多层感知机脑电波分类实验报告

李浩然

2017210914

1. 实验目的

给出20个人的脑电波采样数据以及对应的睡眠阶段。利用前19个人的脑电波-睡眠阶段作为训练数据，训练多层感知机（MLP）。再利用该模型预测第20个人的脑电波采样数据对应的睡眠阶段。将预测结果和真实结果作比较，并进行分析。

1. 数据描述

本次实验数据与之前贝叶斯判决实验使用同一数据集。

1. 实验环境

本次实验在Ubuntu 16.04系统中借助Octave 4.2.1软件完成。

处于编程方便考虑，本次实验将“数据”和“标签”文件夹重命名为“data”和“label”。

1. 实验原理

根据对⾃然神经系统构造和机理的认识，神经系统是由⼤量的神经细胞（神经元）构成的复杂⽹络，⼈们对这⼀⽹络建⽴⼀定的数学模型和算法，设法使他们能够实现诸如基于数据的模式识别、函数映射等带有智能的功能，建成神经⽹络（Neural Networks）,为模式识别的⽅法体系提供了新的⽅法。

⼈⼯神经⽹络（Artificial Neural Network，即ANN ），是20 世纪80 年代以来⼈⼯智能领域兴起的研究热点。它从信息处理⾓度对⼈脑神经元⽹络进⾏抽象，建⽴某种简单模型，按不同的连接⽅式组成不同的⽹络。在⼯程与学术界也常直接简称为神经⽹络或类神经⽹络。神经⽹络是⼀种运算模型，由⼤量的节点（或称神经元）之间相互联接构成。每个节点代表⼀种特定的输出函数，称为激励函数（activation function）。每两个节点间的连接都代表⼀个对于通过该连接信号的加权值，称之为权重，这相当于⼈⼯神经⽹络的记忆。⽹络的输出则依⽹络的连接⽅式，权重值和激励函数的不同⽽不同。⽽⽹络⾃⾝通常都是对⾃然界某种算法或者函数的逼近，也可能是对⼀种逻辑策略的表达。

BP(back propagation)神经⽹络是1986 年由Rumelhart 和McClelland 为⾸的科学家提出的概念，是⼀种按照误差逆向传播算法训练的多层前馈神经⽹络，是⽬前应⽤最⼴泛的神经⽹络。

基本BP 算法包括信号的前向传播和误差的反向传播两个过程。即计算误差输出时按从输⼊到输出的⽅向进⾏，⽽调整权值和阈值则从输出到输⼊的⽅向进⾏。正向传播时，输⼊信号通过隐含层作⽤于输出节点，经过⾮线性变换，产⽣输出信号，若实际输出与期望输出不相符，则转⼊误差的反向传播过程。误差反传是将输出误差通过隐含层向输⼊层逐层反传，并将误差分摊给各层所有单元，以从各层获得的误差信号作为调整各单元权值的依据。通过调整输⼊节点与隐层节点的联接强度和隐层节点与输出节点的联接强度以及阈值，使误差沿梯度⽅向下降，经过反复学习训练，确定与最⼩误差相对应的⽹络参数(权值和阈值)，训练即告停⽌。此时经过训练的神经⽹络即能对类似样本的输⼊信息，⾃⾏处理输出误差最⼩的经过⾮线形转换的信息。

1. 实验过程
   1. 读入数据

本次实验使用与之前两个实验使用相同的读入数据方法，得到仅包含1~3睡眠阶段的训练数据(sam, lab)和测试数据(tsam, tlab)。

* 1. 数据预处理

本次实验沿用上次实验处理过的训练数据。

* + 1. FFT变换

训练数据是时域上的采样，需要转换到频域才能统一各个分量的意义。

* + 1. 正则化

上次实验讨论过，正则化是PCA变换之前重要的步骤，能提高PCA的表示能力。

* + 1. PCA变换

贝叶斯实验表明，保留PCA前6个分量（78.6%方差信息）可以达到最高的准确率，本次实验沿用这个参数。

* + 1. 过滤异常点

上次实验中讨论了总体符合正态分布的样本中存在较多异常数据点，故过滤掉偏离均值三个标准差以上的数据。

* + 1. 线性缩放

SVM实验中需要数据各个分量的规模一致，所以将数据的各个分量线性映射到[-1, 1]区间。

* 1. 多层感知机

本次实验使用Octave Forge中提供的nnet 0.1.13包实现的MLP工具集。

* + 1. 训练

利用函数PR = min\_max(sam)求得样本中各个特征的最小最大值。

利用函数net = ewff(PR, SS)生成每层规模由SS向量指定的、输入信号在PR范围内的多层感知机。

保持默认的参数：

* 相关层的传递函数，默认隐含层为tansig函数，输出层为purelin函数。
* BP神经网络学习训练函数，默认值为trainlm函数。
* 权重学习函数，默认值为learngdm。
* 性能函数，默认值为mse。

利用函数model = train(net, sam, lab)训练得到多层感知机。

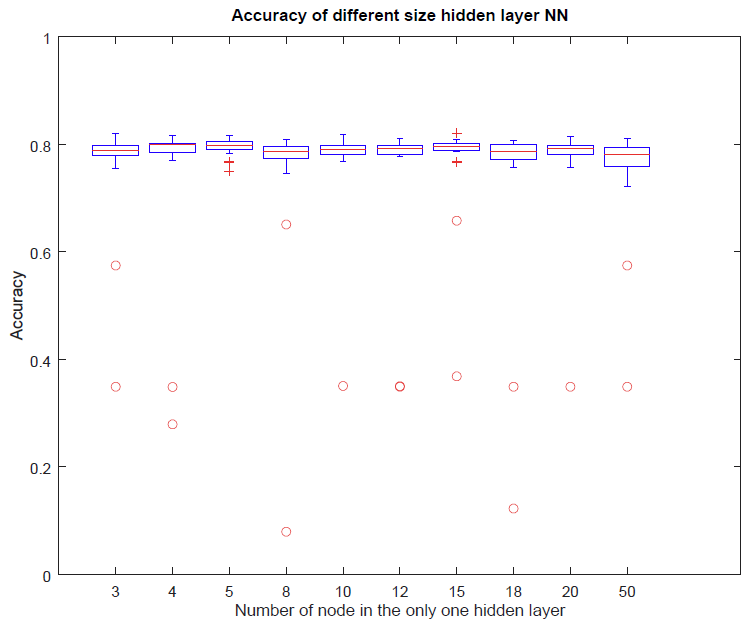
* + 1. 预测

利用函数pred = sim(model, tsam)得到对于测试数据的预测结果。这里pred是概率矩阵，给出各个测试样本属于每一类的概率。所以需要选举一个最终结果：pred = max(pred)。

将结果和给出的测试集标签进行比较，求出最终的准确率。

1. 实验结果
   1. 隐藏层规模

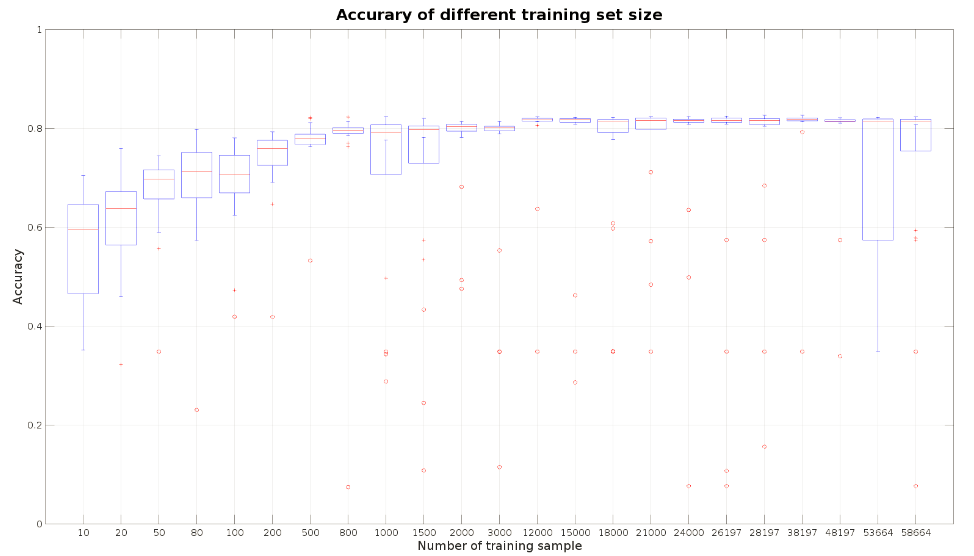
本次实验采用两个隐藏层，为了确定每个隐藏层的规模，尝试不同隐藏层规模，对于每个规模，训练二十次，观察准确率的分布情况。



可以发现不同规模的隐藏层对于结果的印象并不大，所以接下来的实验采用[6 5 4 3]规模的多层感知机进行训练。

* 1. 训练集大小

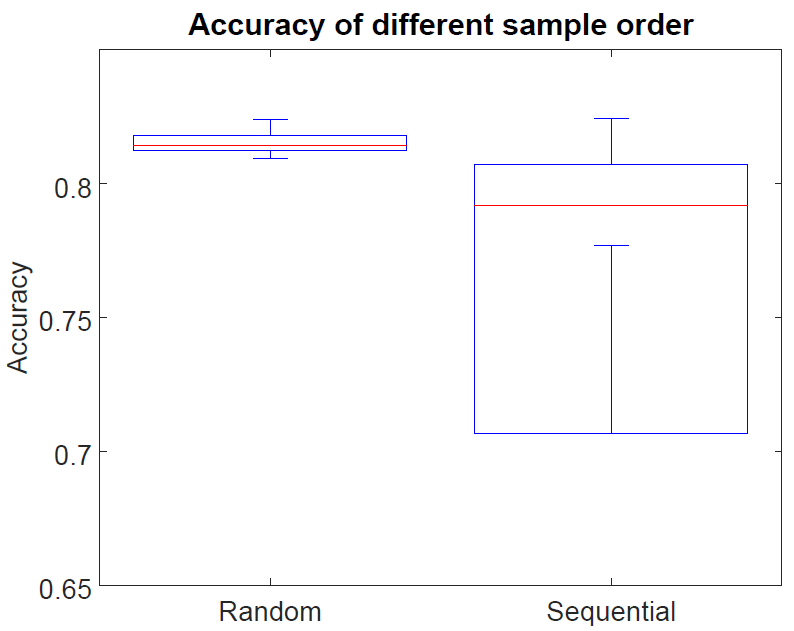
为了确定准确率关于训练样本大小的关系，逐步增加训练样本数量，每次训练二十次，观察准确率的分布。



可以看出随着训练样本数量的增加，准确率收敛到80%左右。

* 1. 训练集顺序

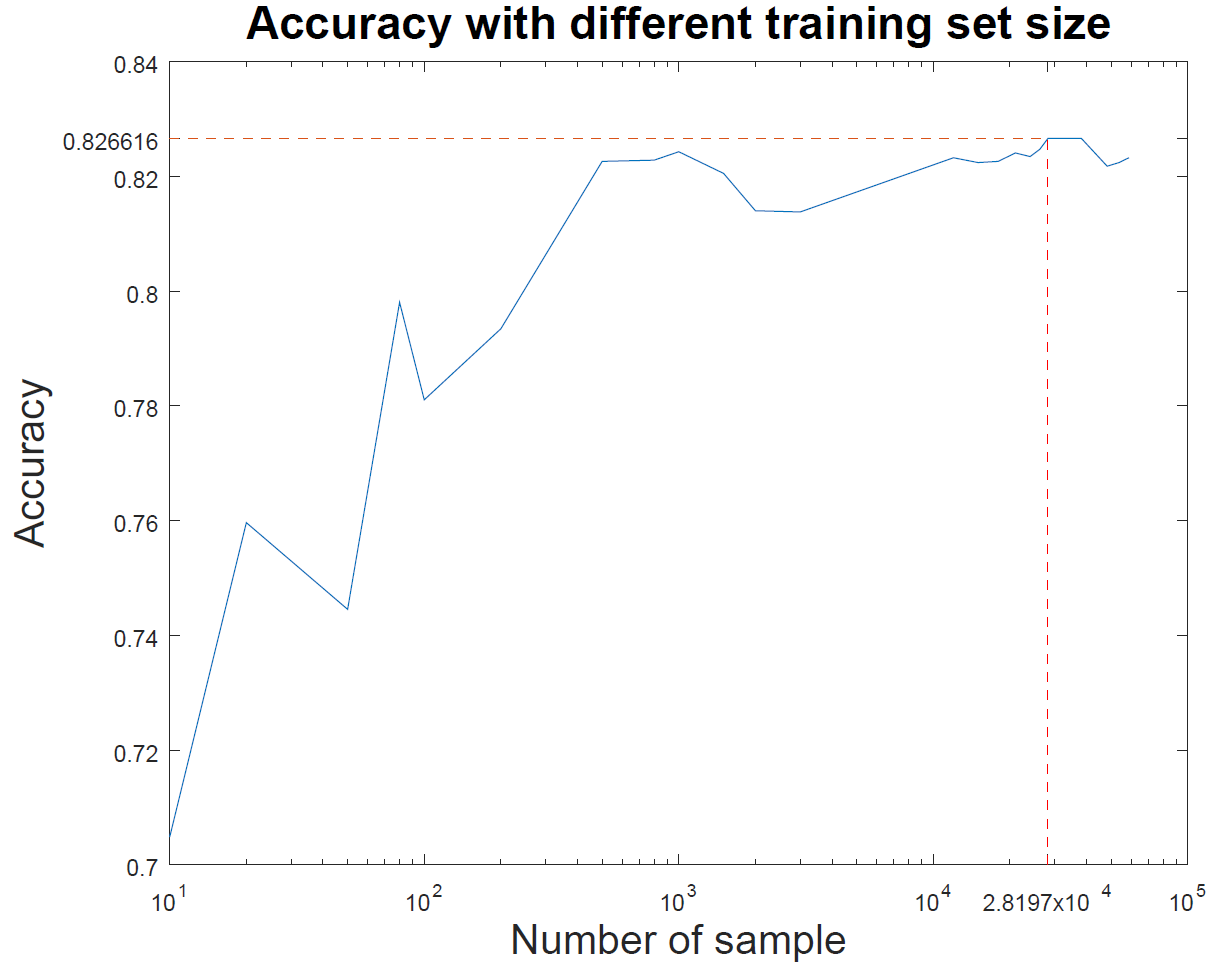
之前的实验中，训练数据高度有序，总是第一类排在最前，第二类排中间，第三类放最后，尝试随机训练数据的排列，观察准确率分布情况。



上图选用每类各500个样本，各进行20次训练。观察可以发现，随机样本的训练结果稳定性以及准确率都更高。

* 1. 最终准确率

将6.2中结果每组取最高准确率得到准确率的最终结果。



由上图可得，本次实验中多层感知机的预测准确率最高约为82.66%。

1. 结果分析

观察实验数据可以发现，随机多次训练多层感知机得到的模型预测准确率不尽相同，大致呈现训练样本数量越小，预测准确率分布越离散的规律。一个原因是训练样本太少不足以表达每一类的特征规律，另一个原因是多层感知机一开始随机设定参数再迭代更新，所以不同的初值可能收敛到不同的极小值，因为训练数据的特征不够明显，某些极小值就可能表现不够优秀。