

第三 章周波数領域における単変量解析，スペクトル 1

3.1. スペクトルの概説 1

3.1.1. はじめに 1

3.1.2. フーリエ変換とスペクトル 2

3.1.3. ピリオドグラム法 3

3.1.4. パーシバルのエネルギー保存定理 4

3.1.5. ウィーナー・ヒンチンの関係 5

3.2. 直接法 6

3.2.1. 平滑化を行わない直接法スペクトルの誤差の概要 6

3.2.2. テーパー 7

3.2.3. ゼロパッチング 10

3.2.4. Matlab でのスペクトル計算 12

3.3. 直接スペクトル推定法の誤差理論 13

3.3.1. 推定誤差理論の基礎 13

3.3.2. 等価自由度への平滑化方法の影響 14

3.3.3. 等価自由度へのテーパーとゼロパッチングの影響 14

3.3.4. スペクトルの計算定順（周波数領域での平滑化） 15

3.3.5. スペクトルの推定誤差のテスト 15

3.4. MEM 18

3.4.1. MEM の概説 18

3.4.2. MEM スペクトルの推定方法 19

3.4.3. Matlab での MEM の計算 19

3.5. スペクトルの有意性判定の注意 22

参考文献 23

第三 章周波数領域における単変量解析，スペクトル

3.1. スペクトルの概説

3.1.1. はじめに

気候における周期的な変動は，もし存在するなら，なにかその周期を説明する複雑な過程があることを意味し，さらに予測可能性を含意するので，大きな興味の対象である．そこで，周期的な変動が存在するか否かは，非常に大きな問題である．これは，スペクトルのピークが存在することを示すことに他ならない．

スペクトルとは，データ関数 $y(t)$ をフーリエ展開して得られる，振幅の二乗であり，周波数の関数として表現される．主要なスペクトルの計算方法は，自己相関係数をフーリエ変換する **ブラックマン・チューキー(Blackman-Tukey)法**，データ関数 $y(t)$ をフーリエ展開してその振幅の二乗を求める **直接法** または **平滑化ピリオドグラム(periodgram)法**，統計的なモデルである自己回帰モデルをデータ関数から推定してそのモデル係数からスペクトルを推定する **最大エントロピー法(Maximum Entropy Method: MEM)** である．このうち，最初の二つは **ノンパラメトリック推定**，最

後の一つをパラメトリック推定とも言う。パラメトリック推定とは、自己回帰モデルのモデルパラメータをまず推定することから来ている。

歴史的には Blackman-Tukey 法がまず使われるようになり、次に FFT の発展に伴って直接法が使われるようになった。MEM は Blackman-Tukey 法とほぼ同時期に提案されたが、一般に広がるのは遅かった。今日最もよく利用されるのは直接法で、ついで MEM が用いられる。MEM は、比較的短いデータから周期成分を推定する場合によく用いられるが、標準的な誤差推定方法がごく最近まで提案されておらず（私の知る限りでは、大気海洋分野ではまだ MEM の誤差推定理論を応用した論文は無い）、MEM で得られるピークが有意かどうかは通常あいまいなまま議論されてきた。また、ブラックマン・チューキー法はスペクトル推定の実用に用いられるというよりも、後に示すウィナー・ヒンチンの関係を示す、教育的な意味では今でも重要である。この資料では、直接法と最大エントロピー法とを紹介しよう。

より最近の手法としては、Singular Spectrum Analysis あるいは Multi-Taper Method (Matlab では `pmtm`) などがあり、特に比較的短い時系列に対して有効である（つまり長周期変動を取り出せる）、として宣伝されている。これらの手法を、私も使う場合がある。しかし、これらの手法はまだ登場してから新しいために、その欠点が十分に明らかになっているとは言えないので、利用に際しては注意が必要である。一般に、強力な手法であればあるほど、落とし穴も深い。

3.1.2. フーリエ変換とスペクトル

離散的にサンプリングされた時系列 $y(t)$ の指数型フーリエ変換 (Fourier transform) は次式で与えられる。

$$\begin{aligned} Y(k) &= \sum_{t=1}^N y(t) \exp(-i2\pi kt/N) \\ &= \sum_{t=1}^N y(t) \exp(-i2\pi f_k t \Delta t); \quad f_k = k/(N\Delta t), \quad k = 0, \dots, N \end{aligned} \quad (3.1)$$

ここで、 Y は本来は周波数の関数であるが、周波数が $1/N\Delta t, 2/N\Delta t, \dots$ の様にサンプリングされているのでその k 番目という意味で、 k の関数としている。解像可能な最大の周波数は、 $f_N = 1/(2\Delta t)$ で与えられ、これをナイキスト周波数 (Nyquist frequency) と呼ぶ。例えば、時系列のサンプリング間隔が 1 なら、ナイキスト周波数は 0.5 である。(3.1) の表現ではナイキスト周波数よりも高い周波数が使われているかのようであるが、次の関係から

$$\begin{aligned} \exp(-i2\pi(N-k)/N) &= \cos(2\pi(1-k/N)) - i \sin(2\pi(1-k/N)) \\ &= \cos(2\pi) \cos(-2\pi k/N) - i \cos(2\pi) \sin(-2\pi k/N) \\ &= \cos(2\pi k/N) + i \sin(2\pi k/N) \\ &= \exp(+i2\pi k/N) \end{aligned}$$

一見ナイキスト周波数よりも高い周波数と見えた成分は、負の周波数を意味している。このために、周波数範囲を $1/(N\Delta t) < f_k < 1/\Delta t$ ではなく、 $-f_N < f_k < f_N$ で示す流儀もよく使われる。また、 $f = 1/(N\Delta t)$ は 1 周期がデータ長である周波数で、これを基本周波数 (fundamental frequency) ということがある。

(3.1) に対応する指数型逆フーリエ変換 (inverse Fourier transform) は、

$$y(t) = \frac{1}{N} \sum_{k=0}^{N-1} Y(k) \exp\left(\frac{i2\pi kn}{N}\right), \quad t = 1, \dots, N, \quad \Delta f = \frac{1}{N\Delta t}$$

で定義される。なお(1/N)をフーリエ変換と逆フーリエ変換でどう配分するかは好みによるが、上の表現は Matlab 関数の **fft** と整合的である。

なお、時系列は時間を独立変数として持つ関数で表される。一方フーリエ係数は(そして次節で紹介するスペクトルも)周波数が独立変数となる。時間を独立変数に持つ空間を**時間領域 (time domain)**、周波数を独立変数に持つ空間を**周波数領域 (frequency domain)**と呼ぶ。

3.1.3. ピリオドグラム法

ピリオドグラム (periodogram)法とは、フーリエ振幅の単に2乗でスペクトルを表現するものである。この方法では、**パワー・スペクトル密度(Power Spectrum Density, PSD)**, S_{yy} は次式で与えられる¹。

$$S_{yy}(f_k) = \frac{1}{N} |Y(k)|^2, \tag{3.2}$$

ここで、 f_k, Y_k は、(3.1)で定義された周波数と、その周波数のフーリエ展開係数である。

また、フーリエ係数を $-f_N < f_k < f_N$ の範囲で表現する流儀を採用した場合、

$$S_{yy}(f_k) = \frac{1}{N} |Y(k)|^2, \quad -f_N < f_k \leq f_N \tag{3.3}$$

と表す場合もある。このスペクトルの表現を**両側スペクトル密度(two-sided Power Spectrum Density)**と呼ぶ。

スペクトルを求める時系列は、通常実数関数である。この場合、フーリエ係数は $Y(-f) = Y(f)^*$ であり、スペクトルは $S_{yy}(f_k) = S_{yy}(-f_k)$ と、正負の周波数について同じ値を持つ。したがって、

正の周波数のみを考慮した、**片側スペクトル密度(one-sided Power Spectrum Density)**

$$k = \begin{cases} 1, \dots, \frac{N}{2}-1, & G_{yy}(f_k) = \frac{2}{N} |Y_k|^2, \\ 0, & G_{yy}(0) = \frac{1}{N} |Y_0|^2, \\ \frac{N}{2}, & G_{yy}(f_{N/2}) = G_{yy}(f_{N/2}) = \frac{1}{N} |Y_{N/2}|^2, \end{cases} \tag{3.4}$$

を示すことが多い。ここで、 $1 \leq k \leq N/2 - 1$ で両側スペクトル密度の値を2倍しているのは、パーシバルの公式を、

$$\sum_{n=1}^N |y_n|^2 = \sum_{n=0}^{N/2} G_{yy} \tag{3.5}$$

というシンプルな形に書きたいためである。しかし、実関数のスペクトルを表現するのに、単純に、両側スペクトル密度の片側だけ

¹ いきなり S_{yy} というように y の添え字が二つもついているが、これは将来クロススペクトルを S_{xy} のように表して、クロススペクトルもただのスペクトルも統一的に表記する狙いである。

$$S_{yy}(f_k) = \frac{1}{N} |Y(f_k)|^2, \quad 0 \leq f_k \leq f_N \tag{3.6}$$

を図示する場合も多い．また， N で割らない

$$S_{yy}(f_k) = |Y(f_k)|^2, \quad 0 \leq f_k \leq f_N \tag{3.7}$$

という表現もしばしば用いられる．この場合，分散1の白色雑音に対するスペクトル密度が1というシンプルな値を持つので，この表現は理論的な扱いで好まれる．また，ゼロ・パディングによって，スペクトルのレベルが変化しないという利点もある．

3.1.4. パーシバルのエネルギー保存定理

有名な，**パーシバルのエネルギー保存定理(Parseval's energy conservation theorem)**，

$$\sum_{n=1}^N |y_n|^2 = \sum_{n=1}^N S_{yy} \tag{3.8}$$

は，左辺の**時間領域で計測されるエネルギー**が，右辺の**周波数領域の**（つまりフーリエ変換された）**エネルギーで保存することを保証**している．

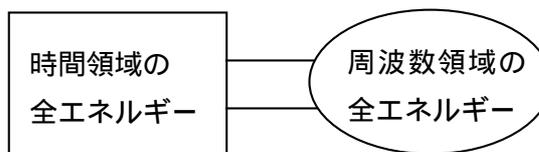


図 3.1 パーシバルのエネルギー保存定理の概念図．時間領域の全エネルギーが，周波数領域の全エネルギーに等しい．

さらに，周波数領域でのエネルギーが，任意の周波数区間において正の値を持つことは重要である．このことと，パーシバルのエネルギーの保存定理から，（時間領域・周波数領域に共通な）全エネルギーを，任意の区間が表現するエネルギーの加算値で考えることができる．例えば，100年の長さの時系列があるとして，その10-20年のエネルギーが50%以上を説明したとしよう．この場合，他の周波数帯全てのエネルギーを足しても，必ず50%以下になる．したがって，スペクトルに限らず，**加算される値が正に限定されるということは，その成分が全体に占める重要性が一意に定まることを意味する．**

パーシバルの定理の証明：

時系列 $y_n, n = 0, N - 1$ を離散フーリエ逆変換で表すと，

$$y(t) = \frac{1}{N} \sum_{k=0}^{N-1} \left[Y(k) \exp \frac{i2\pi kt}{N} \right]$$

左辺の2乗和 $\sum_{n=0}^{N-1} y(t)^2$ を求めよう。すると、

$$\sum_{t=0}^{N-1} y(t)^2 = \left(\frac{1}{N}\right)^2 \sum_{n=0}^{N-1} \left\{ \sum_{k=1}^{N-1} \left[Y(k) \exp \frac{i2\pi kn}{N} \right] \right\} \times \left\{ \sum_{k=1}^{N-1} \left[Y(k) \exp \frac{i2\pi kn}{N} \right] \right\}^*$$

ここで*は共役複素数を表している。直交関数系同士の掛け算を積分(ここでは n についての総和で表されている)すると、常にそうであるように、互いに同じ関数同士の掛け算のみが生き残るので、 $\Delta f = 1/(N\Delta t)$ を用いて

$$\sum_{n=0}^{N-1} y(t)^2 = \left(\frac{1}{N}\right)^2 \sum_{t=0}^{N-1} \left\{ \sum_{k=0}^{N-1} |Y(k)|^2 \right\} = \left(\frac{1}{N}\right)^2 N \sum_{k=0}^{N-1} |Y(k)|^2$$

から、

$$\sum_{t=0}^{N-1} y(t)^2 = \frac{1}{N} \sum_{k=0}^{N-1} |Y(k)|^2 = \sum_{k=0}^{N-1} S_k$$

となって、(3.8)が得られた。

3.1.5. ウィーナー・ヒンチンの関係

ウィーナー・ヒンチンの関係(Wiener-Khinchin relation)は、「自己共分散関数のフーリエ変換はスペクトルであり、スペクトルの逆フーリエ変換は自己共分散関数である。」である。この関係から、自己共分散関数、あるいはその規格化された関数である自己相関関数で示される、情報は全てスペクトルの中に存在していると言える。

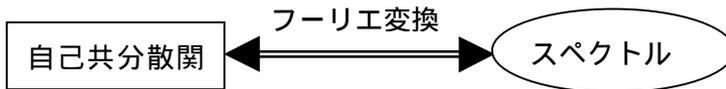


図 3.2 ウィーナー・ヒンチンの関係の概念図

ウィーナー・ヒンチンの関係の証明：

自己共分散関数 $\rho(\tau) = (1/N) \sum_{t=0}^{N-1} x(t)x(t+\tau)$ のフーリエ変換は、

$$\begin{aligned} \sum_{\tau=0}^{N-1} \left[\rho(\tau) e^{(-i2\pi k\tau/N)} \right] &= \sum_{\tau=0}^{N-1} \left[e^{(-i2\pi k\tau/N)} \frac{1}{N} \sum_{t=0}^{N-1} x(t)x(t+\tau) \right] \\ &= \frac{1}{N} \sum_{t=0}^{N-1} x_t \left\{ \sum_{\tau=0}^{N-1} x_{t+\tau} e^{-i2\pi k(t+\tau)/N} \right\} e^{-i2\pi k(-t)/N} \\ &= \frac{1}{N} \sum_{t=0}^{N-1} x_t e^{-i2\pi(-k)t/N} \{ X(k) \} \\ &= \frac{1}{N} X(-k)X(k) = \frac{1}{N} |X(k)|^2 \end{aligned}$$

であるので、自己共分散関数のフーリエ変換が、スペクトルに他ならないことが証明された。したがって、スペクトルのフーリエ逆変換は、自己共分散関数である。

3.2. 直接法

3.2.1. 平滑化を行わない直接法スペクトルの誤差の概要

直接法によって得られるスペクトルは、時系列の線形変換であるフーリエ変換によって得られた実部・虚部を2乗して和をとる。この実部・虚部は各々独立である。したがって、直接計算されたスペクトルは、フーリエ振幅の実部と虚部の二乗和なので、自由度が2のカイ二乗分布に従う。1章で述べたように、自由度 N のカイ二乗分布の平均は N 、分散は $2N$ であるから、自由度2のカイ二乗分布の平均は2で標準偏差も2（分散は4）である。つまり、標準偏差で誤差の大きさを代表すれば、直接フーリエ変換による推定スペクトルのシグナル（平均）とノイズ（標準偏差）が同じ大きさを持つ。注意すべきことは、**時系列の長さを長くしても相対誤差は改善されない**ことである。時系列の長さを長くすると、それに比例して、高い周波数分解能でスペクトルが推定される。しかし、個々の周波数成分におけるスペクトルの相対誤差は変化しない。

そこで、何らかの平滑化を行って、個々の周波数成分でのスペクトルの推定値の誤差を減少させるという方法が一般に採用される。なお、スペクトルの変化幅（ダイナミックレンジ）が非常に大きいのであれば、平滑化を行わなくてもピークを検出できる場合もあるので、平滑化は必須ではない。現在用いられている主な平滑化の方法は、

- 1) 周波数領域での平滑化。
- 2) 時間領域で分割して、複数のスペクトルを求め、それを平均する。

の2方法である。長周期成分に興味がある場合は、前者が有利であろう。後者は、時間領域で分割する場合に、互いにある程度重なり合うようする Welch のオーバーラッピング法(Welch が、良い性能を持つことが知られている)。

いずれの方法も、スペクトルの推定誤差を減少させるには、スペクトルの周波数分解能が犠牲になる。例えば、周波数領域で平滑化する場合を考えよう。 N 点のデータから得られる周波数分解能は、 $2/N$ である。これを、 m 点移動平均を行って平滑化すれば、周波数分解能は $2m/N$ に低下する。一方、平滑化されたスペクトルは自由度 $2m$ のカイ二乗分布に従い、スペクトルの推定誤差は平滑化前よりも著しく小さくなる。個々の応用では、スペクトル推定の目的と真のスペクトル分布とが異なるので、最善の平滑化を先験的に決めることはできない。なお、上の(1)(2)の方法の各々に平滑化ウィンドウとデータ・ウィンドウとが用いられる。この二つは紛らわしいので、簡単に説明しておこう。

- 1) **平滑化ウィンドウ(smoothing window)**：周波数領域で平滑化は、重み付き移動平均で行なわれる。この重み関数が平滑化ウィンドウである。
- 2) **データ・ウィンドウ(data window)**：時間領域で分割する場合に、分割された時系列各々に次節で説明するデータ・ウィンドウをかけることが多い。データ・ウィンドウのことをまた、**時間窓(time window)** あるいは**データ・テーパー(taper)**とも言う。データ・ウィンドウのフーリエ変換が、**スペクトル・ウィンドウ(spectral window)**である。

3.2.2. テーパー

スペクトルの推定は、有限のデータに対して行うために、データの端の影響が推定されたスペクトルに生ずる。有限のデータを全て利用すると、この端の影響のために、スペクトルの主たる峰が存在する周波数から、他の周波数に**スペクトルの漏れ(spectral leakage)**が生ずることが知られている。このスペクトルの漏れは、**テーパー(taper)**を用いることで、軽減することができる。ただし、テーパーを用いると、スペクトルの主たる峰の幅が広がる。つまり、**テーパーを利用することは、スペクトル推定の周波数解像度を減少させるという犠牲の下に、スペクトルの漏れを軽減する**という効果を上げることができる。

テーパーをかけた場合のスペクトルはどのように得られるかを検討しよう。テーパー関数を $h(t)$ とする。すると時系列 $y(t)$ にテーパーを適用し、そのフーリエ変換を求めると、

$$Y'(k) = \Delta t \sum_{t=1}^N h(t) y(t) \exp(-i2\pi kt\Delta f), k = 1, \dots, N, \Delta f = 1/(N\Delta t)$$

である。また、テーパーを用いないフーリエ変換が $Y(k)$ であるとする、その逆変換が、

$$y(t) = \Delta f \sum_{l=1}^N Y(l) \exp(i2\pi lt\Delta f)$$

である。これを上式に代入して、

$$Y'(k) = \Delta t \sum_{t=1}^N h(t) \Delta f \sum_{l=1}^N Y(l) \exp(i2\pi lt\Delta f) \exp(-i2\pi kt\Delta f)$$

和の順序を変えて、

$$\begin{aligned} Y'(k) &= \Delta f \sum_{l=1}^N Y(l) \Delta t \sum_{t=1}^N h(t) \exp(-i2\pi (k-l)t\Delta f) \\ &= \Delta f \sum_{l=1}^N Y(l) H(k-l) \end{aligned} \tag{3.9}$$

ただしここで、 H は $h(t)$ のフーリエ変換

$$H(k) = \Delta t \sum_{t=1}^N h(t) \exp(-i2\pi kt\Delta f),$$

である。つまり、(3.9)は、テーパーを用いたフーリエ変換は、もともとの時系列のフーリエ変換と、テーパーのフーリエ変換との畳み込みで表されることを示している。また、テーパーを用いて推定されるスペクトルは、 $S'(k) = |Y'(k)|^2$ で与えられる。実はテーパーを用いた場合の、フーリエ変換(3.9)ではなく、スペクトル $S'(k)$ がどのように表されるかは簡単ではない。川嶋・酒井 (199X, p. 286) および Percival and Walden (1993, p. 207) はスペクトルの期待値が、

$$\{S'\} = \int_{-f_N}^{f_N} |H(f-f')|^2 S(f') df' \tag{3.10}$$

で与えられることを述べている。

よく利用されるテーパーには、**ハミングテーパー(Hamming taper)**

$$w(n\Delta t) = \frac{1}{2} [1.0 - \cos(2\pi(n-0.5)/N)], \quad n = 1, 2, \dots, N$$

やハニング・テーパー(Hanning taper)

$$w(n\Delta t) = 0.54 - 0.46 \cos(2\pi(n-0.5)/N), \quad n = 1, 2, \dots, N$$

がある Hamming taper は Hanning taper の改良版で、よりスペクトルの漏れが小さく優れている。また、最初と最後の 10% つづを cos とする、cos20 テーパーもしばしば用いられる。cos20 テーパーは、hamming, hanning taper よりも、スペクトルの漏れを減少させる効果は小さいが、データの最初と最後の部分をより生かすテーパーである。

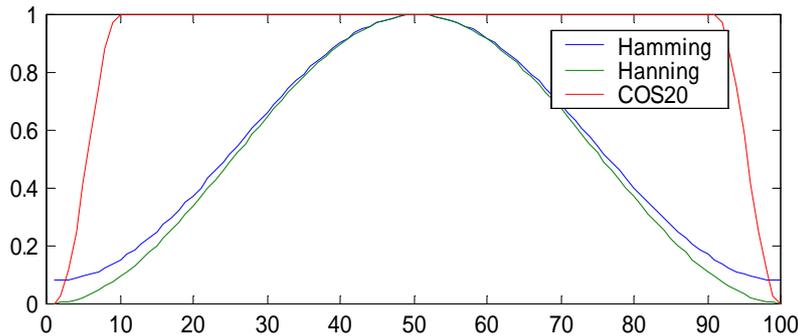


図 3.3 Hanning(青線), Hamming (緑線), cos20(赤線)の各テーパー。

テーパーの効果を、次の

$$y(t) = \cos(2\pi t / 21) + 0.2 \cos(2\pi t / 9)$$

という時間ステップ数が 100 点の時系列について検討しよう。この時系列を、テーパー無し (boxcar taper), cos20, hamming の各テーパーをかけて行ったスペクトルの推定値と、その 95% 信頼限界を図 2 . X に示す。この場合は、boxcar テーパーと cos20 テーパーでは、周期 9 のピークを有意なピークとして検出できなかったが、hamming taper を用いた場合には、明瞭なピークとして検出されている。したがって、hamming taper のようにスペクトルの漏れの小さなテーパーを用いることが、正しいスペクトルピークの検出に必要である。

一方、図 3.4 に示す例は、

$$y(t) = \cos(2\pi t / 50) + \cos(2\pi t / p), \quad p = 13 + (17 - 13)(t - 1) / 99$$

という時系列に対するスペクトル推定である。これは周期 50 と、周期が 13 から 17 に変化する変動との足しあわせである。この場合は、テーパーをかけないスペクトル推定のみが、95% 信頼限界で、二つのピークを弁別することができている。テーパーをかけた場合は、二つの main lobe が混ざり合っその間のスペクトルの谷が弱くなってしまっている。このように、テーパーをかけない方が、正しい結果を与える場合もあり得る。ただしこのような場合は少なく、一般にはテーパーをかける方が適切なスペクトル推定値が得られる。

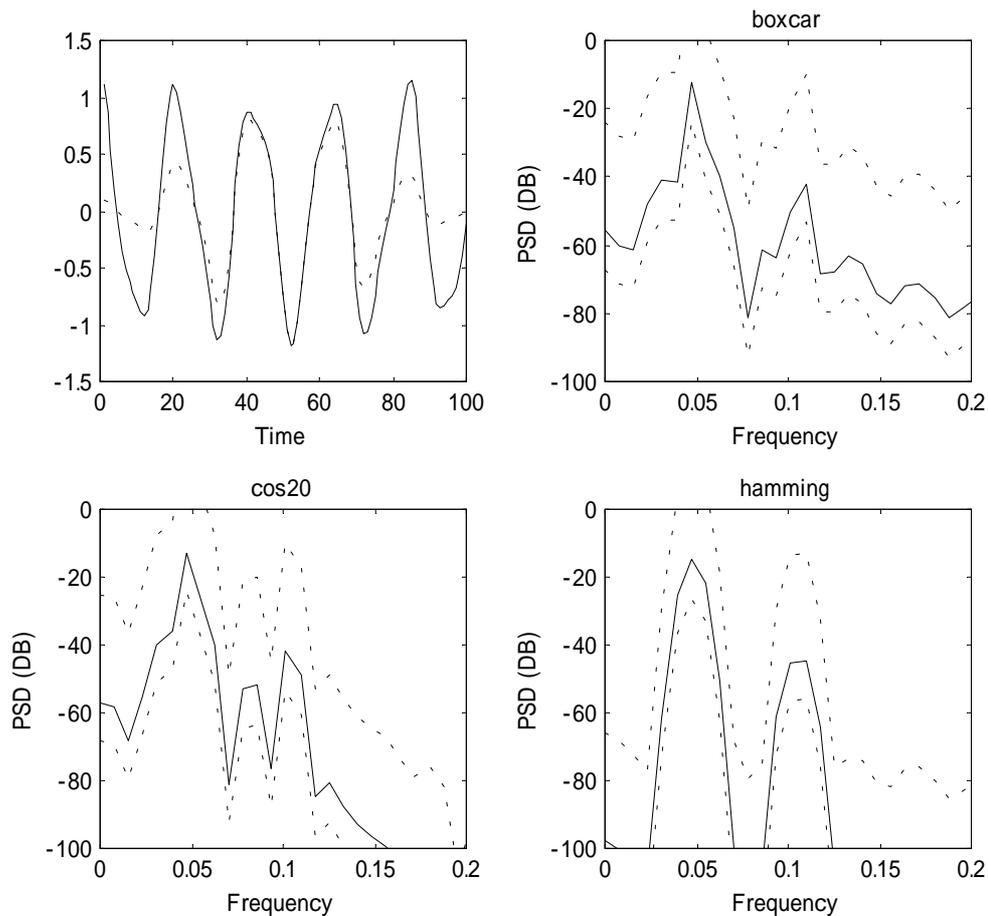


図 3.4 (左上) 入力時系列 (実線) と, 入力時系列に hamming taper を適用した時系列 (破線). この時系列を, boxcar (右上), cos20(左下), hamming (右下)の各テーパーを用いて, 得られたスペクトル密度関数のデシベル ($20 \log_{10}$) 表示. スペクトル推定は, 128 点にゼロパディングして, 平滑化無しで行った. スペクトルのパネルの点線は 95% 信頼限界である.

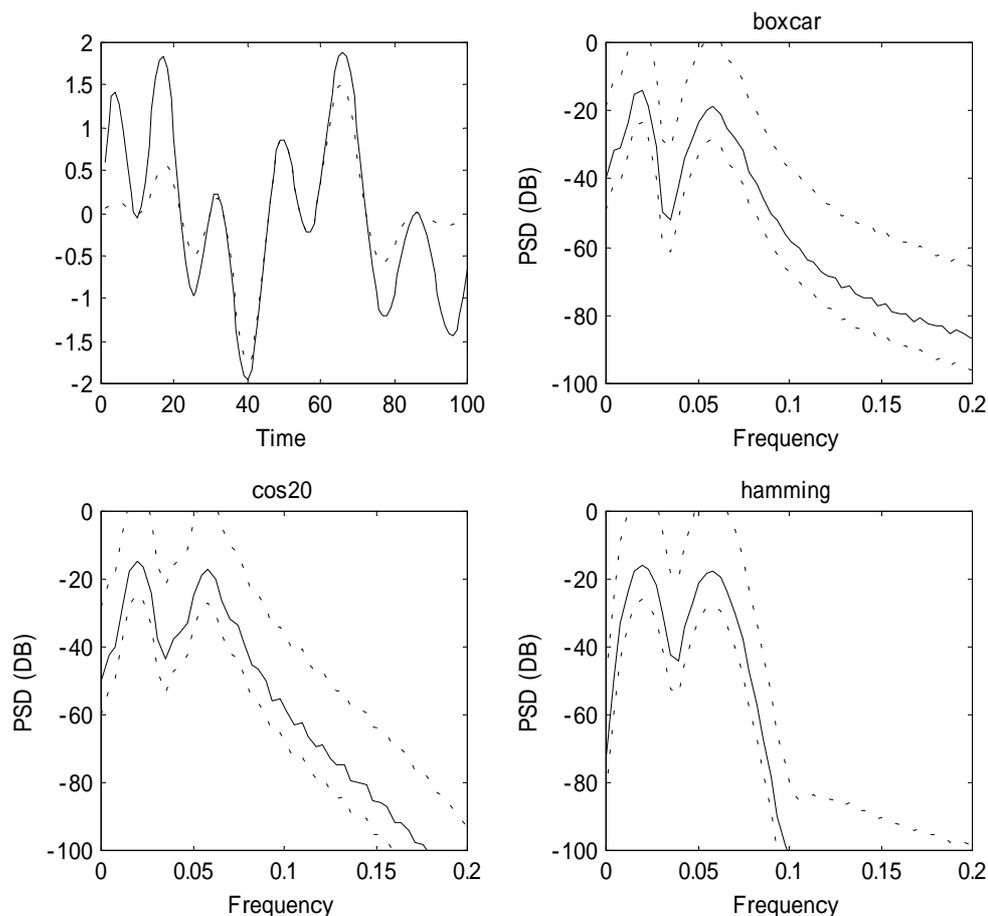


図 3.5 図 3.4 と同じ，ただし式で与えられる時系列について．スペクトル推定は，256 点にゼロパディングして，1-2-1 の重みつき周波数領域における移動平均で平滑化した．

FFT direct method でスペクトルを計算する際に，テーパーを用いないのは rectangle taper を用いるのと等価である．日野(1977)では，時系列の前後 10% づつに cosine 関数かけるテーパーを推奨しているが，一般的には，Hamming window などのより中央部に強い振幅を持つテーパーを奨励している．なお，テーパーを用いた場合のスペクトル推定値がどのように変わるかの議論は，川嶋・酒井(1989)，Percival and Walden (1993)などに紹介されている．

3.2.3. ゼロパディング

多くの場合， N 個のデータの後にゼロを K 個加えてから，フーリエ係数を求める．ゼロを加えることを**ゼロ・パディング(zero-padding)**と呼ぶ．ゼロ・パディングが用いられるのは，次の二つの理由のどちらか，あるいは両方である．

- 1) 高速フーリエ変換のライブラリーが 2 のべき乗個 (2 と 3 と 5 のべき乗個の場合もある) のデータ点からなるデータを扱うようにできている場合，適当な長さにしてからそれらのライブラリーを用いる．
- 2) 長周期の周波数成分の分解能がゼロ・パディング無しでは，荒い場合に，より細かい周波

数分解能を得るためにゼロパディングを行なう。

上の(1)は消極的なゼロパディングの利用であり,(2)は積極的なゼロパディングの利用である。ゼロパディングを行なった場合の両側スペクトル密度は次式で与えられる。

$$S_{yy}(f_k) = \frac{1}{(N+K)} |Y(f_k)|^2 = \frac{1}{N} \left| \sum_{n=0}^{N+K-1} y_n \exp(-i2\pi f_k n \Delta t) \right|^2, \quad (3.11)$$

$$f_k = k / ((N+K)\Delta t), \quad -f_N \leq f_k \leq f_N$$

狭い意味では、ピリオドグラムとは、以下で述べるテーパーを適用せず、また、平滑化も行わないスペクトルの推定値を意味する。この場合、テーパーを用いたスペクトル推定値を**直接スペクトル推定値**(direct spectrum estimator)あるいは**修正されたピリオドグラム**(modified periodogram)と呼ぶ。さらに平滑化を行なったスペクトル推定値を**平滑化された直接スペクトル推定値**(directly smoothed direct spectrum estimator)と呼び(Percival and Walden, 1993)、これが超九節法における最終的なスペクトル推定値である。

ゼロパディングの効果を見るために、100年のデータ長がある場合を考えよう。フーリエ成分の周期は、100年、50年、33年、25年、...と離散的である。ここで、11.1年と10.0年のサンプリング周期の間に当たる10.52年の周期成分のみからなる時系列を作成しよう(図4.1)。その時系列を、ゼロパディング無しで求めたスペクトルと、ゼロパディングを用いて求めたスペクトルとを、図4.2に示す。ゼロパディングを用いた、平滑化されたスペクトル推定値におけるピークのみが、95%の信頼限界で有意なピークとして検出されている。したがって、この場合には、ゼロパディングを用いることが、適切なピークの検出に有効であった。ただし、右上のパネルに示されるように、ゼロパディングを行なった場合に平滑化を行わなければ、side lobe への影響に非常に強い波打ち(ripple)が生じていることが見て取れる。したがって、ゼロパディングを行なった場合に、平滑化を行なうことは必ず必要である。

なお、ここで用いた入力時系列については、hamming テーパーを適用すれば、ゼロパディング無しでも、スペクトルピークを検出することはできる。

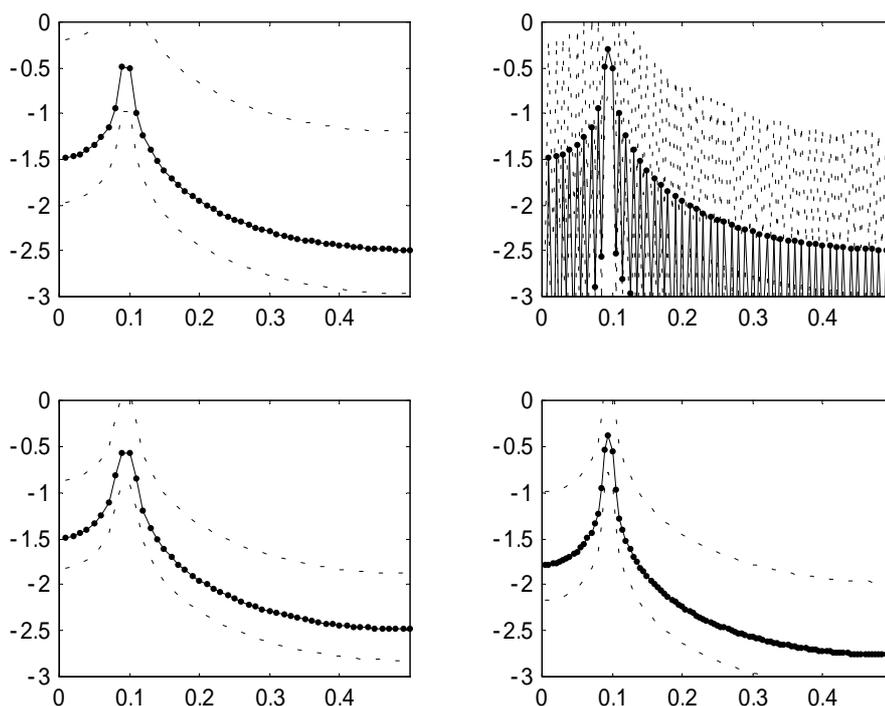


図 3.6 10.52 年の周期成分を持つ 100 年のデータから、ゼロパディング無しで求めたスペクトル（左）と、100 年のゼロを加えて求めたスペクトル密度関数（右）。上段は、平滑化無しでのスペクトルで、下段は周波数領域において 1-2-1 の重み付き移動平均によって平滑化を行なった。実線と点がスペクトル密度を、破線がその 95% 信頼限界を示す。点の水平位置はまた、求められた周波数成分を示している。等価自由度は、1（左上）、2（右上）、2.66（左下）、1.83（右上）である。なお、テーパーは用いていない。

3.2.4. Matlab でのスペクトル計算

Matlab で標準的な直接法のスペクトル計算関数としては、**psd**、**pwelch** という二つの関数が提供されている。ざっと見るにはこれらを使うので十分であろうけれど、研究目的で使うには、自分でプログラムを作る方がよいだろう。そこで、直接法の周波数領域での平滑化によってスペクトルを計算し図示するプログラムを作ってみよう。以下では、`tsrs` にスペクトルを求めるべき時系列が入っているものとする。

```

x=(1:1:100)';
tsrs=sin(x*2*pi/10);
subplot(2,1,1); plot(x,tsrs) %ここまでは時系列の作成と表示
tlng=length(tsrs); %時系列の長さ
nfft=256; %fft を取るデータ数．nfft-tlng がゼロパディングする数
avfnc=ones(5,1); %平滑化ウィンドウ
avfnc=avfnc./sum(avfnc); %後で説明する平滑化ウィンドウの規格化
taper=hamming(tlng); % hamming taper
tsrs_tapered=tsrs.*taper; %テーパーをかける
    
```

```

tsrs_tapered(tlng+1:nfft)=0.;           % ゼロパディング
famp=fft(tsrs_tapered);                 % フーリエ変換
famp=famp(1:nfft/2+1)/tlng;             % 実数時系列のスペクトルなら，この範囲のみが必要．
ds_psd=abs(famp.^2);                    % 直接スペクトル値
dsm_psd=conv(ds_psd,avfnc);             % 畳込みを使って平滑化
nav=(length(avfnc)-1)/2;
dsm_psd = dsm_psd (nav+1:end-nav);      % 畳み込みで増えた部分を除く
frq=(0:nfft/2)/nfft;                    % 周波数
subplot(2,1,2); plot(frq,dsm_psd, frq,ds_psd)
    
```

3.3. 直接スペクトル推定法の誤差理論

3.3.1. 推定誤差理論の基礎

簡単のために，ゼロパディング無し，テーパー無しのピリオドグラムを， n 点移動平均によって平滑化する場合を考えよう．この平滑化によってカイ二乗分布の自由度は2から $2n$ へ増加する．スペクトルの実部と虚部は，平均がゼロで等しい分散を持ち， $\chi^2 = \left\{ \sum_{i=1}^n (Y_i^R)^2 + (Y_i^I)^2 \right\} / Y^2 = n\hat{S}/S$ が自由度 $2n$ のカイ二乗分布に従う．ここで S は真のスペクトル， \hat{S} はスペクトルの推定値である．ある危険率 α でカイ二乗分布に従う変数の分布確率を

$$P\left(\chi_n^2\left(1-\frac{\alpha}{2}\right) < \chi^2 < \chi_n^2\left(\frac{\alpha}{2}\right)\right) = 1-\alpha$$

と表せることから， $\chi^2 = n\hat{S}/S$ を代入すると，

$$P\left(\frac{n\hat{S}}{F_\chi^{-1}\left(\frac{\alpha}{2}, 2n\right)} < S < \frac{n\hat{S}}{F_\chi^{-1}\left(1-\frac{\alpha}{2}, 2n\right)}\right) = 1-\alpha \tag{3.12}$$

となって，スペクトルの誤差を含めて推定することができる．ここで F_χ^{-1} は1章で導入した，カイ二乗分布のICDFである．なお，同じ意味であるけれど，信頼区間を

$$\left(\frac{F_x^{-1}\left(\frac{\alpha}{2}, 2n\right)}{F_x^{-1}(0.5, 2n)}, \frac{F_x^{-1}\left(1-\frac{\alpha}{2}, 2n\right)}{F_x^{-1}(0.5, 2n)}\right) \tag{3.13}$$

で推定することもできる．これは，後に示すようにモンテカルロシミュレーションで中央値を基準にして推定幅を評価するのに適した表現である．

一般のスペクトル計算では， n 点移動平均以外の平滑化が用いられ，ゼロパディングが行な

われ、テーパーが用いられる。このような場合にも、スペクトルの推定誤差を統一的に記述する道具として、**等価自由度(equivalent degree of freedom, EDOF)**を導入しよう。上の最も簡単なピリオドグラム推定では各周波数成分は1の等価自由度を持つと約束する。したがって、カイ二乗分布の自由度は、等価自由度の2倍である。

3.3.2. 等価自由度への平滑化方法の影響

周波数領域で平滑化する場合に、一般にはn点移動平均ではなく、適当なフィルター関数 Q (周波数領域でのウィンドウ関数) を用いて、

$$\bar{s}(f) = \int_{-\infty}^{\infty} s(f') Q(f - f') df'$$

と平滑化を行う。このフィルター関数 Q を、**平滑化ウィンドウ(smoothing window)**と呼ぶ²。ただし全体的なスペクトルのレベルを変えないために、 $\int_{-\infty}^{\infty} Q(f) df = 1$ と規格化しておく。n点移動平均の場合には、nという幅が平滑化の幅であった。他の一般の平滑化の場合にも、同じように幅を定義できれば便利である。そこで、スペクトル・ウィンドウの分散を求めて、それと同じ分散を持つ移動平均の幅を、**等価バンド幅(equivalent band width)**と定義する。n点移動平均の分散は、 $\sigma^2 = \sum_{i=1}^n (1/n)^2 = 1/n$ であり、スペクトル・ウィンドウの分散は、 $\int_{-\infty}^{\infty} Q^2(f) df$ なので、等価バンド幅、 B_e を、

$$B_e = \frac{1}{\int_{-\infty}^{\infty} Q^2(f) df}, \tag{3.14}$$

と定めるのが適当である。この等価バンド幅を用いると、等価自由度は

$$N_e = B_e$$

で与えられる。この2倍の自由度を持つカイ二乗分布について(3.12)や(3.13)を用いることで、スペクトルの推定範囲が決定できる。

3.3.3. 等価自由度へのテーパーとゼロパッチングの影響

テーパーは、データの振幅を減少させる効果がある。テーパー $T(t)$ の最大値を1とする。この場合、平均振幅がどれだけテーパーによって減少するかが \bar{h} で示される。この減少分だけ情報が失われる。ただし、ある周波数についてのスペクトル推定は、依然1の等価自由度を持ち。またその情報の喪失は隣接した周波数でのスペクトル推定値が独立性が減じるという形をとる。これは、テーパーが解像度を低下させるという性質と整合的である。したがって、この影響を考慮

² これを spectral window と呼ぶ流儀もある(例えば大崎(1994))。ここでは、Percival and Walden (1994) p. 238 に従い平滑化ウィンドウと呼び、spectral window は、データ・テーパーのフーリエ変換に確保している。またこの流儀は、von Storch and Zwiers (1999)でも共通であり、主流はこちらと思われる。

した場合の等価自由度は，

$$N_e = \left[1 + (B_e - 1) \bar{h} \right]$$

で与えられる．

ゼロパッチングの効果も，テーパーと同様に，1周波数の自由度を減じるわけではなく，平滑化された後の自由度が低下する．もともとの時系列の長さが N であるとして， K 点の周波数点をゼロパッチングで付け加えるなら，等価自由度は次式で与えられる．

$$N_e = \left[1 + (B_e - 1) \frac{N}{N + K} \bar{h} \right] \quad (3.15)$$

(3.15)式が，平滑化・テーパー・ゼロパッチングを施した場合の等価自由度である．なお(3.15)に相当する式は，いろいろ教科書を探したのだけれど載っていなかった．そこで，モンテカルロ法でテストするとともに考えて，多分これでいいだろう，と結論している．ひょっとしたら嘘かもしれないので，気をつけること．

3.3.4. スペクトルの計算定順（周波数領域での平滑化）

- 1) (必要なら) データの線形増加・減少成分を取り除く
- 2) (必要なら) 観測されたラグ 1 相関係数を用いて， $y'(t) = y(t) - ry(t-1)$ なる操作を行って，新たな時系列 y' を作成してそれでスペクトル推定を行う (pre-whitening)
- 3) (必要なら) スペクトルの漏れを減じるために，データテーパーを用いる．
- 4) スペクトルを計算する．
- 5) 周波数空間で平滑化を行う．
- 6) スペクトルの推定誤差を評価する．推定誤差の評価には，上で述べたカイ二乗分布を用いるものの他に，モンテカルロ法を用いる場合も多い．

3.3.5. スペクトルの推定誤差のテスト

実際にスペクトルの推定誤差をテストしてみよう．まず，`psd_ex1` という次の関数を作ろう．これは上で紹介したスクリプトを Matlab 関数の形にまとめ，かつ，複数の時系列について効率よくスペクトルが計算できるようにしている．

```
function [ds_psd,frq]= psd_ex1(tsrs,nfft,ctaper,avfnc)
% function [ds_psd,frq]= psd_ex1(tsrs,nfft,ctaper,avfnc)
% Power Spectrum Density を直接法によって求める Matlab Script
% 平滑化は周波数領域で行なう．
% 複数時系列 (sz2>1) についても計算する．
% 入力
% tsrs: 入力時系列 (sz1, sz2)各列についてスペクトルを求める
% nfft: fft を計算する時間サンプル数
% ctaper: taper 関数('hamming','hanning'など)
% avfnc: 平滑化ウインドウ
```

```

% 出力
% ds_psd: スペクトル密度 . 1 × nfft/2+1 .
% frq: 周波数 . 1 × nfft/2+1 .
% todo
% PSD の定義に振幅がきちんと合うようにする .
[sz1,sz2]=size(tsrs);
if (sz1==1); tsrs=tsrs(:); end %tsrs を sz2*1 のデータに変換 .
if (sz1>nfft)
    error('*psd_ex1.m* sz1>nfft') %nfft は tlng よりも大きくなくてはならない
end % を満たさないとエラーメッセージを出して終了
avfnc=avfnc./sum(avfnc); %後で説明する平滑化ウインドウの規格化
str=sprintf('%s(%d)',ctaper,sz1); %taper 作成用文字列を例えば hamming(100)のように作る
taper=eval(str); %実際の taper 計算
for n=1:sz2
    tsrs(:,n)=tsrs(:,n).*taper; %時系列にテーパーをかける
end
tsrs(sz1+1:nfft,:)=0.; %ゼロパディング
famp=fft(tsrs); %フーリエ変換
famp=famp(1:nfft/2+1,:)/sz1; %実数時系列のスペクトルなら , この範囲のみが必要 .
d_psd=abs(famp.^2); %直接スペクトル値 (d: direct)
for n=1:sz2
    ds_psd(:,n)=conv(d_psd(:,n),avfnc); %畳み込みを使って平滑化 (ds: direct-smoothed)
end
nav=(length(avfnc)-1)/2;
ds_psd=ds_psd (nav+1:end-nav,:); %畳み込みで増えた部分を除く
frq=(0:nfft/2)/nfft; %周波数を求めておく
if (nargout==0)
    plot(frq,ds_psd); xlabel('frequency'); ylabel('psd')
end
return

```

これを次のようなスクリプトで呼んで , 様々なテストができる .

```

for npnl=1:4
    tlng=100; nsmpl=1000;
    if (npnl==1)
        nfft=100;
        taper=boxcar(tlng); ctaper='boxcar';
        avfnc=[1 1 1 1];
        cttl='no padding, boxcar, 5-running'
    end
end

```

```
elseif (npnl==2)
    nfft=100;
    taper=hamming(tlng); ctaper='hamming';
    avfnc=[1 1 1 1 1];
    cttl='no padding, hamming, 5-running'
elseif (npnl==3)
    nfft=200;
    taper=boxcar(tlng); ctaper='boxcar';
    avfnc=[1 1 1 1 1];
    cttl='double padding, boxcar, 5-running'
elseif (npnl==4)
    nfft=200;
    taper=hamming(tlng); ctaper='hamming';
    avfnc=conv([1 2 1],[1 2 1]);
    cttl='double padding, hamming, double 1-2-1'
end

tsrs=randn(tlng,nsmpl);
[ds_psd,frq]=psd_ex1(tsrs,nfft,ctaper,avfnc);

% Monte-Carlo 法で得る 90%信頼限界
srt_psd=sort(ds_psd,2); clear c;
c(:,1)=log10(srt_psd(:,nsmpl*0.05));
c(:,2)=log10(srt_psd(:,nsmpl*0.95));

avfnc=avfnc./sum(avfnc); % frequency domain の平滑化関数の規格化
c0=srt_psd(:,nsmpl*0.5);

edof=(1+(1/sum(avfnc.^2)-1)*tlng/nfft*mean(taper))    等価自由度の計算
clear ce;
ce(:,1)=log10(c0*chi2inv_nint(0.050,2*edof)/chi2inv_nint(0.50,2*edof));
ce(:,2)=log10(c0*chi2inv_nint(0.950,2*edof)/chi2inv_nint(0.50,2*edof));

subplot(2,2,npnl)
plot(frq,c,'b',frq,ce,'r'); title(cttl);
xlabel('frq'); ylabel('psd');
if (npnl==1); legend('Monte-carlo','Theory'); end

end
```

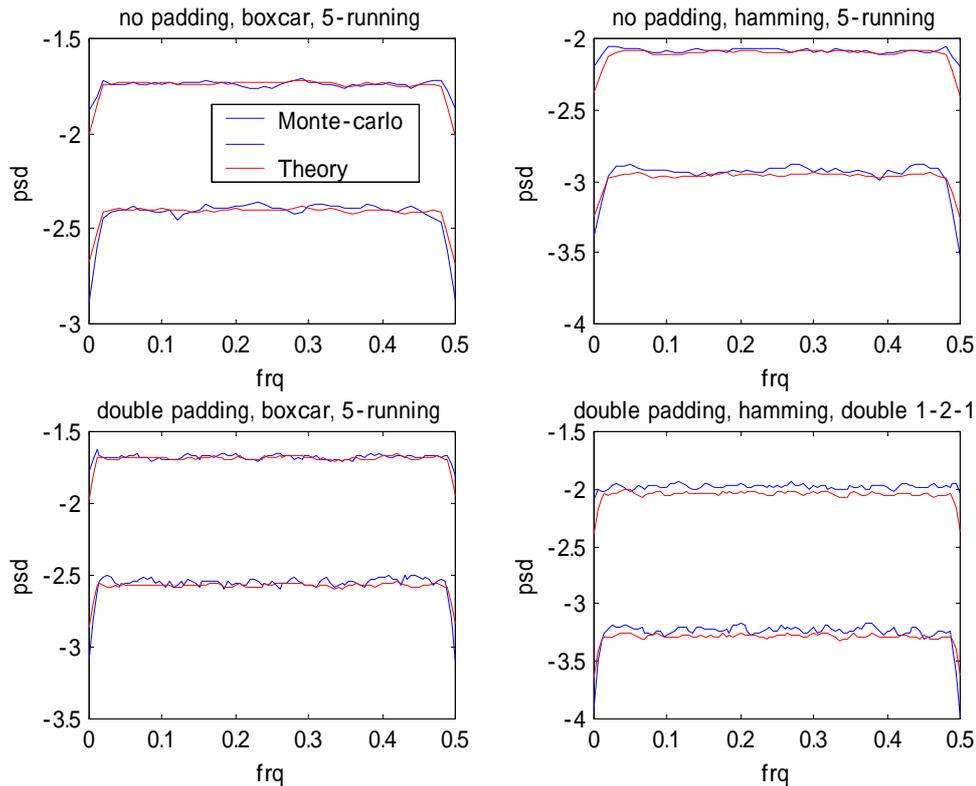


図 3.8 様々な直接スペクトル推定パラメータに対する，モンテカルロ法で得られた 90%信頼限界と，理論的に得られた信頼限界．

3.4. MEM

3.4.1. MEM の概説

すべてのスペクトル推定は，与えられている区間の外側をなんらかの形で推定（補外）している．MEM は最大の情報エントロピーを持つという補外となっており，これが名称である最大エントロピーのいわれになっている．この特徴から，MEM はより良い推定方法であるという主張がなされることがあり，時に妥当でない程度に強く主張されることもある．Press ら(1993)は，これを「新興宗教的な」と表している．

極を表現する MEM が鋭いピークを捕らえることができるのは事実であり，ゼロ周波数に近いスペクトルや，比較的隣接したピークでも運がよければ捕らえられる．しかしその一方で，本来一つのはずのピークが二つに分かれるピークの分離(peak splitting)や，偽のピーク(false peak)という問題があることも知られている．

また，ごく最近まで MEM の標準的な有意性検定方法が提案されていなかったために，偽のピークにだまされた研究もおそらくはあったであろう．一つの有意性検定方法は Percival and Walden (1992)に紹介されており，またその中の参考文献に他のいくつかの提案がある．それらの理論は，統計推定や検定の多くの理論がそうであるように，残念ながら簡単とは言えないが，モンテカルロ法で有意性を検定することは容易である．なお，MEM で現実的なデータから現実的なピークを得ようとするとき，比較的 AR モデルの項数を多くとらなくてはならず，その場合には偽のピークが多数立つことを覚悟しなくてはならない．

3.4.2. MEM スペクトルの推定方法

MEM スペクトルの推定は、2章で述べたように自己回帰モデルを推定し、その自己回帰モデルに基づいてスペクトルを推定するという手続きを取る。2章で説明したように自己回帰モデルは、

$$y(t) + \phi(1)y(t-1) + \dots + \phi(p)y(t-p) + \sigma_w \varepsilon(t) = 0$$

と書くことができる。このフーリエ変換は、

$$\sum_{t=0}^{N-1} \{y(t) + \phi(1)y(t-1) + \dots + \phi(p)y(t-p)\} \exp(-i2\pi k t / N) = -\sigma_w \sum_{t=0}^{N-1} \varepsilon(t) \exp(-i2\pi k t / N) \quad (3.16)$$

ここで、

$$\begin{aligned} \sum_{t=0}^{N-1} y(t-\tau) \exp(-i2\pi k t / N) &= \sum_{t=0}^{N-1} y(t-\tau) \exp(-i2\pi k(t-\tau) / N) \exp(-i2\pi k \tau / N) \\ &= Y(k) \exp(-i2\pi k \tau / N) \end{aligned}$$

であることに注意すれば、(3.16)の左辺は

$$\begin{aligned} \text{左辺} &= Y(k) \left\{ 1 + \phi(1) \exp(-i2\pi k / N) + \phi(2) \exp(-i4\pi k / N) + \dots + \phi(p) \exp(-i2\pi p k / N) \right\} \\ &= Y(k) \left\{ 1 + \sum_{\tau=1}^p \phi(\tau) \exp(-i2\pi k \tau / N) \right\} \end{aligned}$$

と書くことができる。一方右辺の積分は白色ノイズのフーリエ変換なので、位相は定まらないが、大きさは周波数によらず1である。よって、上の左辺の式も、右辺と組み合わせることができるように、2乗して期待値を取ると、

$$\left\langle Y(k)^2 \right\rangle \left| 1 + \sum_{\tau=1}^p \phi(\tau) \exp(-i2\pi k \tau / N) \right|^2 = \sigma_w^2$$

よって、MEM スペクトルのスペクトル推定値が、

$$\hat{Y}(k)^2 = \frac{\sigma_w^2}{\left\{ 1 + \sum_{\tau=1}^p \phi(\tau) \exp(-i2\pi k \tau / N) \right\}^2} \quad (3.17)$$

と求まる。このスペクトルの分母はゼロになり得る、すなわち極を持つが、分子がゼロになるつまりゼロ点を持つことはない。このためにMEM スペクトルは**全極(all pole)**スペクトルとも言う。なお、MA 過程から推定されるスペクトルは、ゼロ点のみがあって極は無いので、**全ゼロスペクトル**という。また、ARMA 過程から求められるスペクトルでは、極もゼロ点存在する。

3.4.3. Matlab での MEM の計算

Matlab での MEM の計算は、**pmem**, **pyulear**, **pburg** で行うことができる。pyulear と pburg はそれぞれ、Yule-Walker のアルゴリズムと Burg のアルゴリズムだけれど、pmem は両者とどういう関係にあるのか不明である。これら3つは例えば、次の結果で見れるように、わずかに異なったスペクトル推定を行う。

```
tlng=100;
```

```

rtsrs=randn(tlng,1);
fs=1; order=10; nfft=100; %fs はサンプリング周波数
[pm1,frq]=pmem(rtsrs,order,nfft,fs);
[pm2,frq]=pburg(rtsrs,order,nfft,fs);
[pm3,frq]=pyulear(rtsrs,order,nfft,fs);
plot(frq,pm1,frq,pm2,frq,pm3)

```

さて、MEM の始めに、MEM は不合理なほどに好まれることがあると述べたが、では他のスペクトル推定と比べて MEM は良いのであろうか？この問題を、次のプログラムを実行することで MEM の性質をいろいろ調べてみる事ができる。

```

%----- <phase randomization で代理時系列を作成する>
tlng=100; nfft=200; nh=nfft/2; nhp1=nfft/2+1; nsmpl=1000;
frqx=(1:1:nhp1)-1)/nhp1*0.5;
frqc1=0.11; fwid1=0.01; amp1=1.0;
spct_gvn=amp1*exp(-(frqx-frqc1).^2./fwid1^2);
frqc2=0.05; fwid2=0.01; amp2=0.7;
spct_gvn=amp2*exp(-(frqx-frqc2).^2./fwid2^2)+spct_gvn;
frqc3=0.25; fwid3=0.02; amp3=0.0;
spct_gvn=amp3*exp(-(frqx-frqc3).^2./fwid3^2)+spct_gvn;
nlvl=0.01; spct_gvn=spct_gvn+nlvl;

```

```

subplot(3,2,1); plot(frqx,log10(spct_gvn)); title('given spectrum')
xlabel('frq'); ylabel('log10(psd)')
famp=sqrt(spct_gvn);
famp=famp(:)*ones(1,nsmpl);
fphs=rand(length(frqx),nsmpl)*2*pi;
fcmp=famp.*exp(i*fphs);
fftamp=zeros(nfft,nsmpl);
fftamp(1:nhp1,:)=fcmp;
fftamp(nhp1+1:end,:)=conj(flipud(fcmp(2:nh,:)));
rtsrs=real(iff(fftamp)); rtsrs=rtsrs(1:tlng,:);
x=[1:1:tlng]';
subplot(3,2,2); plot(x,rtsrs(:,1)); title('sample time series')
xlabel('time'); ylabel('y')
%----- <phase randomization で代理時系列を作成する>

```

```

%----- <直接法でのスペクトル推定>
ctaper='boxcar'; nfft=200; avfnc=[1 2 1];
[ds_psd1,frq_ds1]=psd_ex1(rtsrs,nfft,ctaper,avfnc);

```

```

ctaper='hamming'; nfft=100; avfnc=[1 2 1];
[ds_psd2,frq_ds2]=psd_ex1(rtsrs,nfft,ctaper,avfnc);
%----- <直接法でのスペクトル推定>

```

```

%----- <MEM でのスペクトル推定>
fs=1.0; nfft=256;
clear pmy pmb;
order=33;
for n=1:nsmpl
    [pmb1(:,n),frq]=pyulear(rtsrs(:,n),order,nfft,fs);
end

```

```
order=10;
for n=1:nsmpl
    [pmb2(:,n),frq]=pyulear(rtsrs(:,n),order,nfft,fs);
end
%----- </MEM でのスペクトル推定>

clear pmyest pmbest pdsest1 pdsest2;
srt_dsp1=sort(ds_psd1,2)*tln;
pdsest1(:,1)=srt_dsp1(:,0.05*nsmpl);
pdsest1(:,2)=srt_dsp1(:,0.5*nsmpl);
pdsest1(:,3)=srt_dsp1(:,0.95*nsmpl);
subplot(3,2,3); plot(frq_ds1,log10(pdsest1));
title('Direct-PSD, double-zero, 1-2-1, boxcar')
xlabel('frq'); ylabel('log10(psd)')
srt_dsp2=sort(ds_psd2,2)*tln;
pdsest2(:,1)=srt_dsp2(:,0.05*nsmpl);
pdsest2(:,2)=srt_dsp2(:,0.5*nsmpl);
pdsest2(:,3)=srt_dsp2(:,0.95*nsmpl);
subplot(3,2,4); plot(frq_ds2,log10(pdsest2))
title('Direct-PSD, no-zero, 1-2-1, hamming')
xlabel('frq'); ylabel('log10(psd)')
srt_pmb1=sort(pmb1,2);
pmb1est(:,1)=srt_pmb1(:,0.05*nsmpl);
pmb1est(:,2)=srt_pmb1(:,0.5*nsmpl);
pmb1est(:,3)=srt_pmb1(:,0.95*nsmpl);
subplot(3,2,5); plot(frq,log10(pmb1est))
title('pburg, 33-order');
xlabel('frq'); ylabel('log10(psd)')
srt_pmb2=sort(pmb2,2);
pmb2est(:,1)=srt_pmb2(:,0.05*nsmpl);
pmb2est(:,2)=srt_pmb2(:,0.5*nsmpl);
pmb2est(:,3)=srt_pmb2(:,0.95*nsmpl);
subplot(3,2,6); plot(frq,log10(pmb2est))
title('pburg, 10-order');
xlabel('frq'); ylabel('log10(psd)')
legend('5%','50%','95%')
```

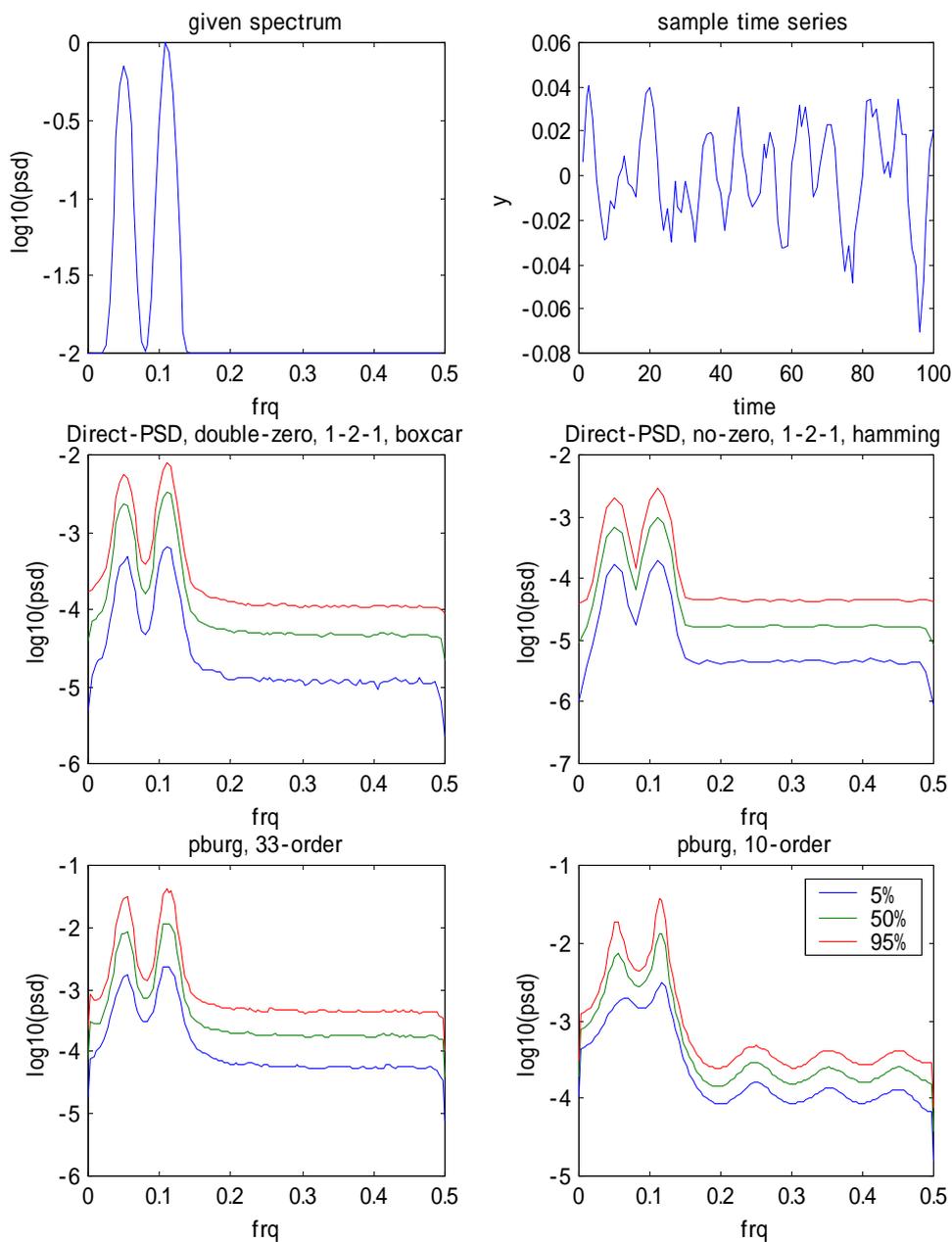


図 3.9. 上のプログラムの実行結果 .

自分で実際にいろいろとパラメータを代えて見れば , 適切に自己回帰モデルの次数を選ぶならば , MEM は概して直接スペクトルと同等かよりよい検出力を発揮することが分かるであろう . しかし , 適切な次数を推定することは , それ自身容易ではない .

3.5. スペクトルの有意性判定の注意

例えば 50 点の周波数で独立なスペクトルの推定値が得られる場合を考えよう . さらに , 赤色ノ

イズで期待されるスペクトル分布の上位 5% に観測スペクトルが入った場合に、それを 5% 有意水準で有意とみなすとする。では、単なる赤色ノイズがどこかの周波数に有意とみなされるスペクトルを持つ確率はどの程度あるだろう？

一見すると 5% のように感じるかもしれないが、実際には $1-0.95^{50}=0.92$ であって、有意とみなされるピークが 1 個以上存在する確率は 92% である。また平均して 2.5 個の有意なスペクトルが見出されると期待される。

5% 信頼限界で有意になるということは、偶然で説明できる確率は 5% 以内ということの意味するが、それは個々の周波数に関して有意なスペクトルが偶然得られる確率が 5% ということであって、任意の周波数のどれかに有意とみなされるスペクトルが偶然得られる確率はずっと大きい。したがって、有意なスペクトルが得られたとしても、それが本当に物理的に意味があるかどうかは、他の方法も併用して注意深く調べる必要がある。

参考文献

大崎 順彦, 1994: 新・地震動のスペクトル解析入門. 鹿島出版会, p. 299.

Percival D. B. and A. T. Walden, 1993: Spectral analysis for physical applications: Multaper and conventioinal univariate techniques. Cambridge University Press, pp. 583.

Press, W. H., B. P. Flannery, S. A. Teukolsky, W. T. Vetterling, 1993: Numerical Recipes in C [日本語版], 技術評論社, pp. 685.

von Storch, H. and F. W. Zwiers, 1999: Statistical analysis in climate research. Cambridge University Press, pp. 483 (ISBN: 0 521 45071 3)