The Periodogram

- Recall: the discrete Fourier transform

\[ d(\omega_j) = n^{-1/2} \sum_{t=1}^{n} x_t e^{-2\pi i \omega_j t}, \quad j = 0, 1, \ldots, n - 1, \]

- and the periodogram

\[ I(\omega_j) = |d(\omega_j)|^2, \quad j = 0, 1, \ldots, n - 1, \]

- where \( \omega_j \) is one of the Fourier frequencies

\[ \omega_j = \frac{j}{n}. \]
Sine and Cosine Transforms

• For $j = 0, 1, \ldots, n - 1$,

$$d(\omega_j) = n^{-1/2} \sum_{t=1}^{n} x_t e^{-2\pi i \omega_j t}$$

$$= n^{-1/2} \sum_{t=1}^{n} x_t \cos(2\pi \omega_j t) - i \times n^{-1/2} \sum_{t=1}^{n} x_t \sin(2\pi \omega_j t)$$

$$= d_c(\omega_j) - i \times d_s(\omega_j).$$

• $d_c(\omega_j)$ and $d_s(\omega_j)$ are the cosine transform and sine transform, respectively, of $x_1, x_2, \ldots, x_n$.

• The periodogram is $I(\omega_j) = d_c(\omega_j)^2 + d_s(\omega_j)^2$. 
Sampling Distributions

- For convenience, suppose that $n$ is odd: $n = 2m + 1$.

- White noise: orthogonality properties of sines and cosines mean that $d_c(\omega_1), d_s(\omega_1), d_c(\omega_2), d_s(\omega_2), \ldots, d_c(\omega_m), d_s(\omega_m)$ have zero mean, variance $\frac{1}{2}\sigma_w^2$, and are uncorrelated.

- Gaussian white noise: $d_c(\omega_1), d_s(\omega_1), d_c(\omega_2), d_s(\omega_2), \ldots, d_c(\omega_m), d_s(\omega_m)$ are i.i.d. $N\left(0, \frac{1}{2}\sigma_w^2\right)$.

- So for Gaussian white noise, $I(\omega_j) \sim \frac{1}{2}\sigma_w^2 \times \chi_2^2$. 
• General case: \( d_c(\omega_1), d_s(\omega_1), d_c(\omega_2), d_s(\omega_2), \ldots, d_c(\omega_m), d_s(\omega_m) \) have zero mean and are \textit{approximately} uncorrelated, and

\[
\text{var}[d_c(\omega_j)] \approx \text{var}[d_s(\omega_j)] \approx \frac{1}{2} f_x(\omega_j),
\]

where \( f_x(\omega_j) \) is the spectral density function.

• If \( x_t \) is Gaussian,

\[
\frac{I_x(\omega_j)}{\frac{1}{2} f_x(\omega_j)} = \frac{d_c(\omega_j)^2 + d_s(\omega_j)^2}{\frac{1}{2} f_x(\omega_j)} \sim \text{approximately } \chi^2_2,
\]

and \( I_x(\omega_1), I_x(\omega_2), \ldots, I_x(\omega_m) \) are approximately independent.
Spectral ANOVA

- For odd $n = 2m + 1$, the inverse transform can be written

$$x_t - \bar{x} = \frac{2}{\sqrt{n}} \sum_{j=1}^{m} \left[ d_c(\omega_j) \cos(2\pi \omega_j t) + d_s(\omega_j) \sin(2\pi \omega_j t) \right].$$

- Square and sum over $t$; orthogonality of sines and cosines implies that

$$\sum_{t=1}^{n} (x_t - \bar{x})^2 = 2 \sum_{j=1}^{m} \left[ d_c(\omega_j)^2 + d_s(\omega_j)^2 \right]$$

$$= 2 \sum_{j=1}^{m} I(\omega_j).$$
ANOV A table:

<table>
<thead>
<tr>
<th>Source</th>
<th>df</th>
<th>SS</th>
<th>MS</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_1$</td>
<td>2</td>
<td>$2I(\omega_1)$</td>
<td>$I(\omega_1)$</td>
</tr>
<tr>
<td>$\omega_2$</td>
<td>2</td>
<td>$2I(\omega_2)$</td>
<td>$I(\omega_2)$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>$\omega_m$</td>
<td>2</td>
<td>$2I(\omega_m)$</td>
<td>$I(\omega_m)$</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>2$m = n - 1$</td>
<td>$\sum(x_t - \bar{x})^2$</td>
<td></td>
</tr>
</tbody>
</table>
Hypothesis Testing

Consider the model

\[ x_t = A \cos(2\pi \omega_j t + \phi) + w_t. \]

Hypotheses:

- \( H_0 : A = 0 \Rightarrow x_t = w_t \), white noise;
- \( H_1 : A > 0 \), white noise plus a sine wave.

Note: no autocorrelation in either case.
Two cases:

- \( \omega_j \) known: use

\[
F_j = \frac{I(\omega_j)}{(m-1)^{-1} \sum_{j' \neq j} I(\omega_{j'})}
\]

which is \( F_{2,2(m-1)} \) under \( H_0 \).

- \( \omega_j \) unknown: use \( \max(F_1, F_2, \ldots, F_m) \), or equivalently

\[
\kappa = \max \left\{ \frac{I(\omega_j)}{m^{-1} \sum_j I(\omega_j)}, j = 1, 2, \ldots, n \right\}
\]

and

\[
P(\kappa > \xi) \approx 1 - \exp \left\{ -m \exp \left[ -\xi \left( \frac{m - 1 - \log m}{m - \xi} \right) \right] \right\}.
\]
Example: the Southern Oscillation Index

- Using SAS: proc spectra program and output.

- Using R:

```r
par(mfcol = c(2, 1))
# Use fft() to calculate the periodogram directly; note that
# frequencies are expressed in cycles per year, and the
# periodogram values are similarly scaled by 12:
freq = 12 * (0:(length(soi) - 1)) / length(soi)
plotit = (freq > 0) & (freq <= 6)
soifft = fft(soi) / sqrt(length(soi))
plot(freq[plotit], Mod(soifft[plotit])^2 / 12, type = "l")
# Use spectrum(); override some defaults to make it match:
spectrum(soi, log = "no", fast = FALSE, taper = 0, detrend = FALSE)
```