

## 实验一：MLP

---

- 数据预处理：按编号将被试人员的数据分组读入，共读入十二组数据，每组数据的62\*5维数据直接摊平为310维的数据，因此每个被试的数据是 `num*310` 维。同时，对每个组的数据进行归一化处理，因为是个三分类问题，因此我们采用了Z-score 标准化进行归一化处理。
- 为了实现跨被试留一，将每个被试的数据都进行编号，在训练和测试时利用 `LeaveOneGroupOut` 实现跨被试留一测试
- 模型结构：采用简单MLP，共四层，分别为 `(310, 256), (256, 128), (128, 64), (64, 32), (32, 16)`，每层之间均采用 `ReLU` 作为激活函数，最后的输出层维度为32维，采用 `Softmax` 函数将输出转换为概率分布，从而实现多分类的目标
- 超参设置：由于该模型较小，我们可以采用网格搜索来寻找最合适的超参
- `'learning_rate': [0.1, 0.01, 0.001]`
- `'hidden_layer_sizes': [[256, 128, 64, 32], [128, 64, 32], [64, 32]]`,
- `'activation_function': [nn.ReLU, nn.Tanh, nn.LeakyReLU]`,
- `'optimizer': ['Adam', 'SGD']`,
- `'num_epochs': [25, 60, 100, 250]`
- 最终的超参组合是： `lr:0.001, layer:[256, 128, 64, 32], activation:nn.ReLU, optimizer:Adam, epoch:60`
- 最终结果： `Mean Accuracy: 0.75 Std Accuracy 0.08`
- 结果分析：数据归一化是很关键的操作，归一化前准确率均值只有0.5左右，归一化之后可以到0.75

## 实验二：LSTM

---

- 数据预处理：按编号将被试人员的数据分组读入，共读入十二组数据，每组数据的62\*5维数据直接摊平为310维的数据，因此每个被试的数据是 `num*310` 维。同时，对每个组的数据进行归一化处理，因为是个三分类问题，因此我们采用了Z-score 标准化进行归一化处理。
- 模型特性：因为是时间序列模型，我们构造了窗口为3的时间序列，每连续3组数据作为一个时间窗口进行训练。
- 模型结构：采用LSTM模型，线性全连接层。
- 超参设置：根据MLP的调参经验，设置隐藏层单元数为128，训练轮数50，但这些参数仅仅是凭经验经过少量测试得出。
- 最终结果： `Mean = 0.7506, Std = 0.0799`
- 结果分析：由于数据在采集时是具有时序关系的，因此我们的确可以使用LSTM进行训练，但事实上窗口的大小在本次实验中是作为超参来调整的，但事实上应当根据数据实验实际情况进行设置。

## 实验三：CNN

---

- 数据预处理：对于一组  $62 \times 5$  的数据，我们按照数据来源的空间分布将数据拼成一个  $8 \times 9$  的矩阵，且  $\text{channel}=5$ ，对于不存在的数据位置，我们全设置为0，并采用Z-score 标准化进行归一化处理，从而使得数据可以作为一个图片输入CNN。
- 模型特性：因为是时间序列模型，我们构造了窗口为3的时间序列，每连续3组数据作为一个时间窗口进行训练。
- 模型结构：使用 3D CNN，两个卷积层，一个池化层，两个全连接层，两个卷积核的大小都是  $(3, 3, 3)$ ，填充是  $(1, 1, 1)$ ，池化层为  $(2, 2, 2)$
- 参数设置：与前两个实验基本一致，由于模型训练比较缓慢因此并没有进行太多的调参
- 最终结果：Mean = 0.7340, Std = 0.0797
- 结果分析：使用CNN进行训练依赖于数据的空间分布，将数据按照空间正确的分布可以让模型学到不同采集点之间的关系，从而更好的预测。