero padding and tapering. We discussed that zero padding can increase the number of points along the frequency axis but cannot change the frequency resolution. We also explored the trade-off between three different tapers: the rectangular taper, the Hanning taper, and the multitaper method. The Hanning taper reduces the side lobes present in the rectangular taper, but fattens the spectral peaks. The multitaper method reduces the variance of the spectrum at the cost of worsened frequency resolution. Applying all three measures to the ECoG data allowed us to explore the rhythmic activity of these data in different ways. All three methods suggest rhythmic content at low frequencies, near 6–7 Hz, consistent with the visual inspection of the time series. Applying the Hanning taper, we uncovered activity in the 10–15 Hz range. Applying the multitaper method, we uncovered broadband activity at 30–50 Hz. This activity, hidden in the noisy spectrum, only became apparent upon increasing the number of tapers. However, this increase necessarily reduces the frequency resolution and can hide the low-frequency rhythms (compare figure 4.11d with figure 4.14d). In this case, we find it useful to examine the spectrum in a variety of ways.

Here we have only touched the surface of these concepts. Further discussions of zero padding, tapering, and the multitaper method may be found in [8, 9, 11].

Problems

4.1. Consider a 1 s sinusoid of 10 Hz activity (e.g., see figure 4.4). How does the spectrum differ when computed using the Hanning taper and rectangular taper? How does including 10 s of zero padding affect the results in each case?

4.2. Consider the file `Ch3-EEG-1.mat`, available at

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

These data served as the case study data in chapter 3; see that chapter for a detailed analysis of these data. Load these data into MATLAB, and answer the following questions.

a. Plot the spectrum versus the frequency using a rectangular taper. Do so without zero padding the data and then after adding 10 s of zero padding to the data.

b. Plot the spectrum versus the frequency using a Hanning taper. Do so without zero padding the data and then after adding 10 s of zero padding to the data.

c. Plot the spectrum versus the frequency using the multitaper method. Do so using time-bandwidth products of 2 and of 10. In addition, for each choice of time-bandwidth product, compute the spectrum without zero padding the data and then after adding 10 s of zero padding to the data.
d. In a few sentences interpret the spectra and describe the rhythms present in the signal. Compare the three spectra results. Do the analyses agree or disagree? What choice of taper and zero padding do you prefer for these data?

4.3. Consider the file `Ch3-EEG-2.mat`, available at

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

Load these data into MATLAB. If you have not done so, analyze these data as discussed in problem 3.4 of chapter 3. In addition, answer the following questions.

a. Plot the spectrum versus the frequency using a rectangular taper. Do so without zero padding the data and then after adding 10 s of zero padding to the data.
b. Plot the spectrum versus the frequency using a Hanning taper. Do so without zero-padding the data and then after adding 10 s of zero padding to the data.
c. Plot the spectrum versus the frequency using the multitaper method. Do so using time-bandwidth products of 2 and of 10. In addition, for each choice of time-bandwidth product, compute the spectrum without zero padding the data and then after adding 10 s of zero padding to the data.
d. Interpret the spectra and describe the rhythms present in the signal. Compare the three spectra results. Do the analyses agree or disagree? What choice of taper and zero padding do you prefer for these data?

4.4. Consider the file `Ch3-EEG-3.mat`, available at

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

Load these data into MATLAB. If you have not done so, analyze these data as discussed in problem 3.5 of chapter 3. Then answer the following questions.

a. Plot the spectrum versus the frequency using a rectangular taper. Do so without zero padding the data and then after adding 10 s of zero padding to the data.
b. Plot the spectrum versus the frequency using a Hanning taper. Do so without zero padding the data, and then after adding 10 s of zero padding to the data.
c. Plot the spectrum versus the frequency using the multitaper method. Do so using time-bandwidth products of 2 and of 10. In addition, for each choice of time-bandwidth product, compute the spectrum without zero padding the data and then after adding 10 s of zero padding to the data.

d. Interpret the spectra and describe the rhythms present in the signal. Compare the three spectra results. Do the analyses agree or disagree? What choice of taper and zero padding do you prefer for these data?

4.5. Consider the file `Ch3-EEG-4.mat`, available at

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

Load these data into MATLAB. If you have not done so, analyze these data as discussed in problem 3.6 of chapter 3. Also answer the following questions.

a. Plot the spectrum versus the frequency using a rectangular taper. Do so without zero padding the data and then after adding 10 s of zero padding to the data.

b. Plot the spectrum versus the frequency using a Hanning taper. Do so without zero padding the data and then after adding 10 s of zero padding to the data.

c. Plot the spectrum versus the frequency using the multitaper method. Do so using time-bandwidth products of 2 and of 10. In addition, for each choice of time-bandwidth product, compute the spectrum without zero padding the data and then after adding 10 s of zero padding to the data.

d. Interpret the spectra and describe the rhythms present in the signal. Compare the three spectra results. Do the analyses agree or disagree? What choice of taper and zero padding do you prefer for these data?

4.6. Consider the file `Ch3-EEG-5.mat`, available at

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

Load these data into MATLAB. If you have not done so, analyze these data as discussed in problem 3.7 of chapter 3. Also answer the following questions.

a. Plot the spectrum versus the frequency using a rectangular taper. Do so without zero padding the data and then after adding 10 s of zero padding to the data.

b. Plot the spectrum versus the frequency using a Hanning taper. Do so without zero padding the data and then after adding 10 s of zero padding to the data.

c. Plot the spectrum versus the frequency using the multitaper method. Do so using time-bandwidth products of 2 and of 10. In addition, for each choice of time-bandwidth product, compute the spectrum without zero padding the data and then after adding 10 s of zero padding to the data.

d. Interpret the spectra and describe the rhythms present in the signal. Compare the three spectra results. Do the analyses agree or disagree? What choice of taper and zero padding do you prefer for these data?

4.7. Simulate a signal consisting of the sum of two sinusoids oscillating at 10.5 Hz and 10.8 Hz. Set $T = 1$ s and compute the spectrum. Can you resolve the two different frequencies? Zero-pad the signal by adding 19 s of zeros. Now can you resolve the two peaks? How much data would you need to simulate (i.e., how big should T be) to resolve the two frequencies? Show this in simulation.

Appendix: Multiplication and Convolution in Different Domains

We stated in this chapter the important fact that multiplication in the time domain is equivalent to convolution in the frequency domain. Mathematically, we may express this relation as,

$$FT\left[x\,w\right] = FT\left[x\right] \star FT\left[w\right], \tag{4.3}$$

where x and w are two time series, $FT\left[x\right]$ is the Fourier transform of x, and $X \star Y$ indicates the convolution of X and Y,

$$X \star Y\left[\beta\right] = \int_{-\infty}^{\infty} X\left[b\right] Y\left[\beta - b\right] db.$$

The convolution of two functions (with arguments b in this formula) is itself a function of the same argument (with symbol β in this formula). Equation (4.3) states that the Fourier transform of the element-by-element product of x and w equals the convolution of the Fourier transform of x and the Fourier transform of w. We consider here an equivalent, alternative statement: that convolution in the time domain is equivalent to multiplication in the frequency domain. Mathematically,

$$FT\left[x \star w\right] = FT\left[x\right] FT\left[w\right]. \tag{4.4}$$

This equation states that the Fourier transform of the convolution of x and w equals the product of the Fourier transform of x and the Fourier transform of w. To prove this relation, let's consider the Fourier transform of the convolution of x and w. We use the expression for the continuous-time Fourier transform (3.27) from chapter 3,

$$FT\left(x \star w\left[\tau\right]\right) = \int_{-\infty}^{\infty} \left(x \star w\left[\tau\right]\right) e^{-2\pi i f \tau} d\tau,$$

where the notation $\left[\tau\right]$ indicates that the convolution ($x \star w$) is a function of time τ. Now, let's substitute the definition of convolution into this expression and simplify using an introduction of a second exponential expression,

$$\begin{aligned}
FT\big(x \star w[\tau]\big) &= \int_{-\infty}^{\infty} \Big(\int_{-\infty}^{\infty} x[t] w[\tau - t] dt \Big) e^{-2\pi i f \tau} d\tau \\
&= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x[t] w[\tau - t]\, dt\, e^{-2\pi i f(\tau - t)} e^{-2\pi i f t} d\tau \\
&= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \big(x[t] e^{-2\pi i f t}\big)\big(w[\tau - t] e^{-2\pi i f(\tau - t)}\big) dt\, d\tau.
\end{aligned}$$

Setting $T \equiv \tau - t$, we find

$$\begin{aligned}
F\big(x \star w[\tau]\big) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \big(x[t] e^{-2\pi i f t} dt\big)\big(w[T] e^{-2\pi i f(T)} dT\big) \\
&= \Big(\int_{-\infty}^{\infty} x[t] e^{-2\pi i f t} dt \Big)\Big(\int_{-\infty}^{\infty} w[T] e^{-2\pi i f(T)} dT \Big) \\
&= FT[x]\, FT[w]
\end{aligned}$$

and therefore conclude that the Fourier transform of the convolution of x and w equals the element-by-element product of their Fourier transforms.

In MATLAB we may compute a simple example to illustrate this relation:

```
x = [3 4 5 6];            %Define a simple signal x,
w = [-1 0.1 -0.2 1];      %... and another simple signal w.
a=fft(conv(w,x));         %Take the FT of the convolution,
b=fft([w 0 0 0]).*fft([x 0 0 0]);%... and the product of the FTs.
```

In the first two lines, we define two simple signals; each consists of only four elements, which is enough to illustrate the relation. In the third line, we first compute the convolution of `w` and `x`, and then the Fourier transform. In the last line, we compute the Fourier transform of each variable, and then their element-by-element product. Notice that we zero-pad both variables before computing the Fourier transform in the last line. We do so to avoid computing circular correlations between the variables (i.e., wrapping around one variable when comparing it to another). Also, we make the lengths of variables `a` and `b` the same. Evaluating the statement, we find `a` equals `b`; to see this, print out both variables at the MATLAB command line.