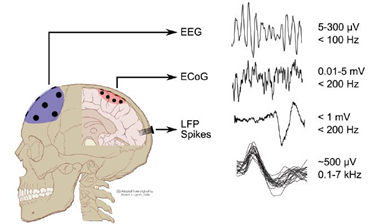
脑机接口中的数据处理

内容摘要：脑机接口数据处理包括数据采集、放大、滤波、数学转化，其中数学转化是数据处理的核心。人工智能常用数据数学处理方法主要有主成分分析PCA，非负矩阵分解NMF和分布随机邻域嵌入t-SNE等降维方法。

关键词：脑机接口、数据处理、降维。

一、脑电信号的采集

脑电信号由神经元化学电位变化传导产生，神经元是脑电信号的基本单位，分离每个神经元的电位变化是脑电信号分析的最终目标。大脑由约860亿个神经元，相互关系更为复杂，神经元活动时表现出的电信号有三种形式，分别对应侵入式电极、半侵入式电极与非侵入式电极的接受情况。



（图片来源：https://zhuanlan.zhihu.com/p/28390875）

1.侵入式：接收到局部场电位（LFP）。局部场电位由插入式金属电极采集，是位于同一区域的多个神经元动作电位合成的，参数为500μV、0.1-7kHz。可视化的结果是高度同步的动作电位图像直接叠加。这样的电信号具有同步化高、电位变化明显、受环境影响小的特点，在电脑分析的时候降低了解码难度。但由于侵入电极植入对人体创伤较大，成本较高，不是未来脑机接口普及化的主要方式。

2.半侵入式：接收到皮层脑电（ECoG）。皮层脑电由电极阵列在大脑皮层，颅骨之下提取，参数为0.01-5mV、<200Hz。可视化结果是较为无序的电位震动，是由于采集信号范围相对较大，神经元分布更广而关联性较弱，所以处理这类信号需要利用独立成分分析（ICA）分离不同神经元的电位信号。由于采集信号在颅骨之下，信号受环境影响也不大，手术创伤相对侵入式小一点，但依旧不能大范围推广。

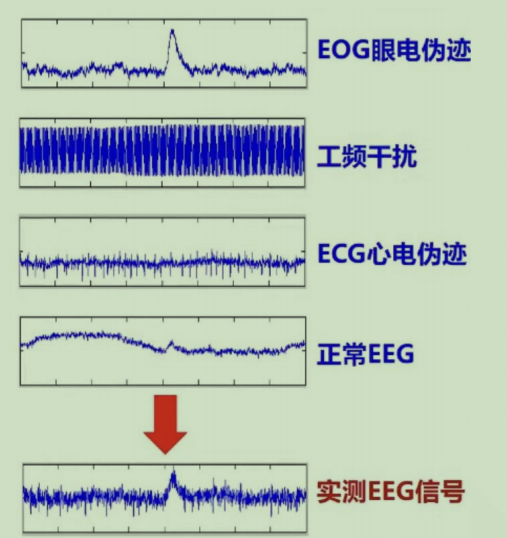
3.非侵入式：接受到头皮脑电（EEG）。头皮脑电由体外电极在头皮上采集，参数为5-300μV、<100Hz。可视化结果是无序的电位震动，相较于ECoG信号强度更弱。由于信号采集在体外，分析时除了分离信号源外，还要进行针对不同干扰类型的滤波处理，所以数据处理难度很大，但由于设备对人体没有创伤，是未来脑机接口普及化的重要方向。

采集到数据后，要根据数据特征的不同分出不同指令下的脑电状态。比如我们要区分人抬手与放手的脑电信号，以便操控机械臂，直观看数据，人很难分出不同的信号种类，甚至采集到高维信号，无法直接可视化。而通过主成分分析等特征提取、可视化手段，可以实现对数据的高效分类，从而搭建其脑电与动作指令之间的映射关系。

接下来主要对头皮脑电信号进行数据处理进行讨论。

二、脑电信号的放大、滤波

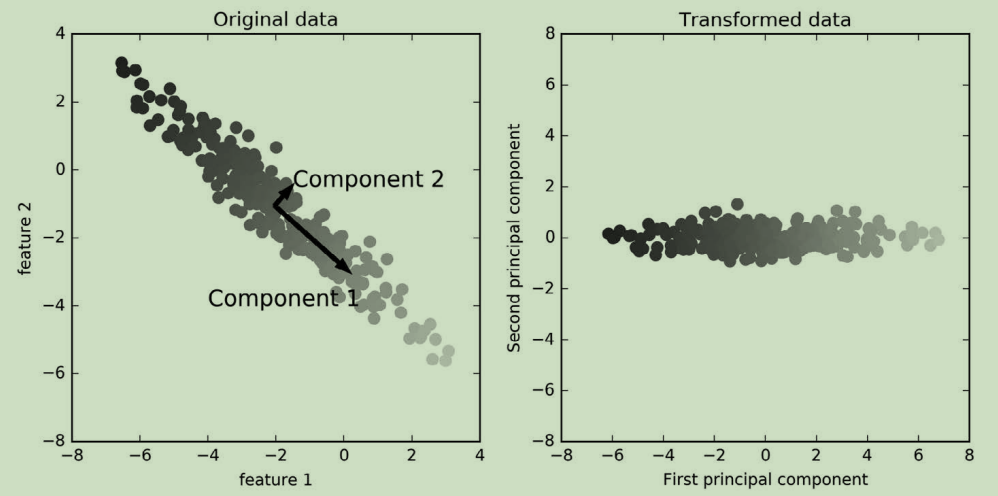
由于EEG极其微弱、受环境影响大，对其放大、滤波需要有“高增益、低噪声”的要求。高增益需要经EEG信号分2~3级逐级放大，低噪声要求针对脑电信号频率区域滤波。由于脑电频率处于约0.5~100Hz之间，其中包含50Hz的工频噪声，所以要对50Hz的工频干扰噪声专门滤除。



三、数据处理中使用的三个模型

1.主成分分析（PCA）：

主成分分析（principal component analysis，PCA）通过旋转数据，找到方差最大、关联最小的方向。这样做的目的是找到数据最大的“分歧点”，以便对数据进行分类。



（图片来源：《Python机器学习基础教程完整电子版》）

结合上图解释PCA原理：通过数学方面的分析，确定数据点方差最大、关联最小的方位（Component1），即“跨度”最长的方位。通过旋转，将数据点从新排布在平行于X轴的带状区域中，最终投影在X轴上，于是实现了数据从二维降为一维的特征提取过程。与之类似，通过同样的原理，可以将数据从n维降为k维（k<n）。具体数学实现要用到矩阵理论，下面展示我在专业课程作业中使用的PCA代码：

要求：“对于data4.npy中给定 10000 个一维且Shape为(128)的向量,将它们聚类成N类”。

import os

import numpy as np

import matplotlib as mpl

import matplotlib.pyplot as plt

Data = np.load(os.path.join('data', 'data4.npy'))

data = np.zeros\_like(Data)

mean = np.array([np.mean(Data[:,i]) for i in range(128)])

data = Data - mean

cov = np.cov(np.transpose(data))

Evalue, Evector = np.linalg.eig(cov)

newData = np.transpose(np.dot(Evector, np.transpose(data)))

Epairs = [(np.abs(Evalue[i]), Evector[:,i]) for i in range(len(Evalue))]

Epairs.sort(reverse=True)

key = np.zeros((2, 128))

key[0] = Epairs[0][1]

key[1] = Epairs[1][1]

chip = np.transpose(np.dot(key, np.transpose(data)))

plt.scatter(chip[..., 0], chip[..., 1])

plt.show()

（代码参考资料：https://www.jianshu.com/p/4528aaa6dc48）

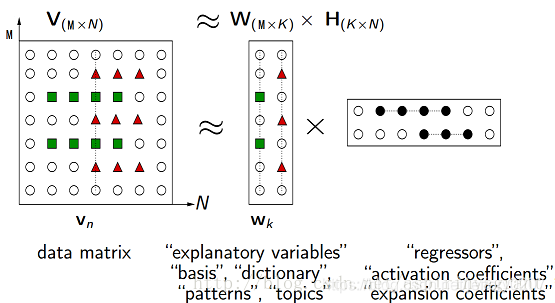
第一步：中心化。求各维平均数，减平均数，使各维数据中心在原点。

第二步：求特征值、特征向量。通过协方差得到。

第三步：找到表征方向。找到绝对值最大的特征值对应的特征向量，与原数据点乘，即得降维后的数据。

2.非负矩阵分解（NMF）：

非负矩阵分解（non-negative matrix factorization，NMF）类似于PCA，用于数据降维，本质是线性变换，将数据表示成基的线性组合。与PCA使用正交基不同，NMF中的基与系数非负，这样可以避免正负抵消带来的特征减少的问题。

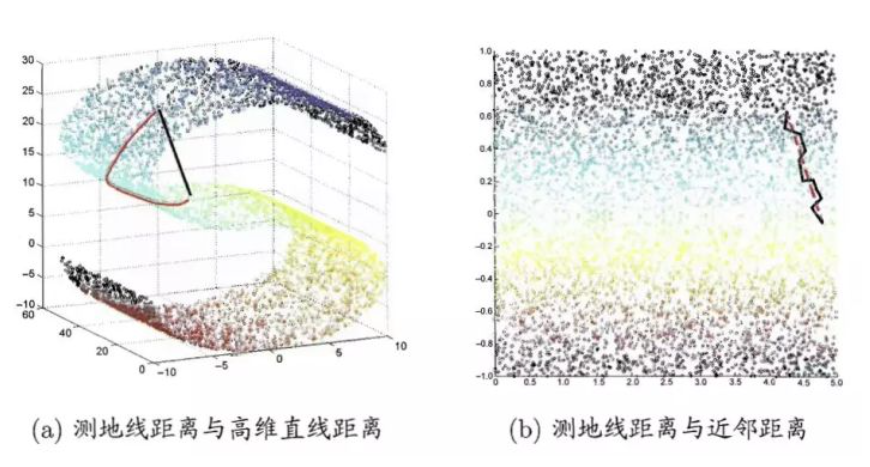


（图片来源：https://blog.csdn.net/jeffery0207/article/details/84348117）

所谓非负矩阵分解，就是把原来的数据矩阵分解为两个非负矩阵的矩阵乘积。V[m, n] = W[m, k]\*H[k, n]。V为原数据矩阵，W为基矩阵，H为权值矩阵。用H代替V，则实现了降维。

3. 分布随机邻域嵌入（t-SNE）：

分布随机领域嵌入（t-Distributed Stochastic Neighbor Embedding，t-SNE）是流形学习算法中的一种算法。所谓流形学习算法，即用低维坐标表示高维数据，可以想象成将空间中的点用空间曲面串联，最后铺展在二维平面上。



（图片来自：https://zhuanlan.zhihu.com/p/64664346）

t-SNE的目标是按照原空间中数据点之间的距离，使较近的数据更加靠近，较远的数据更加远离，最后表示在二维平面上。数学手段是用条件概率表示欧式空间距离、梯度下降等。

除了以上三种数据处理方法外，还有诸如叠加平均、陷波器滤波、独立成分分析等方法。但这些方法距离精确到每个神经元的最终目标还有一定距离，由于这样的算力局限，意识分析还不是近期能实现的目标。

参考文献：

1.李丽明教授《2022夏 神经科学前沿问题 第二讲 BCI分类及系统构成》PPT

2.《Python机器学习基础教程完整电子版》

3.【放大、滤波部分的文本内容】张学军、温炜《脑电信号调理电路设计》

4.【主成分分析部分的代码】JxKing《PCA（主成分分析）python实现》

5.【非负矩阵分析】jeffery0207《【机器学习】NMF(非负矩阵分解)》

6.【分布随机邻域嵌入】马上科普《t-SNE：最好的降维方法之一》

致谢：

感谢李丽明老师的深入讲解！

感谢王凯灵组长的框架与组织！

感谢四组组员的相互帮助！