

内容安排的基本逻辑



第六讲 频域分析与处理

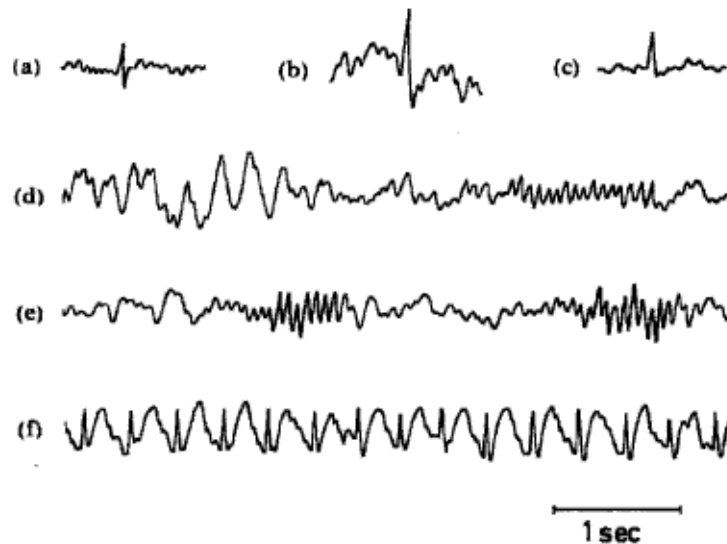
— 傅里叶变换与功率谱估计

学习要点

- 确定性信号的快速傅里叶变换
 - 基本原理与编程实践
 - 掌握傅里叶变换应用中的常见问题
 - 时间分辨率、频率分辨率、频谱泄漏、加窗等
- 随机信号的功率谱估计
 - 理解为什么做随机信号的功率谱估计
 - 掌握功率谱估计的经典方法
 - 掌握非参功率谱估计的改进方法
- 功率谱分析在睡眠分期的应用
 - 应用条件
 - 特征提取与分析

振荡是基本的生命现象

- 周期性活动是物理现象、生命现象、工程系统中的基本信息
- 例如如下脑电信号



EEG patterns occurring during various conditions

- (a) Biphasic spike
- (b) Biphasic sharp wave
- (c) Monophasic sharp wave
- (d) Sleep EEG (N3, N2)
- (e) Sleep EEG (N2)
- (f) Epilepsy (spike-and-waves)

振荡消失可能对应着生命的消逝

EEG guidelines in the diagnosis of brain death

L'EEG dans le diagnostic de mort cérébrale

W. Szurhaj^{a,b,*}, M.-D. Lamblin^a, A. Kaminska^c, H. Sediri^a

^a Service de neurophysiologie clinique, CHRU, 59037 Lille cedex, France

^b Faculté de médecine, Université Lille 2, Lille, France

^c Laboratoire de neurophysiologie clinique, Hôpital Necker-Enfants Malades, AP-HP, Paris, France

Neurophysiologie Clinique/Clinical Neurophysiology (2015) 45, 97–104



Disponible en ligne sur

ScienceDirect
www.sciencedirect.com

Elsevier Masson France

EM|consulte
www.em-consulte.com/en

例如法国对脑死亡诊断的EEG标准制定了指导手册 (Szurhaj et al., 2015)
In France, for the determination and diagnostic validation of **brain death** the law requires either two EEG recordings separated by a 4-hour observation period, both showing **electrocerebral inactivity** (间隔4小时观察的两次脑电记录均显示无脑皮层的电生理活动, 确诊为脑死亡) ...

Electrocerebral inactivity may be confirmed when a 30-minute good quality EEG recording shows **complete electrocerebral silence**, defined as no cerebral activity greater than 2 μ V, having first ruled out the possible influence of sedative drugs, metabolic disorders or hypothermia. 在先排除镇静药物、代谢紊乱、或者失温的情况下, 30分钟高质量EEG显示完全安静的皮层电活动, 脑电小于2微伏, 也可以确认脑皮层电活动失活。

傅里叶分析是最基本的谐波分析方法

- 傅里叶分析（频谱分析）是揭示振荡规律的重要工具
 - 应用：提取信号的频率特征
 - 特别地，后面的实例展示在有些情况下，频率特征具有更高的信噪比；需要理解为什么？（联系在建模一节讨论白噪声的模型）
- 在频率域内进行信号处理更加直观和高效
 - 例如频域内滤波，借助快速傅里叶-逆傅里叶变换，更快速实现滤波算法

确定性信号的傅里叶分析

讨论要点

- 傅里叶变换的基本条件
- 信号类型与傅里叶分析形式
 - 周期信号 vs. 非周期信号
 - 连续时间信号 vs. 离散信号
- 傅里叶变换应用的注意点
 - 时间分辨率
 - 频率分辨率与延拓
 - 频谱泄露与加窗

级数展开

- 常数项级数
 - 如果 $s_n = \sum_{i=1}^n u_i$ 当 $n \rightarrow \infty$ 的时候, 数列 $\lim_{n \rightarrow \infty} s_n = s$ 有限, 则称该常数项级数收敛
 - 例如 $1, 1/2, 1/4, 1/8, \dots, 1/2^n$ 的和的极限为 $1/(1-1/2) = 2$
 - 实际上, 对于任何等比数列, 比例为 q , $|q| < 1$ 的时候, 数列之和收敛
- 函数的级数展开

函数的级数展开

• 例如：

设 $f(x)$ 在 x_0 的邻域内任意阶可导，则

$$f(x) = \sum_{n=0}^{\infty} a_n (x - x_0)^n,$$

$$\text{其中 } a_n = \frac{f^{(n)}(x_0)}{n!} \quad (n = 0, 1, 2, \dots).$$

特别地，当 $x_0 = 0$ 时，

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(0)}{n!} x^n,$$

称 $\sum_{n=0}^{\infty} \frac{f^{(n)}(0)}{n!} x^n$ 为 $f(x)$ 的麦克劳林级数

傅里叶级数

- 法国数学家约瑟夫·傅里叶(Fourier)提出满足一定条件的函数可以展开为三角级数

$$f(t) = A_0 + \sum_{n=1}^{\infty} A_n \sin(n\omega t + \varphi_n)$$

其中 A_0, A_n, φ_n ($n = 1, 2, 3, \dots$) 都是常数

$$A_n \sin(n\omega t + \varphi_n) = A_n \sin \varphi_n \cos n\omega t + A_n \cos \varphi_n \sin n\omega t$$

并且令 $\frac{a_0}{2} = A_0, a_n = A_n \sin \varphi_n, b_n = A_n \cos \varphi_n, \omega = \frac{\pi}{l}$ (即 $T = 2l$), $f(t)$ 的级数

可以改写为

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_n \cos \frac{n\pi t}{l} + b_n \sin \frac{n\pi t}{l} \right).$$

傅里叶变换存在的条件

- **傅里叶积分定理** 若函数 $f(t)$ 在区间 $(-\infty, \infty)$ 或者周期 $[0, T]$ 上满足狄利克雷条件(Dirichlet Conditions)
 - 连续或只有有限个第一类间断点
 - 极大值和极小值的数目应是有限个
 - 信号是绝对可积的
- 一般我们遇到的周期信号都能满足狄利克雷条件
- 狄利克雷条件是一个信号存在傅里叶变换的充分不必要条件

周期为P的信号f(t)的傅里叶级数

- 定义

$$\begin{aligned}f(t) &= \sum_{m=-\infty}^{\infty} z_m \Phi_m(t) = \sum_{m=-\infty}^{\infty} z_m e^{j2\pi m f_0 t} \\&= \cdots + z_{-m} \Phi_{-m}(t) + \cdots + z_{-1} \Phi_{-1}(t) + z_0 \Phi_0(t) + z_1 \Phi_1(t) + \cdots + z_m \Phi_m(t) + \cdots \\&= \cdots + z_{-m} e^{-j2\pi m f_0 t} + \cdots + z_{-1} e^{-j\omega_0 t} + z_0 + z_1 e^{j\omega_0 t} + \cdots + z_m e^{j2\pi m f_0 t} + \cdots\end{aligned}$$

- 其中

$$\begin{aligned}\Phi_m(t) &= e^{jm\omega_0 t} \quad \text{for } -\infty \leq m \leq \infty \text{ and } \omega_0 = \frac{2\pi}{P} = 2\pi f_0 \\Z_m &= \frac{1}{\lambda_m} \int_0^P f(t) \Phi_m^*(t) dt \\&= \frac{1}{P} \int_0^P f(t) e^{-j2\pi m f_0 t} dt \quad \text{for } -\infty \leq m \leq \infty\end{aligned}$$

- 连续周期性的时域信号 \leftrightarrow 离散非周期的级数（频率域）
- P越大，频率 ω_0 越小，频率分辨率越高（即 $m\omega_0$ 和 $(m+1)\omega_0$ 的间隔越小），P无限增大，则频率点接近连续

傅里叶变换

- P可以为无穷大，则一般形式

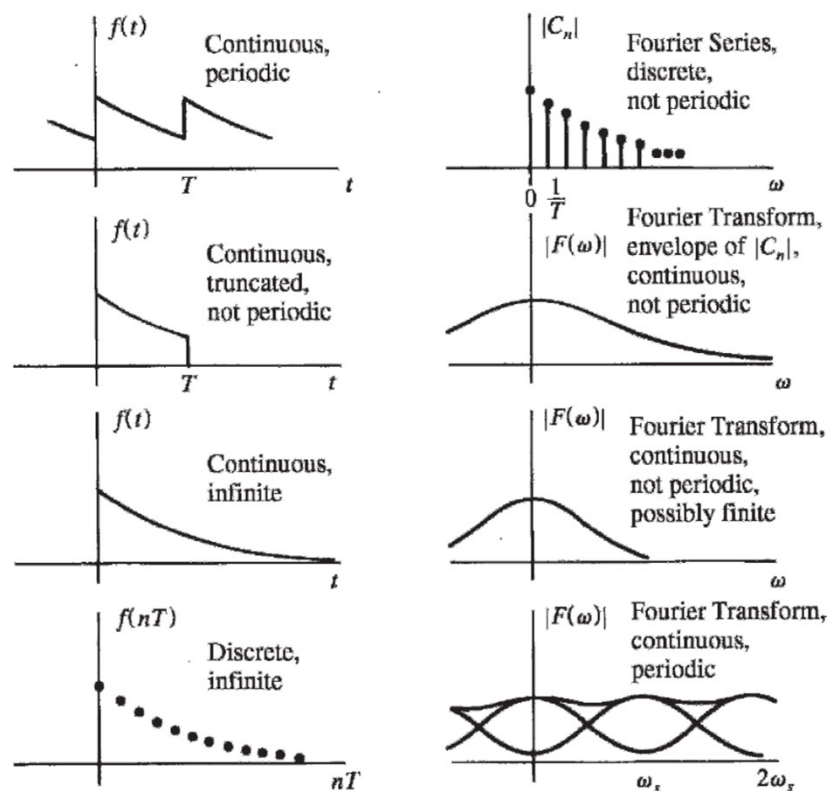
$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt \quad \text{and} \quad x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df$$

- 并没有要求信号的周期性，积分区间为 $(-\infty, \infty)$
- 时域连续非周期，频域连续非周期

周期与非周期信号 vs. 谱特性

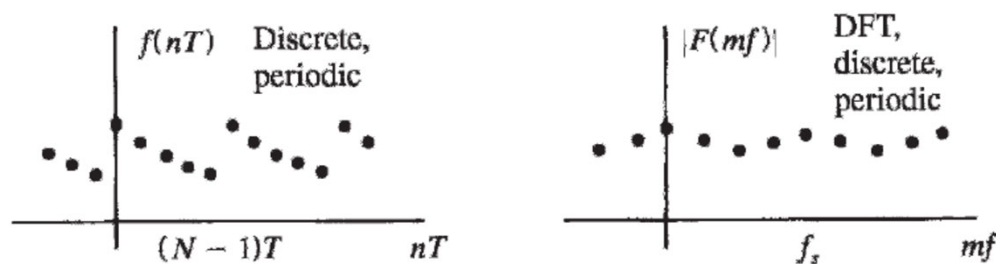
$$\Phi_m(t) = e^{jm\omega_0 t} \quad \text{for } -\infty \leq m \leq \infty \text{ and } \omega_0 = \frac{2\pi}{P} = 2\pi f_0$$

- 周期信号
 - 离散的谱线
 - 间隔为 ω_0
- 如果周期 $P \rightarrow \infty$
 - 则 $\omega_0 \rightarrow 0$
 - 频域内的谱线间隔趋近为0, 即连续谱



离散傅里叶变换

- 离散时间信号，采样率 F_s
 - 周期延拓，变换成周期信号，经过傅里叶变换，获得频域内离散的线谱
- 频率分辨率，即相邻频率点间隔， $f_0=1/P$
 -
- 实际的一段类 $\Phi_m(t) = e^{jm\omega_0 t}$ for $-\infty \leq m \leq \infty$ and $\omega_0 = \frac{2\pi}{P} = 2\pi f_0$
 - 时域周期延拓，傅里叶变换，获得离散频谱
 - 对频谱行周期延拓，傅里叶逆变换，获离散时域信号



信号类型与傅里叶分析形式

	傅里叶级数	傅里叶变换
时域 ↕ 频域	$\tilde{x}(t) = \sum_{k=-\infty}^{+\infty} a_k e^{j\omega_0 k t}$ <p>时域是连续周期的</p>	$x(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} X(j\omega) e^{j\omega t} d\omega$ <p>时域是连续非周期的</p>
	$a_k = \frac{1}{T_0} \int_{T_0} \tilde{x}(t) e^{-j\omega_0 k t} dt$ <p>频域是离散非周期的</p>	$X(j\omega) = \int_{-\infty}^{+\infty} x(t) e^{-j\omega t} dt$ <p>频域是连续非周期的</p>
时域 ↕ 频域	$\tilde{x}(n) = \sum_{k=(N)} \tilde{a}_k e^{jk(2\pi/N)n}$ <p>时域是离散周期的</p>	$x(n) = \frac{1}{2\pi} \int_{2\pi} X(e^{j\omega}) e^{j\omega n} d\omega$ <p>时域是离散非周期的</p>
	$\tilde{a}_k = \frac{1}{N} \sum_{n=(N)} \tilde{x}(n) e^{-jk(2\pi/N)n}$ <p>频域是离散周期的</p>	$X(e^{j\omega}) = \sum_{n=-\infty}^{+\infty} x(n) e^{-j\omega n}$ <p>频域是连续周期的</p>

红框内为数字信号处理的内容

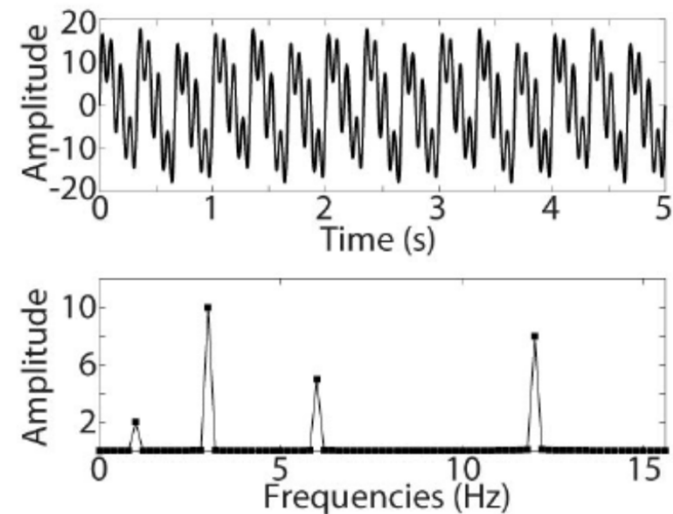
采样率、采样时间、频率分辨率

- 采样率 F_s , 采样间隔: $t_{\text{resol}} = 1/F_s$ (sec)
- 信号处理时间窗口长度 T (sec)
 - 也称为信号分析的时间窗口宽度——时间分辨率
- 时域数据长度 $N_t = T * F_s$;
- 频域数据长度 $N_f = N_t$
- 频域分辨率 $f_{\text{resol}} = F_s/N_f = F_s/N_t = F_s/(T * F_s) = 1/T$ (Hz)

多个正弦信号的傅里叶变换编程实践

- 产生多个正弦信号

```
sampRate = 1000;  
tTick = 0:1/sampRate:5; % seconds  
sLen = length(tTick);  
amp = [10 2 5 8];  
freq = [3 1 6 12]; % Hz  
sWave = zeros(size(tTick));  
for i=1:length(amp)  
    sWave = sWave + amp(i)*sin(2*pi*freq(i)*tTick);  
end
```

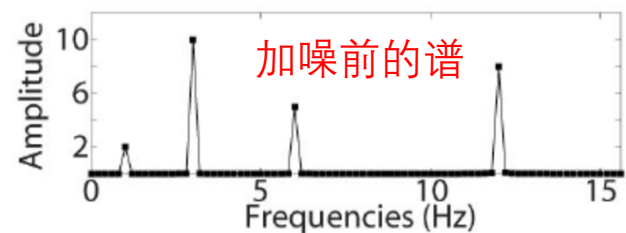
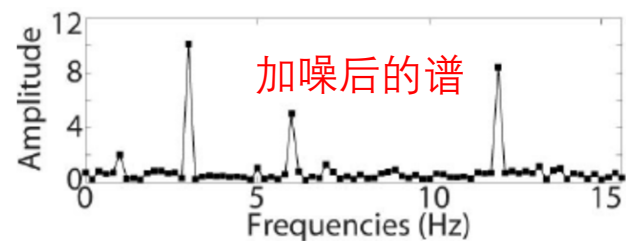
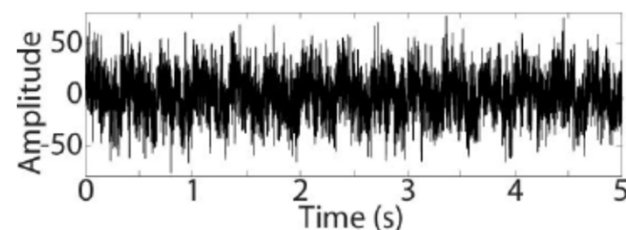


- 傅里叶变换

```
fftWave = fft(sWave)/sLen;  
fTick = [0: sLen-1]/ sLen*sampRate;  
figure; plot(fTick, abs(fftWave), '.-'); xlim([0 max(freq)*2]);
```

多个正弦信号和噪声

- 编程实践
 - 在前述信号上加入高斯白噪声
- ```
sWaveN = sWave + randn(size(swave))*20;
```
- 思考问题
    - 观察第一幅图，鉴别加噪的信号在时域内的频率成分
    - 观察第二幅图，鉴别信号内的频率成分
    - 为什么不同的域内观察频率成分的难易度不同？



## 练习题：采样时长与频谱分辨率

---

- 两个正弦信号，频率分别是10 Hz和10.5 Hz，采样率 $F_s$ 为200 Hz。请问最少需要采集多长时间的数据才能获得这两个信号的独立的傅里叶谱线。
  - (1) 根据上一页PPT内容进行推导数据的长度（时间）；
  - (2) 写计算机程序进行验证。

提示：为了区分两个频率，对频谱的频率分辨率有最低需求，从而对数据的时间长度有基本要求

# 频率分辨率与补零操作

---

- 频率分辨率

- $f_{\text{resol}} = F_s/N_f = F_s/N_t = F_s/(T \cdot F_s) = 1/T \text{ (Hz)}$

- 增加 $N_t$ ，从而提高频率分辨率？

- 在实测数据 $D$ （长度为 $N_d$ ）之后补 $k$ 个零

- $f_{\text{resol\_Nd}} = F_s/N_d$

- $f_{\text{resol\_new}} = F_s/(N_d+k)$

- 思考：是否可以无限补零，从而无限提高频率分辨率？

## 练习题：信号补零与频谱分辨率

---

- 仿真一个振幅为2，频率为10 Hz的正弦波，叠加一个振幅为0.5，频率为3 Hz的正弦波，采样频率为100 Hz
  - 分别计算采样时间为100毫秒，1秒，10秒的傅里叶谱
  - 观察三种条件下的谱线
  - 在100毫秒，1秒的采样时间条件获得数据后补零，获得2秒的总时长，重新计算傅里叶谱，并且对照观察补零的效果
- 结论
  - 补零可以提高频率分辨率，但是如果原始数据过短，无法保证低频谱线能否体现原始信号内的真实低频成分

# 频谱泄漏与窗函数

---

- 应用中，有限时长的数据记录，是对无限长信号的加一个矩形窗的结果（窗口宽度为 $T$ ， $t_0 \sim t_0 + T$ ）
  - 数学上，信号对应的函数 $x(t)$ 与矩形时间窗口 $w(t)$ 的乘积
- 时域内的乘积，对应频域的卷积
  - 时域加窗，等价于信号的傅里叶谱 $X(f)$ 与窗口函数的傅里叶谱 $W(f)$ 的卷积。我们知道有限时间窗口的傅里叶谱，结果为每一根实际信号的傅里叶谱线与窗口函数傅里叶谱的卷积，出现向邻近的频率范围内泄漏的情况
  - 频谱泄漏的程度与窗函数的傅里叶谱分布有关
  - 矩形窗函数傅里叶谱的旁瓣较宽较大，频谱泄漏严重
  - 不同窗口的性能见下页



# 典型窗口的谱旁瓣宽度与强度

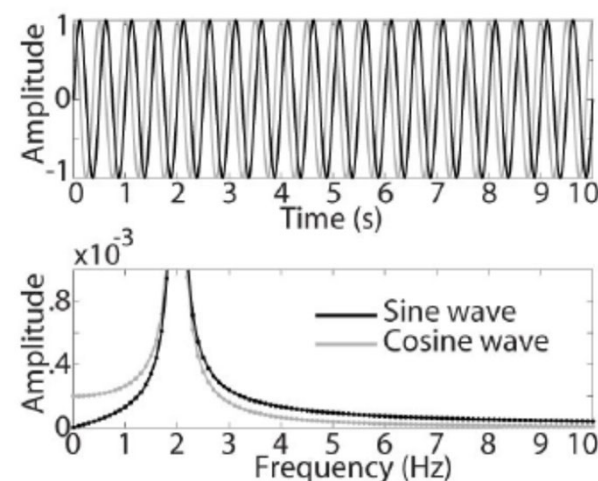
| Spectral Window                                                                                                                                                                      | Main Lobe Width | Side Lobe Level |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-----------------|
| $: N,$                                                                                                                                                                               |                 |                 |
| <b>RECTANGULAR</b>                                                                                                                                                                   |                 |                 |
| $D_R(f) = T \frac{\sin(\pi fNT)}{\sin(\pi fT)} e^{-j\pi fT(N-1)}$                                                                                                                    | $\frac{2}{NT}$  | 22.4%           |
| <b>TRIANGULAR-BARTLETT</b>                                                                                                                                                           |                 |                 |
| $D_B(f) = \frac{2T}{N} \left( \frac{\sin(\pi fNT/2)}{\sin(\pi fT)} \right)^2 e^{-j\pi fT(N-1)}$                                                                                      | $\frac{4}{NT}$  | 4.5%            |
| <b>HAMMING</b>                                                                                                                                                                       |                 |                 |
| $D_H(f) = 0.54D_R(f) + 0.23D_R\left(f + \frac{1}{NT}\right) + 0.23D_R\left(f - \frac{1}{NT}\right)$                                                                                  | $\frac{4}{NT}$  | 0.9%            |
| <b>BLACKMAN</b>                                                                                                                                                                      |                 |                 |
| $D_{BL}(f) = 0.42D_R(f) + 0.25D_R\left(f + \frac{1}{NT}\right) + 0.25D_R\left(f - \frac{1}{NT}\right) + 0.04D_R\left(f + \frac{2}{NT}\right) + 0.04D_R\left(f - \frac{2}{NT}\right)$ | $\frac{6}{NT}$  | 0.3%            |
| <b>PARZEN</b>                                                                                                                                                                        |                 |                 |
| $D_P(f) = \frac{64T}{N^3} \left( \frac{3}{2} \frac{\sin^4(\pi fNT/4)}{\sin^4 \pi fT} - \frac{\sin^4(\pi fNT/4)}{\sin^2(\pi fT)} \right) \cdot e^{-j\pi fT(N-1)}$                     | $\frac{6}{NT}$  | 0.22%           |

# 傅里叶变换的边缘效应 (矩形窗泄漏)

- 编程

```
srate=1000; t=0:1/srate:10; n=length(t);
x1 = sin(2*pi*2*t + pi/2);
x2 = sin(2*pi*2*t);
subplot(211), plot(t,x1)
hold on, plot(t,x2,'r')
xlabel('Time'), ylabel('amplitude')
hz = linspace(0,srate/2,floor(n/2)+1);
x1X = fft(x1)/n;
x2X = fft(x2)/n;
subplot(212),
plot(hz,2*abs(x1X(1:length(hz))), 'b.-'), hold on
plot(hz,2*abs(x2X(1:length(hz))), 'r.-')
xlabel('Frequency (Hz)'), ylabel('amplitude')
set(gca,'xlim',[0 10], 'ylim',[0 .001])
```

- 解释



# 信号加窗改善频谱泄漏

- 加窗的频域效应

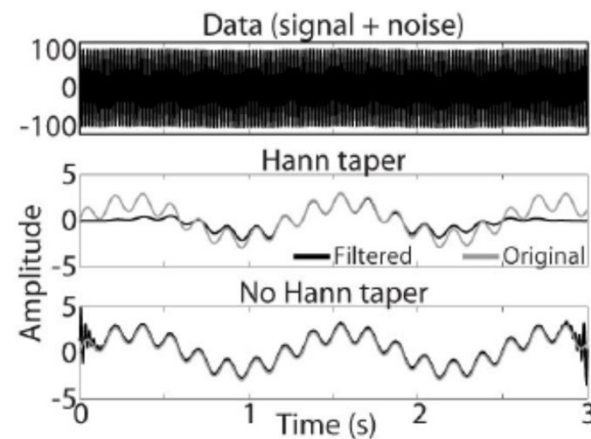
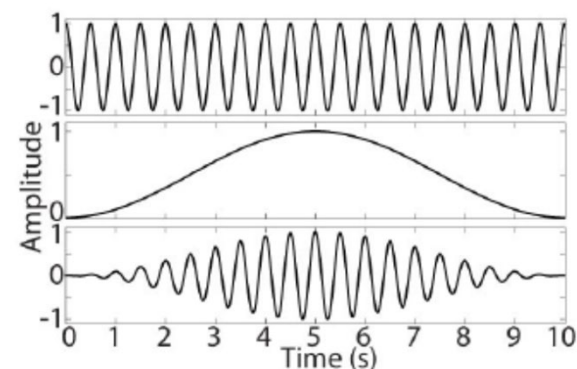
```
hannwin = .5*(1-cos(2*pi*linspace(0,1,n)));
subplot(311), plot(t,x1)
subplot(312), plot(t,hannwin)
subplot(313), plot(t,x1.*hannwin)
```

- 加窗对信号重建的影响

- 反傅里叶变换结果

- 加窗的应用原则

- 频率成分分析
  - 加窗减少频率泄露，精确定位谱线
- 信号重建
  - 不加窗，保持信号能量
  - 但是要注意信号边缘的失真



# 应用要点

---

- 记录长度选择 – 时间和频率分辨率的权衡
  - 因为傅里叶分析失去时间信息，因此时间分辨率等于数据记录的时间长度
- 补零法增加频率分辨率 – 注意对低频成分的影响
  - 原始信号需要足够长，才能在频域如实体现低频信号成分，否则补零也无法补救
- 加窗法减少频谱泄漏 – 但是注意加窗对能量的损失
  - 精确谱分析 – 加窗
  - 信号滤波与重构 – 不加窗

# 随机信号的谱分析

# 讨论要点

---

- 理解随机信号与确定信号谱分析的关键区别
  - 谱估计的概念
- 了解随机信号功率谱估计的间接法和直接法原理
- 掌握直接法功率谱估计的改进方法
  - 平均周期图
  - Welch法
- 掌握功率谱估计的应用要点
  - 频率分辨率, 窗口形状, 性能改进

# 随机信号的谱估计的概念

---

- 确定信号  $x(t)$  唯一确定，谱也唯一确定

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt \quad \text{and} \quad x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df$$

- 对于随机信号，我们仅能获得样本函数  $x(t)$ ，样本函数是一个随机函数，不是唯一确定，因此  $X(f)$  也是一个随机函数
  - 样本函数  $x(t)$  的傅里叶变换  $X(f)$  仅是随机信号谱的一个样本函数
  - 需要获得多个样本函数才能估计随机信号的谱
- 时域内，通过一系列的样本函数  $x(t)$ ，可以估计随机信号的矩函数
- 频域内，通过获得一系列样本函数的谱  $X(f)$ ，可以估计随机信号的谱 —— 谱估计

# 随机信号谱分析间接法

---

- 随机信号
  - 无限时长，非周期性，非平方可积
    - 不能离散傅里叶变换，因为周期无限长，对应连续谱
- 样本函数
  - 对一段样本进行傅里叶分析
  - 但单一样本不能代表总集
- 间接法
  - 找到用样本描述总集的统计量
  - 如果该统计量满足傅里叶变换的条件
  - 对该统计量进行傅里叶分析



# 随机信号的自相关函数特性

---

- 平稳遍历的随机信号 $X(t)$ 的自相关函数

$$R_{xx}(\tau) = \int_{-\infty}^{\infty} X(t)X^*(t + \tau) dt$$

- 在 $\tau=0$ 处获得最大值
- 常常是有限时长信号，随着 $\tau$ 增大，迅速衰减
- 自相关函数是有限能量信号，可以做傅里叶变换
- 进行周期延拓，成为周期信号，对应离散的频谱

- 实际应用时可从单一样本估计自相关函数

$$R_{xx}(m) = \lim_{N \rightarrow \infty} \frac{1}{2N + 1} \sum_{n=-N}^N x(n)x^*(n + m)$$

# 维纳-钦欣定理

---

- 对于零均值平稳遍历随机信号
- 总集意义上的功率谱

$$S_x(\omega) = \lim_{T \rightarrow \infty} E \left\{ \frac{|X_T(\omega)|^2}{2T} \right\}$$

- 可以写成维纳-辛钦定理形式，表示成自相关函数的傅里叶变换

- 反变换形式  $S_x(\omega) = \int_{-\infty}^{\infty} R_x(\tau) e^{-j\omega\tau} d\tau$

- 功率谱估计转换  $R_x(\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S_x(\omega) e^{+j\omega\tau} d\omega$  的估计

## 间接法功率谱估计的步骤

---

- 估计随机样本函数的自相关函数

$$R_{xx}(m) = \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N x(n)x^*(n+m)$$

- 根据维纳-辛钦定理，计算自相关函数的傅里叶变换获得功率谱

$$P_{xx}(e^{j\omega}) = \sum_{m=-\infty}^{+\infty} R_{xx}(m)e^{j\omega m}$$

- 间接法：先估计自相关函数，然后对自相关函数进行傅里叶变换获得功率谱

# 直接法功率谱估计

---

- 根据自相关函数的傅里叶变换获得的功率谱

$$P_{xx}(e^{j\omega}) = \sum_{m=-\infty}^{+\infty} R_{xx}(m)e^{j\omega m}$$

- 上式代入自相关函数 $R_{xx}(m)$ 变形为

$$P_{xx}(e^{j\omega}) = \lim_{N \rightarrow \infty} E\left(\frac{1}{2N+1} \left| \sum_{n=-N}^N x(n)e^{j\omega n} \right|^2\right)$$

- 直接法：对样本函数直接进行傅里叶变换，对获得的各频率系数平方求平均以获得功率谱

# 直接法功率谱估计的改进

---

- 直接法也叫周期图法
- 存在的问题：直接法功率谱估计不是一致估计
  - 即功率谱估计的方差 $\sigma^2$ 不会随着样本量增大而减小
- 改进方法 —— 平均周期图法（Bartlett法）
  - 原理：方差为 $\sigma^2$ 的独立高斯随机变量的K个样本的均值的方差等于 $\sigma^2/K$ 
    - 与时域的叠加平均算法环节讨论的原理一致
  - 随着数据量增加，样本个数K增加，方差减小
    - 即改进的方案可以获得一致性估计

# 平均周期图算法

---

- 将长度为N点的样本函数分为K段，每段长度为M
  - $M = N/K$
- 计算每一段M点数据的功率谱 $P_i(m)$
- 对K段周期图取平均  $P(m) = \frac{1}{K} \sum_{i=1}^K P_i(m)$ 
  - 优点：改善了方差特性—— $\sigma^2/K$
  - 缺点：每一段的数据长度M远远小于N——更短的窗口，更大的频谱泄漏
- 频谱泄漏改进方法——加窗
  - 见前面对确定性信号环节的讨论
- 功率谱的置信区间（不做要求）

# 平均周期图法M与K值选取

---

- 将N点数据分为K段，每段长度为  $M = N/K$
- K段间无重叠，如果每段独立，则估计均值不变，方差减少K倍，即  $\sigma^2/K$ 
  - K越大，方差越小，但是M 越小，估计偏差越大
- K和M的取值的相互妥协
  - M需要满足频率分辨率的需求
  - 在此基础上选择K
  - 对于有窄峰的谱，M值的选择非常重要
    - 为什么？

## 练习题：分段长度M的选取

---

- 长度为8192个数据点的脑电信号，采样率为 $F_s = 256\text{Hz}$ ，感兴趣的两个频率分量为7.75Hz和8Hz。用平均周期图法估计该段信号的功率谱，比较感兴趣的两个频率分量的相对功率，讨论最大的方差改进程度。

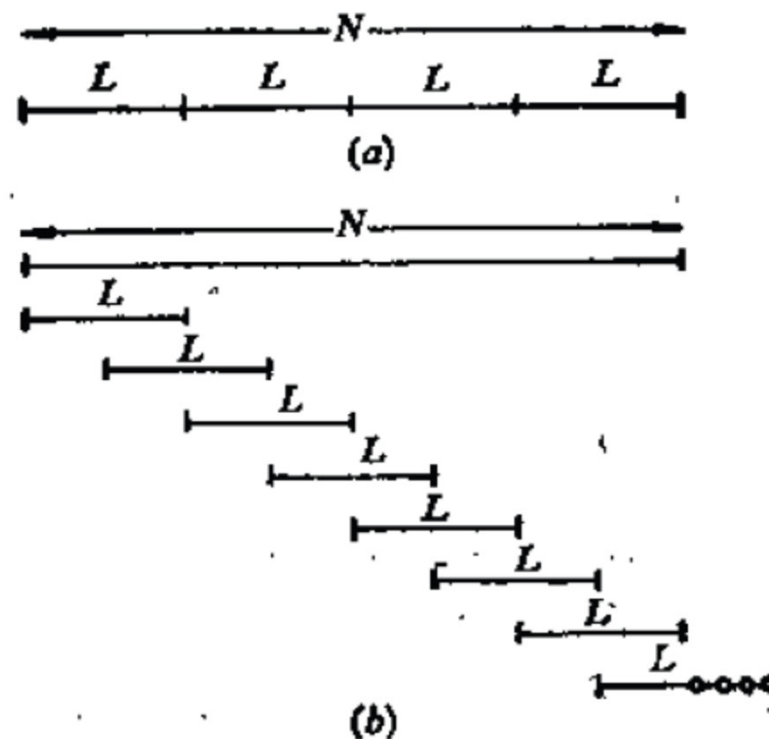
解：为比较两个分量，频率分辨率至少为0.25 Hz

- 信号长度至少为  $T_{\min} = 1/0.25 = 4\text{秒}$ ，则  $M_{\min} = T * F_s = 1024$
- $K = N/M = 8192/1024 = 8$
- 最多可以分成8段，方差的最大改善度为原来方差的1/8



# 功率谱估计改进的Welch法

- 应用更广泛
- 实质：平均法
  - 分段：  $K = N/L$
  - 每段可以重叠
    - 从而获得更大段数  $K$
  - 每段数据时域加窗
  - 每段数据傅里叶变换
  - 求所有分段功率谱的均值
- 有特定matlab函数
  - 练习 `pwelch()` 函数的用法

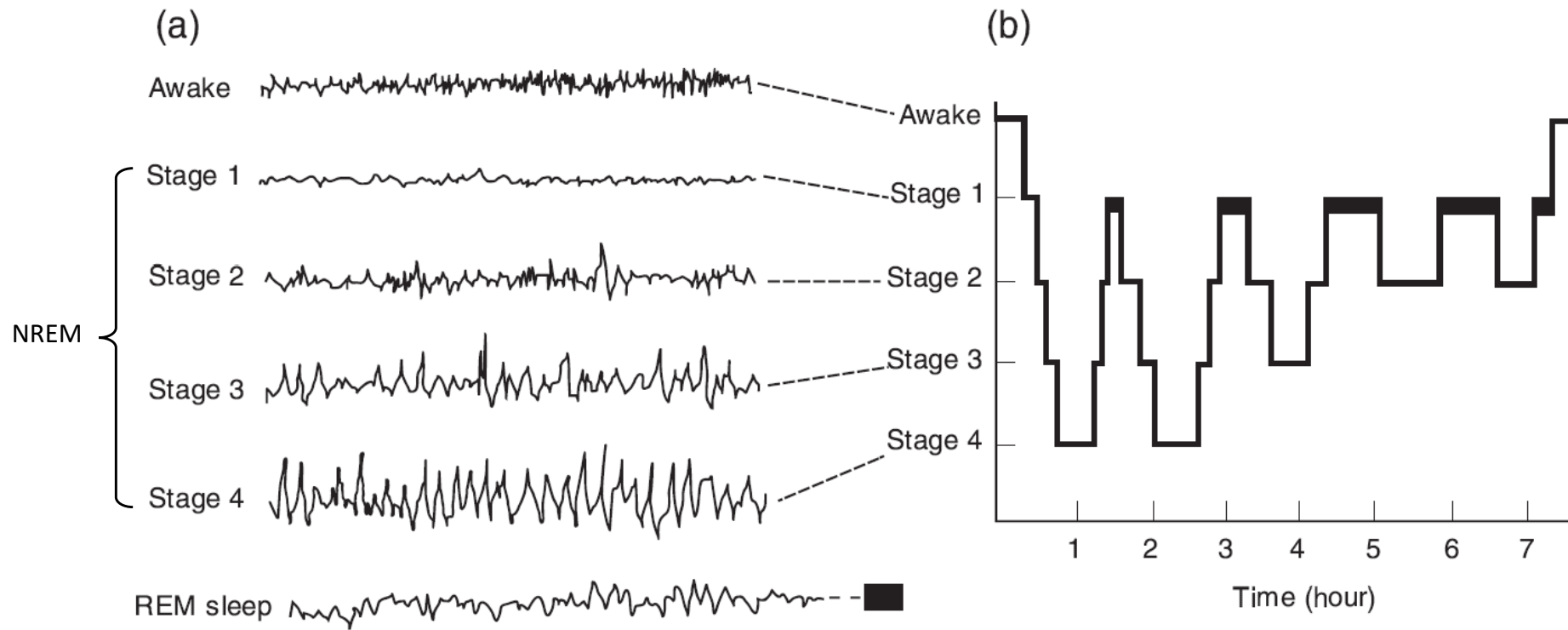


# 直接法功率谱估计的要点

- 数据采样频率 $F_s$ 
  - 首先要了解被测信号的频率带，选取合适采样率进行数据采集
  - 信号数字化之前使用抗混叠滤波器
- 数据预处理
  - 上述分析的基本假设：被分析的随机信号具有平稳遍历特性
  - 去直流和某些特定周期分量，从而减少delta函数性质谱峰，避免这些谱峰频谱泄露造成附近频率的失真
  - 用线性或者多项式拟合的方法去趋势项，减少趋势项造成的失真
- 每段数据长度 $M$ 的选择
  - 满足频率分辨率要求，可以采用适当加零点的方式提高谱线密度
    - 但是需要满足基本的频率分辨率要求的数据长度和不损失信号高频成分的采样率
    - 补零不能弥补原始采样数据不足造成的信息丢失
  - 数据也不可以太长，太长的数据更加容易违反随机信号“平稳性”的假设
    - 例如有同学计算252500个数据点的功率谱（采样率为500赫兹），即505秒

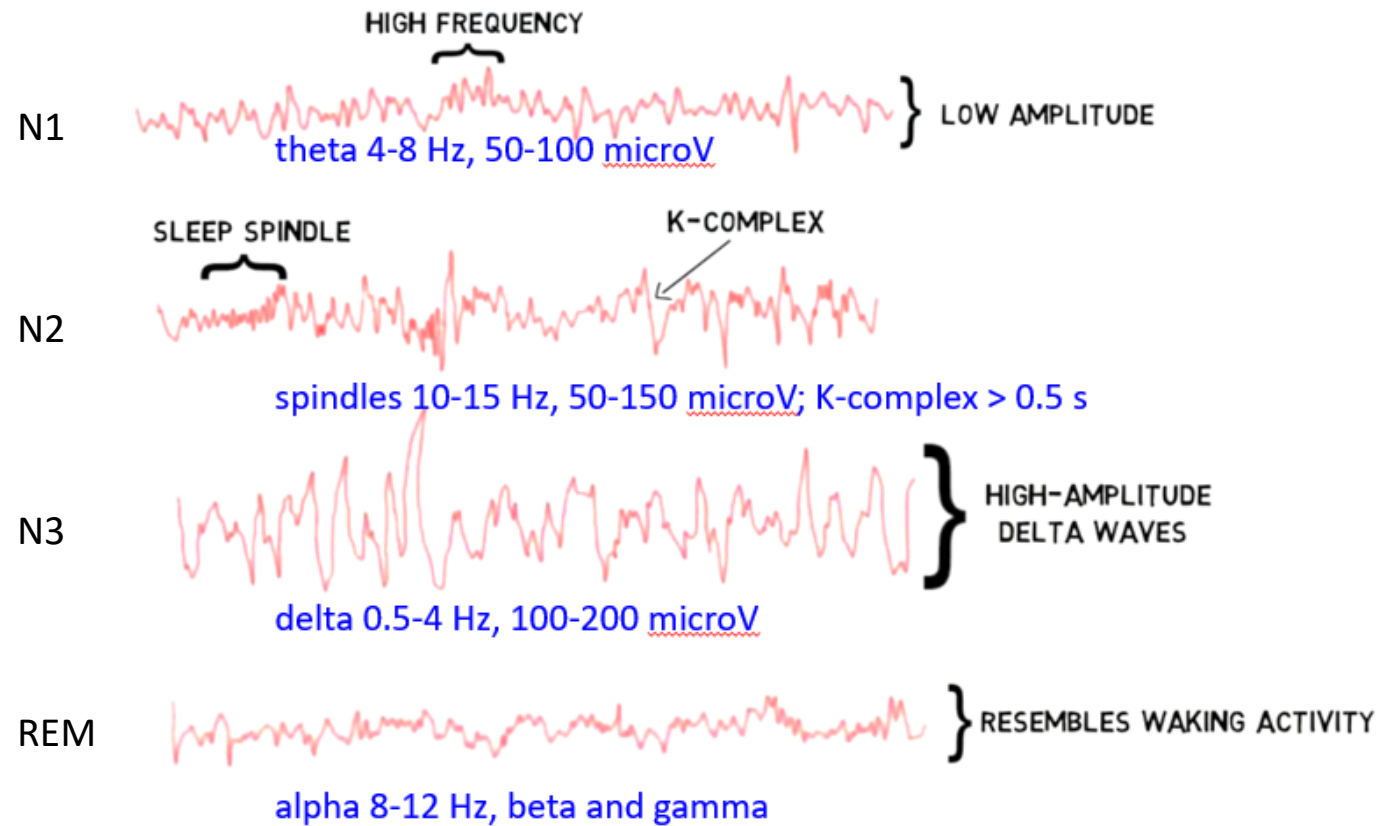
# 功率谱分析在睡眠分期的应用

# 睡眠分期



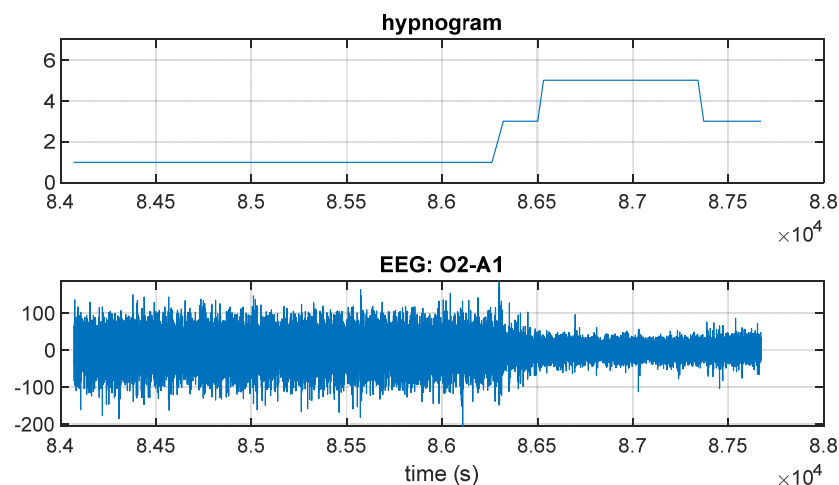
McNamar (2019) 睡眠可以分成多个阶段，各阶段的具有典型的神经振荡特征，可用脑电进行测量和分析

# 脑电特性与睡眠分期



## 应用案例——基于脑电的睡眠分期

- 现给定数据文件 `n7_iBlk2_O1A2_hyp.mat`。请用Welch法估计这两段信号的功率谱，对照上图对波形特征的介绍，解释这段数据对应的睡眠分期。



# 可视化原始数据

---

- `clear all;`
- `load('n7_iBlk2_O1A2_hyp.mat');`
- 
- `% hypHeight = 1 for S4, 2 for S3, 3 for S2, 4 for S1, 5 for REM, and 6 for WAKE`
- `% to align hypnogram with EEG data, note every data point in hypnogram represents 30 seconds`
- 
- `%% visualize the data`
- `figure;`
- `ax(1) = subplot(2, 1, 1); plot(tTick_hypnogram_iBlk, hypHeight_iBlk);`
- `title('hypnogram'); ylim([0 7]); grid on;`
- `% tTick_hypnogram_iBlk(1)/3600 = 23.3531 -- 23.3531 hour`
- 
- `ax(2) = subplot(2, 1, 2); plot(tTick_hypnogram_iBlk(1) + [0:length(chData_iBlk)-1]/sampRate,`  
`...`
- `chData_iBlk); title(['EEG: ', chLabel]);`
- `xlabel('time (s)');`
- `linkaxes(ax, 'x'); grid on;`

# 根据睡眠分期分割脑电数据并计算功率谱

```
• %% find the EEG data segment for S4 and the segment for REM
• % and calculate PSD using pwelch with parameters:
• winSec = 2;
• noverlap = winSec*0.5*sampRate;
• npFFT = winSec*sampRate;
•
• idx_S4 = find(hypHeight_iBlk == 1); % for S4
• idx_REM = find(hypHeight_iBlk == 5); % for REM
•
• for iS4 = 1:length(idx_S4)
• eeg30sData = chData_iBlk((idx_S4(iS4)-1)*30*sampRate+1 ...
• :idx_S4(iS4)*30*sampRate);
• [eegPSD, fTick, psdCI] = pwelch(eeg30sData, winSec*sampRate, noverlap, npFFT, sampRate, ...
• 'ConfidenceLevel', 0.95);
• eegS4(iS4).eegData = eeg30sData;
• eegS4(iS4).eegPSD = eegPSD;
• end
•
•
```



# REM分期的数据和功率谱

---

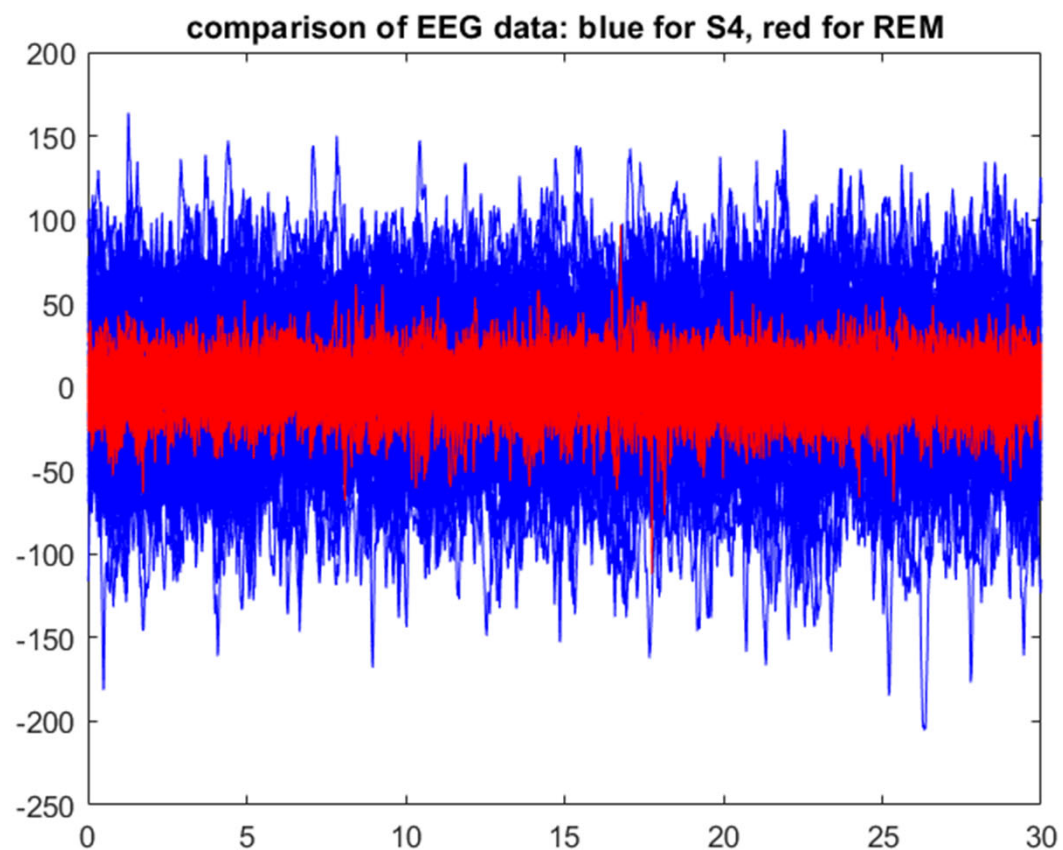
- `for iREM = 1:length(idx_REM)`
- `eeg30sData = chData_iBlk((idx_REM(iREM) - 1)*30*sampRate+1 ...`
- `:idx_REM(iREM)*30*sampRate);`
- `[eegPSD, fTick, psdCI] = pwelch(eeg30sData, winSec*sampRate, noverlap, npFFT, sampRate, ...`
- `'ConfidenceLevel', 0.95);`
- `eegREM(iREM).eegData = eeg30sData;`
- `eegREM(iREM).eegPSD = eegPSD;`
- `end`

# 可视化S4和REM分期的脑电数据

---

```
• %% visualization for S4 and REM data
•
• % in time domain
•
• figure;
• for iS4 = 1:length(idx_S4)
• plot([1:length(eegS4(iS4).eegData)]/sampRate, eegS4(iS4).eegData, 'b-');
• hold on;
• end
• for iREM = 1:length(idx_REM)
• plot([1:length(eegREM(iREM).eegData)]/sampRate, eegREM(iREM).eegData, 'r-');
• hold on;
• end
• title('comparison of EEG data: blue for S4, red for REM');
```

# 原始数据可视化（30秒数据分段）

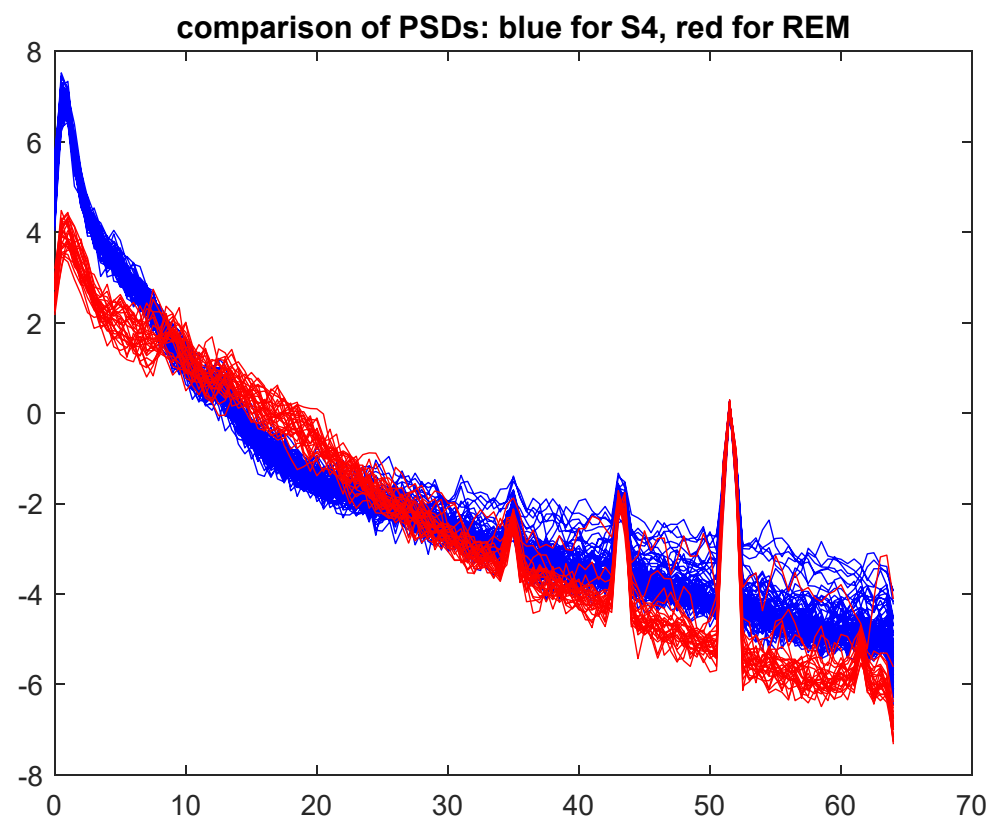


# 可视化功率谱

---

- % in Frequency domain
- figure;
- for iS4 = 1:length(idx\_S4)
- plot(fTick, log(eegS4(iS4).eegPSD), 'b-'); hold on;
- end
- for iREM = 1:length(idx\_REM)
- plot(fTick, log(eegREM(iREM).eegPSD), 'r-'); hold on;
- end
- title('comparison of PSDs: blue for S4, red for REM');

# S4和REM分期的功率谱



# 经典频率区间的功率

---

- % summarize the PSD in delta (0.5~4 Hz), theta (4.5-8 Hz), alpha (8.5~12 Hz),
- % beta (12.5~30 Hz)
- iCount = 0;
- for iS4 = 1:length(idx\_S4)
- iCount = iCount +1;
- bandPower(iCount) = mean(eegS4(iS4).eegPSD(2:8));
- group(iCount, :) = 'delta S4 ';
- dPowerS4(iS4) = bandPower(iCount);
- 
- iCount = iCount +1;
- bandPower(iCount) = mean(eegS4(iS4).eegPSD(9:16));
- group(iCount, :) = 'theta S4 ';
- tPowerS4(iS4) = bandPower(iCount);
-

# 经典频率区间的功率

---

- `iCount = iCount +1;`
- `bandPower(iCount) =  
mean(eegS4(iS4).eegPSD(17:24));`
- `group(iCount, :) = 'alpha S4';`
- `aPowerS4(iS4) = bandPower(iCount);`
- 
- `iCount = iCount +1;`
- `bandPower(iCount) =  
mean(eegS4(iS4).eegPSD(25:60));`
- `group(iCount, :) = 'beta S4';`
- `bPowerS4(iS4) = bandPower(iCount);`
- `end`

# 经典频率区间的功率

---

- `for iREM = 1:length(idx_REM)`
- `iCount = iCount +1;`
- `bandPower(iCount) = mean(eegREM(iREM).eegPSD(2:8));`
- `group(iCount, :) = 'delta REM';`
- `dPowerREM(iREM) = bandPower(iCount);`
- 
- `iCount = iCount +1;`
- `bandPower(iCount) = mean(eegREM(iREM).eegPSD(9:16));`
- `group(iCount, :) = 'theta REM';`
- `tPowerREM(iREM) = bandPower(iCount);`
-



# 经典频率区间的功率

---

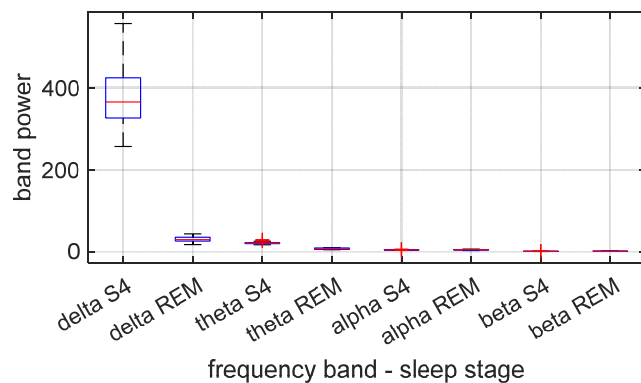
- `iCount = iCount +1;`
- `bandPower(iCount) =`  
`mean(eegREM(iREM).eegPSD(17:24));`
- `group(iCount, :) = 'alpha REM';`
- `aPowerREM(iREM) = bandPower(iCount);`
- 
- `iCount = iCount +1;`
- `bandPower(iCount) =`  
`mean(eegREM(iREM).eegPSD(25:60));`
- `group(iCount, :) = 'beta REM ';`
- `bPowerREM(iREM) = bandPower(iCount);`
- `end`

# 可视化各频率区间S4和REM分期的功率

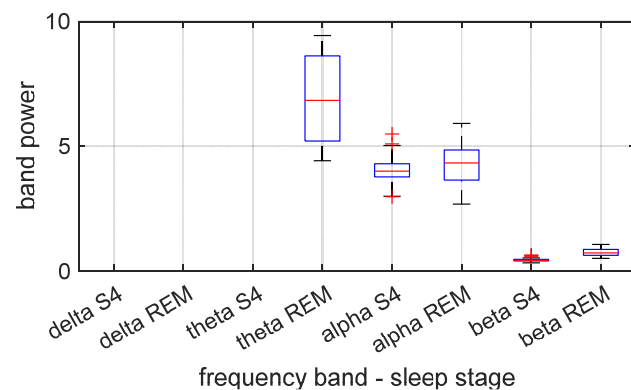
---

- `figure, boxplot(bandPower, group, 'grouporder', ...`
- `{'delta S4' 'delta REM' 'theta S4' 'theta REM' 'alpha S4' ...`
- `'alpha REM' 'beta S4' 'beta REM'} );`
- `grid on; ylabel('band power'); xlabel('frequency band - sleep stage');`
- `% ylim([0 10]); % 突出比较S4和REM的alpha和beta功率的大小`

# 各频率区间S4和REM分期的功率



全局图：最明显的特征是S4睡眠阶段的delta波功率最高，这是S4期最典型的脑电特征



局部图：REM的alpha和beta波比S4睡眠阶段更高

# 小结

---

- 确定信号的傅里叶变换的要点
  - 频率分辨率
  - 时间分辨率
  - 频谱泄漏与加窗
- 随机信号的功率谱估计
  - 功率谱的随机特性
  - 功率谱估计方法
  - 功率谱估计的方差改进
  - 频率分辨率与方差改进方法的矛盾
- 应用要点
  - 信号的平稳性、窗口长度、是否加窗等