

脑电数据预处理基本步骤（九步走）

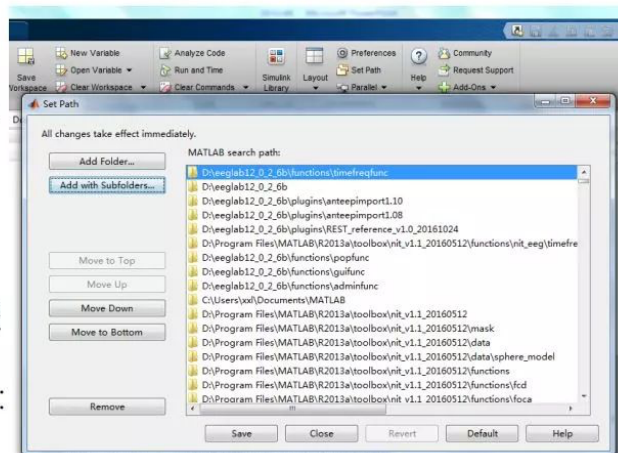
- 1.导入数据
- 2.定位电极
- 3.剔除无用电极
- 4.重参考（即重新选择参考电极，位置随意）
- 5.滤波
- 6.分段（可在此步骤保存数据）
- 7.插值坏导和剔除坏段
- 8.Run ICA（可在此步骤保存数据）
- 9.剔除眼动成分等（保存）。如：眨眼，眼瞟，明显的噪声（乱动）。

来源：彭薇薇老师

准备工作

启动EEGLAB

- 拷贝eeglab12_0_2_6b.zip.
也可以在EEGLAB的网站上下
载最新版本
- 解压缩
- 打开matlab
- Set path→Add with
subfolders, 然后选择已经
解压缩的文件夹
- 在matlab的任务窗口输入：
eeglab



解读：eeglab 是基于 MATLAB 的一个工具包，一般加载的时候都从添加子文件夹导入。在添加多个工具包，记得只保留必要的工具包，避免兼容报错的问题。

一、导入数据

步骤 1: File - Import data - 不同的数据格式不同的导入方法

解读: 其中 BP 设备和 ANT 设备的数据, 都是从.vhdr 中导入。

步骤 2: File - Load existing dataset - eeglab_data.set

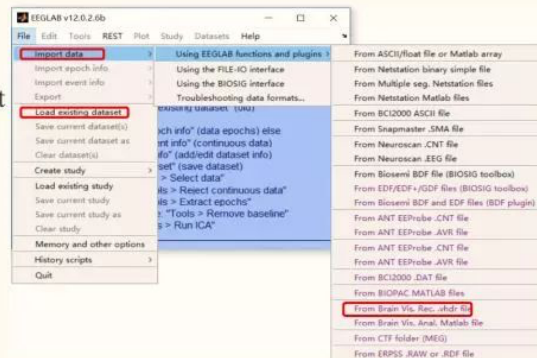
解读: 若是导入 eeglab 保存的数据, 则直接按照步骤 2 导入即可。

来源: AffectiveNeuroscience

Data import (时间视角)

➤File→Import data (不同的数据格式不同的导入方法)

➤File→Load existing dataset →eeglab_data.set



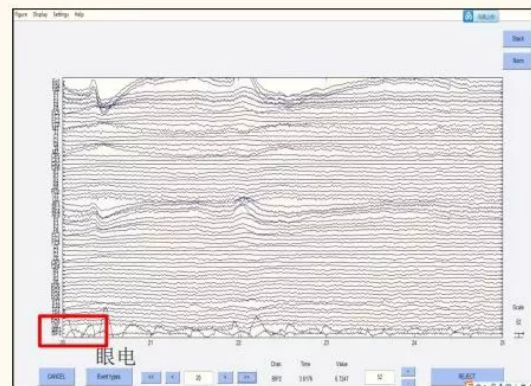
二、定位电极 (时间)

步骤: Plot - Channel data(scroll) - settings - time range to display -setting - number of channels to display value(调整幅度)

解读: 可以在数据分析之前, 浏览一下原始数据, 自己对数据的好坏有一个评估。

Channel data (scroll) (位置任意)

- Plot→Channel data (scroll)
- Settings→Time range to display
- Settings→Number of Channels to display
- Value (调整幅度)



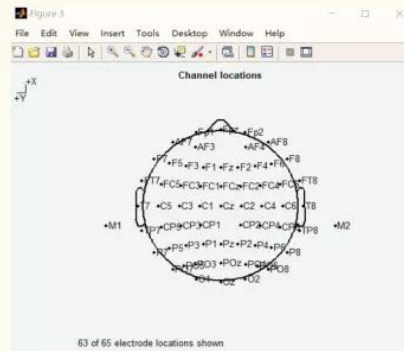
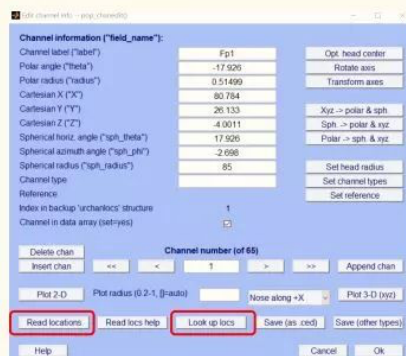
三、定位电极 (空间)

步骤：Edit - Channel locations - read locations (look up locs) - eeglab_chan64.locs(plot 2D)

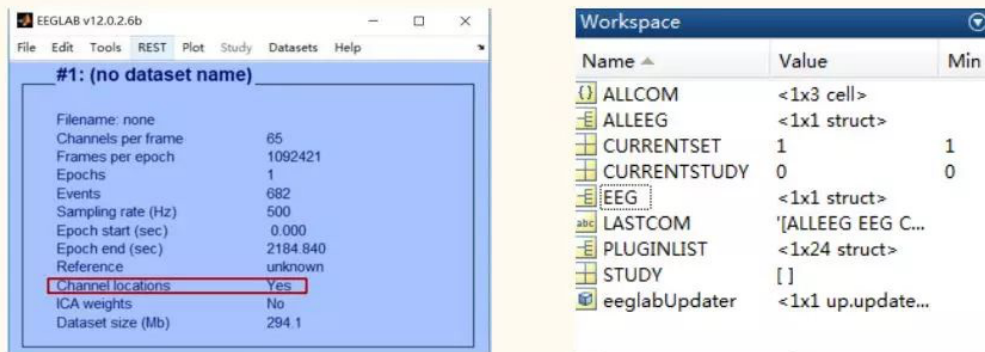
解读：在数据分析之前，查看电极点的分布图，方便后期进行使用插值法进行坏点替换。

Channel locations (空间视角)

- Edit→Channel locations→Read locations (Look up locs) →eeglab_chan64.locs (plot 2D)



界面变化



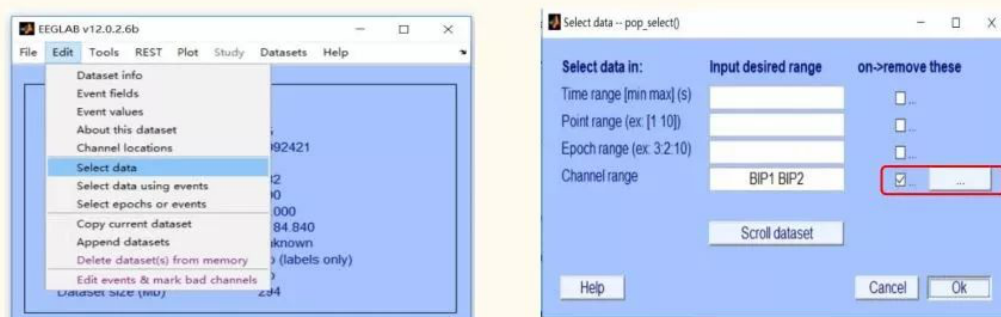
四、删除无用电极

步骤：Edit - Select data - 点上√，即删除选取的电极；不点√，则是删除剩余的电极。

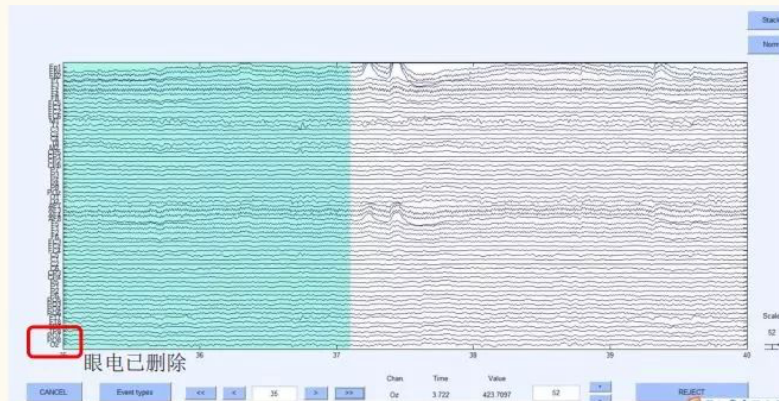
解读：删除记录多余的电极，只选取自己需要的电极。

Select data

➤ Edit→Select data（删除多余电极）



界面变化



四、重参考

步骤 1： Edit - Re-reference - re-reference data to channel(s) - 电极点 M1 M2 (TP9 TP10) （双侧乳突）

步骤 2： Edit - Re-reference - compute average reference （全脑平均）

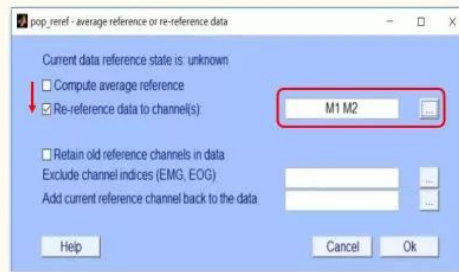
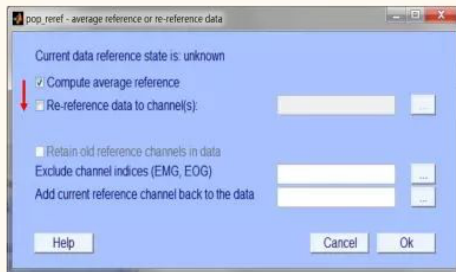
步骤 3： 使用姚德忠老师的 **re-reference** 插件（零参考）

解读： 重参考的方法常用的有双侧乳突、全脑平均、零参考，具体选取那种方法根据以往的参考文献和自己的需要来进行选择。

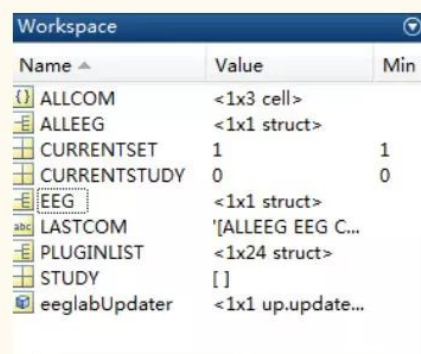
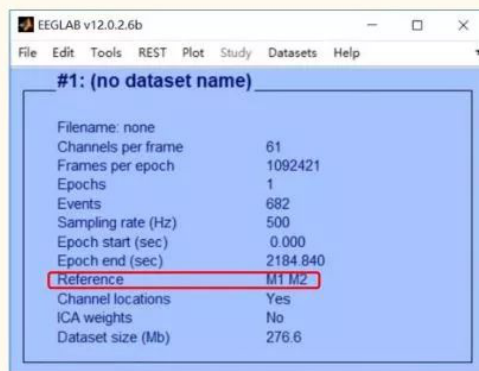
重参考也是一种空间滤波，主要是通过另外一个角度来看问题。所以不同的在线参考其实对于离线参考没有太大影响。

Re-reference

➤ Tools → Re-reference



界面变化



五、滤波

步骤 1: Tools - Filter the data - basic FIR filter - (1 Hz high pass filter first) - Overwrite it in memory(根据需要获取目标频段)

步骤 2: Tools - Filter the data - Basic FIR filter - (30 Hz low pass filter second) - Overwrite it in memory

解读: 高通滤波,是指高频信号能正常通过,而低于设定临界值的低频信号则被阻隔、减弱。因而是进行 1Hz 的高通滤波,而选择的时候,1Hz 是频率通过的下限。

低通滤波,是指低频信号能正常通过,而超过设定临界值的高频信号则被阻隔、减弱。因而是进行 30Hz 的低通滤波,而选择的时候,30Hz 是频率通过的上限。

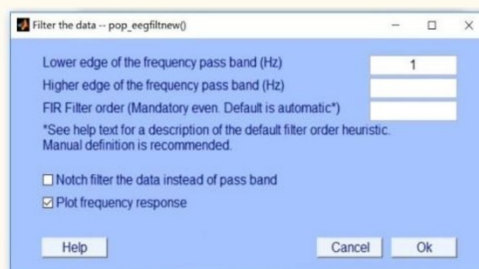
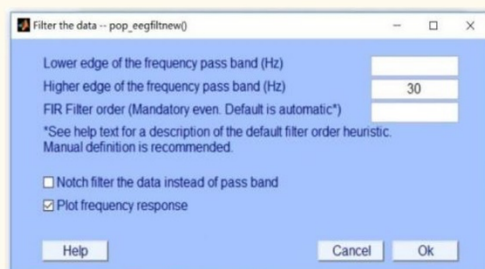
早期的 eeglab 版本,不能够同时滤波,容易卡死;最新版本的 eeglab 是可以同时进行空间滤波的。

若是后期要做时频分析,可以滤波的范围选择更宽一点,选择 0.1-100。若是只进行传统的 ERP 分析,可以选择 1-30 左右。

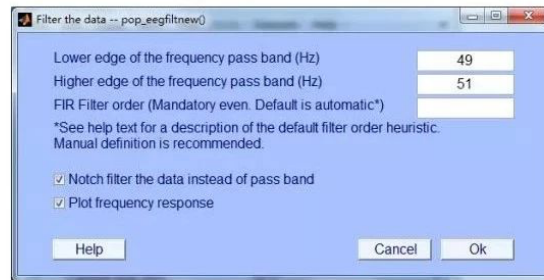
此外,若是进行 0.1-100Hz 的滤波,为了消除市电的干扰,可以进行 50Hz 的凹陷滤波。

Basic filter

- Tools→Filter the data→Basic FIR filter → (1 Hz high pass filter first)→Overwrite it in memory (根据需要获取目标频段)
- Tools→Filter the data→Basic FIR filter → (30Hz Low pass filter second)→Overwrite it in memory (同时滤波容易卡死,故分开滤波)



- 凹陷滤波（消除特定频率的干扰，如50Hz工频干扰）



六、分段和基线校正

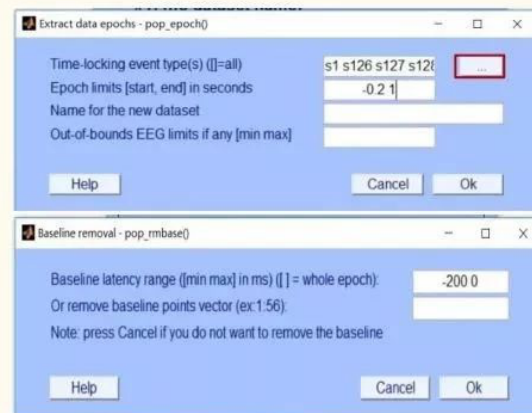
步骤： Tools - Extract epochs(分段 marker 全选) - Automatic baseline correction

解读：分段的步骤，可以在去除眼电之前，也可以在去除眼电之后。其实最好是在去除眼电之后，因为连续的数据在跑 ICA 时更好，只是数据量比较大，跑的速度比较慢。

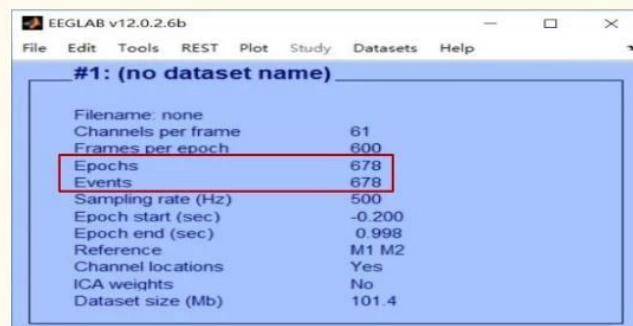
但是若是实验设计当中有出声、身体动，造成伪迹较多，数据杂乱，可以先分段，只是可以在分段的时候，尽量分段长一点。

Epoch and remove baseline

- Tools → Extract epochs (分段marker 全选)
- Automatic baseline correction



界面变化



七、伪迹去除

步骤 1: Plot - Channel data(scroll) (删除坏的 trial) - 差值坏导 (代码)

步骤 2: File - Save as - 保存文件

解读: 伪迹去除是去除眼电之外的其他杂乱的伪迹,使得在 Run ICA 之前的数据比较干净,容易找到眼电成分。

Artifact rejection

- Plot → Channel data (scroll) （删除坏的trial）
- 插值坏导（代码）：
`EEG.data(a,:,:) = mean(EEG.data([b c d e],:,:));`

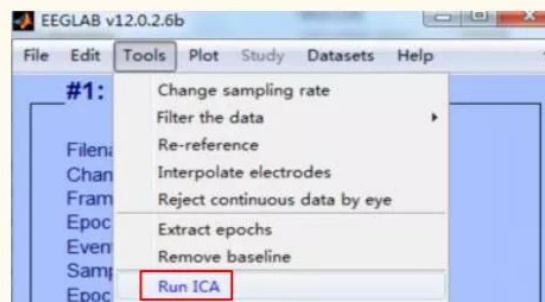
八、Run ICA

步骤：Tools - Run ICA - 'extended', 1 'pca', 30 - OK

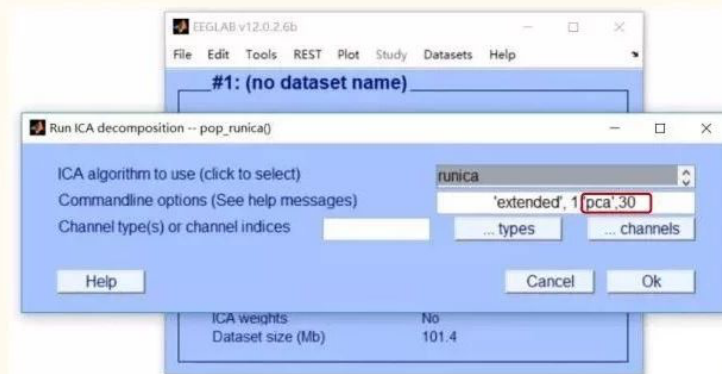
解读：Run ICA 的时候，可以写 30 个，也可以写 60 个主成分。去除的时候，需要去除自己最肯定的成分，一般都是去除眼电成分。

Independent component analysis

- 进一步去除脑电中的噪声（主要是眼电），可以多次使用。
- Tools → Run ICA

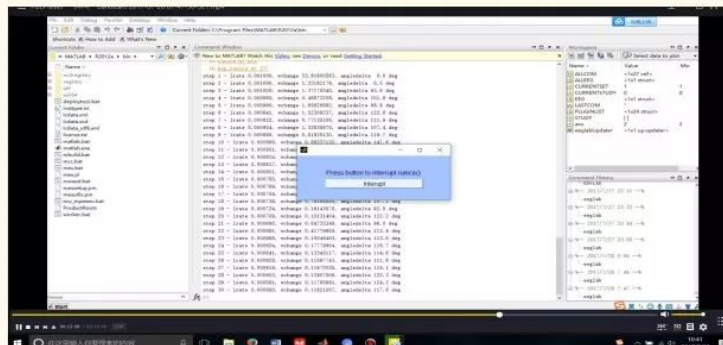


Independent component analysis



Independent component analysis

运行中...



九、眼电去除

步骤 1: Tools - Reject data using ICA - Reject component by map

步骤 2: Select each IC and observe

步骤 3: Tools - remove components - 填写删除电极的数字 - OK

解读：眼电判断的时候，除了看地形图，也可以看频率分布图、每个成分的波形图，以及矫正之后的脑电图是否有差异，进行综合的评估。

只有自己很确定的伪迹主成分才删除，不然可能会删除自己想要的成分。

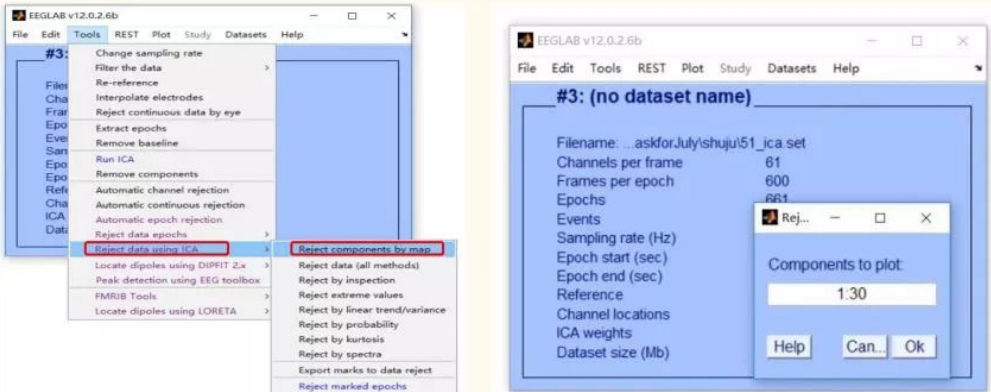
若是出来的成分，没有出现明显的伪迹成分，可以多跑几次 **ICA**；或者不分段重新跑一次；又或者再看看原始数据，删除杂乱的成分，重新再跑一次。

这一步是很需要经验的步骤，若是前期自己把握不准，可以让有经验的人帮忙看看。

通过 **Adjust 插件**，也可以进行一个简单的伪迹成分的评估，不过最终的确定还是需要自己来进行评估。

Independent component analysis

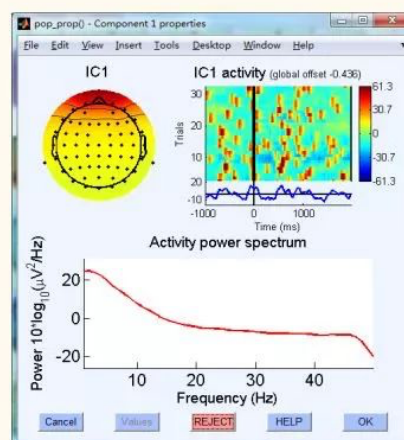
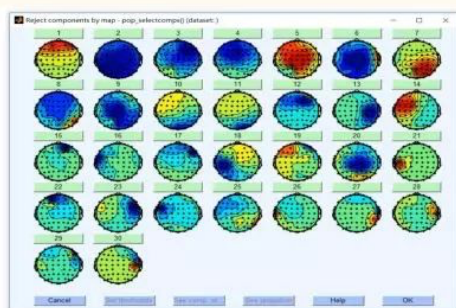
➤ Tools → Reject data using ICA → Reject component by map



The image displays two screenshots from the EEGLAB v12.0.2.6b software interface. The left screenshot shows the 'Tools' menu with 'Reject data using ICA' selected, and a sub-menu with 'Reject components by map' highlighted. The right screenshot shows the 'Reject components by map' dialog box with 'Components to plot' set to 1.30.

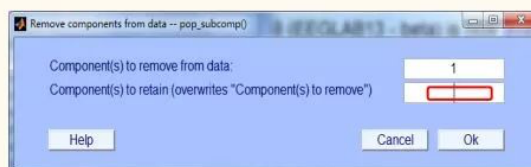
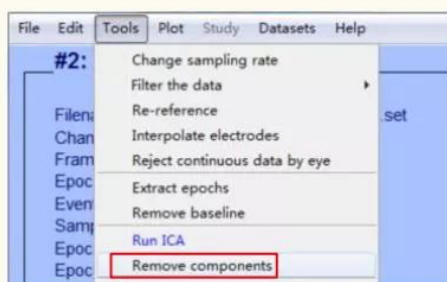
Independent component analysis

- Select each IC and observe



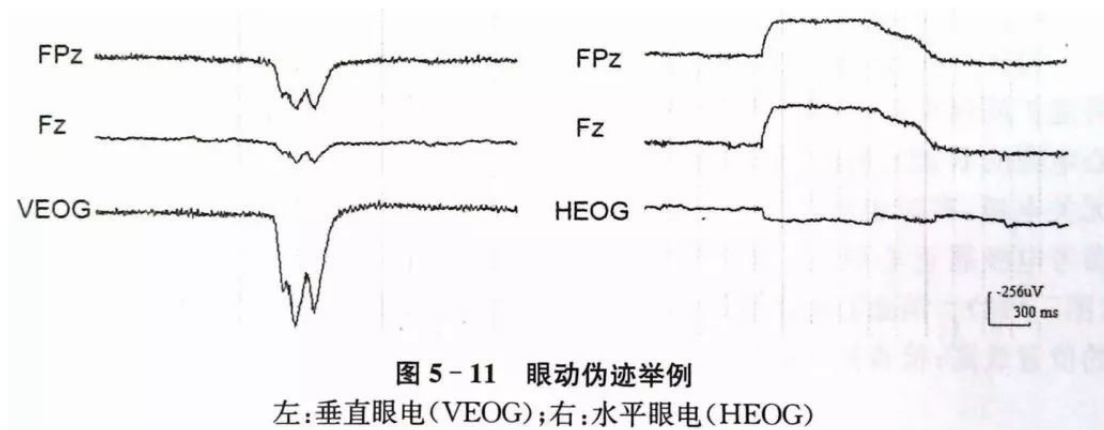
Independent component analysis

- Tools → Remove components



眼电去除的标准

来源：彭薇薇老师

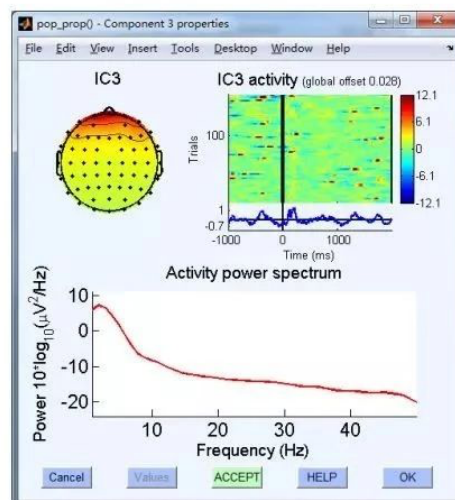


- 经典成分-眨眼

- 判断依据：

- 1 前端分布
- 2 小方块
- 3 随机分布
- 4 低频能量高
- 5 成分排序靠前

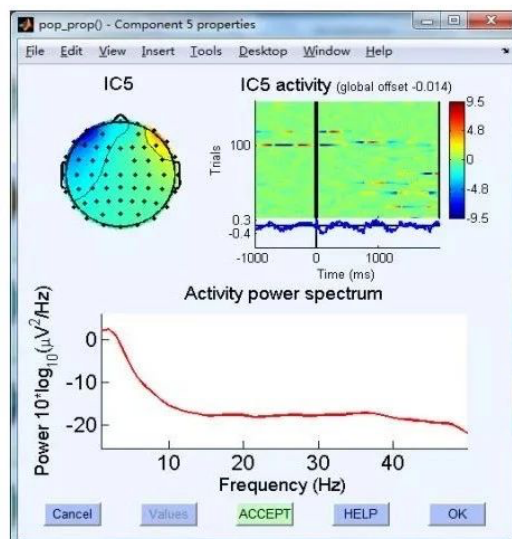
- 注意：颜色的深浅有意义，但是红蓝无所谓，大红大蓝都可以。



- 经典成分-眼漂

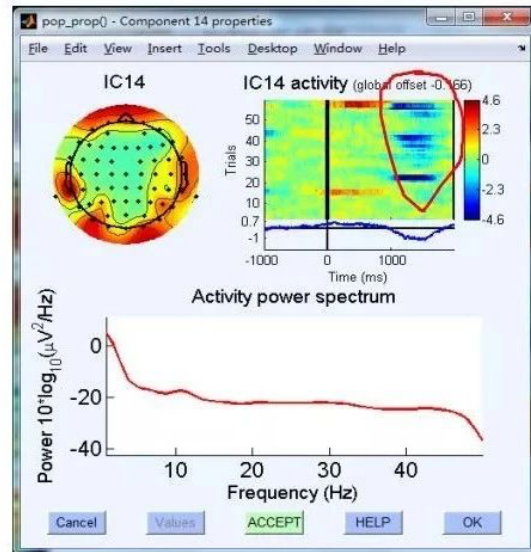
- 判断依据：

- 1 前端两侧分布，红蓝相对
- 2 长条状，红蓝相间
- 3 随机分布
- 4 低频能量高
- 5 成分排序靠前，但一般排在眨眼后面



- 经典成分：**头动**

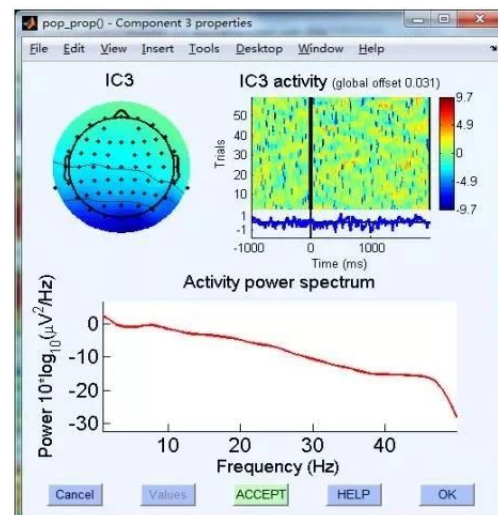
- 1 周围分布
- 2 长条状
- 3 随机分布
- 4 在单个trial里有非常明显的漂移
非常长的飘移



- 其他成分

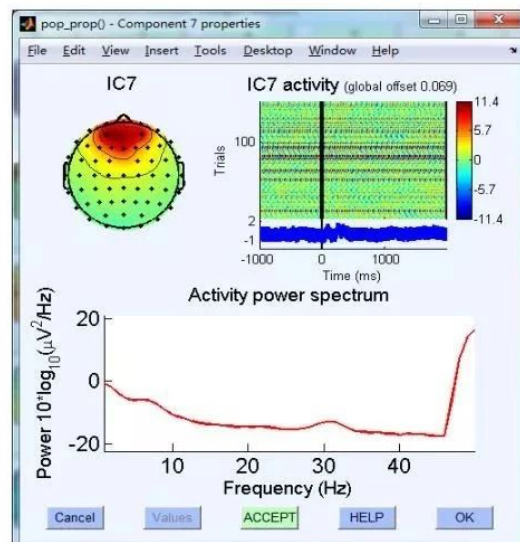
- **心电**

- 呈雨点般散落状。



- 其他成分：
工频干扰

- 1 分布在地线周围
- 2 单个Trial上的分布非常有规律
- 3 50Hz左右能量最高



十、预处理的批处理

步骤: eeg.history - 出现之前处理的代码 - 进行每个被试的批处理 - 然后处理完之后再手动去除眼电成分 - 保存为处理干净的脑电信号 - 进行下一步的分析

解读: 脑电数据的预处理是为了提高信噪比，去除噪音，得到比较干净的数据。从而进行下一步的分析。

Matlab scripts

```
• >> EEG.history (预处理)
• EEG = pop_loadbv('C:\Users\11125\Desktop\TaskforJuly\eeeg\',[num2str(i)'\vhdr'], [], []);
• EEG = eeg_checkset( EEG );
• EEG = pop_chanedit(EEG, 'lookup','C:\Users\11125\Desktop\TaskforJuly\eeeglab12_0_2_6b\plugins\dipfit2.2\standard_BESA\standard-10-5-cap385.elp');
• EEG = eeg_checkset( EEG );
• EEG = pop_select( EEG,'nochannel',{'BIP1' 'BIP2'});
• EEG = eeg_checkset( EEG );
• EEG = pop_reref( EEG, [13 19] );
• EEG = eeg_checkset( EEG );
• EEG = pop_eegfiltnew(EEG, [], 1, 1650, true, [], 1);
• EEG = eeg_checkset( EEG );
• EEG = pop_eegfiltnew(EEG, [], 30, 220, 0, [], 1);
• EEG = eeg_checkset( EEG );
• EEG = pop_epoch( EEG, { 's1' 's126' 's127' 's128' 's2' 's25' 's26' 's27' 's3' 's4' 's5' 's6' }, [-0.2 1], 'epochinfo', 'yes');
• EEG = eeg_checkset( EEG );
• EEG = pop_rmbase( EEG, [-200 0]);
• EEG = eeg_checkset( EEG );
• EEG = pop_saveset(EEG,'filename',[num2str(i)'\.set'],'filepath','C:\Users\11125\Desktop\TaskforJuly\eeeg\epoch\');
```