CHAPTER 6

EXERCISES ON THE POWER SPECTRUM

In this homework you will determine power spectra from models and data. To do this, you will need to use the Fast Fourier Transform, which is performed by the R function `fft()`. The output of `fft` is somewhat technical, so I give you the following program to convert the output into a periodogram:

```
periodogram = function(x) {
## COMPUTES THE PERIODOGRAM OF A VECTOR X USING THE FFT
## ASSUMES THE LENGTH OF X IS EVEN AND HIGHLY COMPOSITE
##   (E.G., EXPRESSIBLE AS POWERS OF 2,3,5)
# INPUT:
#    X[NTOT]: DATA VECTOR
# OUTPUT:
#    PGRAM[NTOT/2]: VECTOR OF PERIODOGRAM AMPLITUDES AT HARMONIC FREQUENCIES
#      1/N, 2/N, . . ., 1/2 CYCLES PER TIME STEP (A TOTAL NTOT/2 FREQUENCIES)
# COMMENT: (BIASED) VARIANCE = 2 * SUM(PGRAM[1:(NTOT/2-1)]) + PGRAM[NTOT/2]

ntot  = length(x)
if ( ntot %% 2 != 0 ) stop('length of x must be even')
x.fft = fft(x)
pgram = Re(x.fft * Conj(x.fft))[1:(ntot/2)+1] / ntot
pgram
}
```

You should verify that this function works by setting `x` to be specific sum of sines and cosines and ensuring that the output gives the appropriate amplitudes, when appropriately scaled. The following example shows a successful test of this function:

```
> set.seed(4)
> ntot = 32
> t    = 1:ntot - 1
> x    = 1 * cos(2*pi*t*5/ntot) - 3 * sin(2*pi*t*13/ntot)
      + 8 * cos(2*pi*t*(ntot/2)/ntot)
> x.pgram = periodogram(x)
> x.amplt = sqrt(4*x.pgram/ntot)   # convert pgram to fourier amplitude
# special conversion for last harmonic
> x.amplt[ntot/2] = sqrt(x.pgram[ntot/2]/ntot)
> x.amplt
 [1] 2.519803e-15 2.181560e-15 7.616681e-16 1.930989e-15 1.000000e+00
 [6] 1.260945e-15 7.422817e-16 3.202373e-15 2.989367e-15 2.373832e-15
[11] 3.143748e-15 1.223717e-15 3.000000e+00 3.084970e-16 1.271623e-15
[16] 8.000000e+00
```

The output shows that the absolute amplitude of the 5th, 13th, and 16th harmonics are 1, 3, 8, as they should, while the other amplitudes are effectively zero, as expected. Another check is to verify the formula

$$\sum_{t=1}^{N} (x_t - \overline{x})^2 = 2 * \sum_{j=1}^{\frac{N}{2}-1} \hat{P}(j/N) + \hat{P}(N/2), \tag{6.1}$$

which holds for even N. The following lines perform this check:

```
> set.seed(1)
> x       = rnorm(ntot)
> ssx     = sum((x-mean(x))^2)
>
> x.pgram = periodogram(x)
> ssx.fft = 2*sum(x.pgram[1:(ntot/2-1)]) + x.pgram[ntot/2]
> print(all.equal(ssx,ssx.fft))
[1] TRUE
```

Exercise 6.1. Write an R function to compute the spectral average

$$\hat{p}_M(\omega_j) = \frac{1}{2M+1} \sum_{m=-M}^{M} \hat{p}(\omega_j). \tag{6.2}$$

The preamble of this R function should be as follows:

```
spectrum.smooth = function(pgram,window,alpha=0.1) {
## COMPUTES SPECTRAL RUNNING MEAN AS:
##   SPEC_AVE = SMOOTHED PERIODOGRAM FROM M = -WINDOW TO M = WINDOW
## INPUT:
##    PGRAM[NTOT/2]: PERIODOGRAM VALUES FROM PERIODOGRAM.R
##    WINDOW: HALF-WIDTH OF AVERAGING WINDOW
##    ALPHA: THE (1-ALPHA)100% LEVEL FOR THE CONFIDENCE LIMITS
## OUTPUT: LIST
##    PGRAM.AVE[NTOT/2]: AVERAGED PERIODOGRAM
##    FREQ[NTOT/2]: FREQUENCIES (CYCLES PER TIME STEP)
##    BW[2]: BANDWIDTH C(-WINDOW/NTOT,WINDOW/NTOT)
##    CI.LIMITS[2]: PROPORTIONALITY CONSTANT FOR LOWER AND UPPER CONFIDENCE LIMITS
##       (DOF/CHI-SQUARE[ALPHA/2, DOF], DOF/CHI-SQUARE[1-ALPHA/2,DOF])
##    PGRAM[NTOT/2]: REPEATS PERIODOGRAM (FOR LATER PLOTTING)
```

The spectral average near the extreme low and high frequencies, where the smoothing cannot be performed consistently, should be set to `NA`. □

Exercise 6.2. Generate a time series of length 64 from a standardized Gaussian white noise process, perform a spectral average using $M = 5$, and write out the results, as follows:

```
> set.seed(1)
> x.sd        = 1
> window      = 5
> ntot        = 64
> freq        = 1:(ntot/2) / ntot
> x           = rnorm(ntot,sd=x.sd)
> x.pgram     = periodogram(x)
> x.pgram.smooth = spectrum.smooth(x.pgram,window)
> cbind(freq,x.pgram,x.pgram.smooth$pgram.ave)
```

Plot the results using the R function `spectrum.plot.R`, provided on the class website. Based on this result, would you say that the estimated power spectrum is consistent with a white noise process? Explain why or why not. □

Exercise 6.3. What is the theoretical (i.e., population) power spectrum for the Gaussian white noise process? Superimpose this theoretical power spectrum on your spectrum plot. Is the estimated power spectrum consistent with the theoretical spectrum? □

Exercise 6.4. Generate a time series of length 64 from an AR(1) process with parameter $\phi_1 = 0.8$, as follows:

```
ntot        = 64
phi1        = 0.8
x           = ar1.ts(phi1,ntot,iseed=1)
```

Perform a spectral average of this time series using $M = 5$, and write out the results, as follows:

```
> window         = 5
> x.pgram        = periodogram(x)
> x.pgram.smooth = spectrum.smooth(x.pgram,window)
> cbind(freq,x.pgram,x.pgram.smooth$pgram.ave)
```

Make a plot of the power spectrum, smoothed periodogram, bandwidth, and 90% confidence interval using `spectrum.plot.R`. Is the estimated spectrum more consistent with white noise or an AR(1) process? Explain why or why not. □

Exercise 6.5. What is the theoretical power spectrum of the above AR(1) process? Superimpose this theoretical spectrum on your spectrum plot. Is your estimated power spectrum consistent with the theoretical spectrum? □

Exercise 6.6. Estimate the power spectrum of the NINO3.4 index. The NINO3.4 index can be extracted using the following R program:

```
fname       = paste(dir.enso,'nina34.data',sep='')
line1       = scan(fname,nlines=1,quiet=TRUE)
nino34.yr   = line1[1]:line1[2]
nino34.nyrs = length(nino34.yr)
cnames      = c('YEAR',month.abb[])
data2       = read.table(fname,skip=1,nrows=nino34.nyrs,
   na.strings=-99.99,col.names=cnames)

## CREATE VECTOR WITH NINO34 TIME SERIES AND TIME AXIS
nino34.ts   = as.numeric(t(as.matrix(data2[,1:12+1])))
nino34.time = seq(from=data2[1,1],length.out=length(nino34.ts),by=1/12)
```

The resulting time series will have `NA`'s which need to be stripped out before performing spectral analysis. Trim the length of the index to be highly composite (i.e., a product of powers of 2, 3, 5). State the final length that you use. What are the units of "frequency"? Are there any significant peaks? Do these peaks make sense (do you expect certain peaks)? Could an AR-process fit this spectrum? ENSO is often said to be characterized by oscillations around 4-7 years. Does your estimated power spectrum support this characterization? Explore different bandwidths, choose a single one that you believe is most revealing, and show that one. □