0. 摘要

本次项目研究的是通过脑电信号直接评估抑郁症，希望通过几种算法基于脑电抑郁症建模并使用交叉验证对模型性能进行性能评估。通过这类算法我们得到了脑电信号指标和抑郁症的关系，也得到了针刺对治疗抑郁症效益的相关性。本次项目结果说明通过建模我们可以很好的评估拟合预测患者是否患有抑郁症，且中医针刺作为中国传统的医疗手段对此类症状也有较好的治疗效果。

1. 背景介绍

在当今社会，抑郁症已经成为许多上班族甚至学生们的热词。它已经严重影响到患者的学习、工作和正常社交生活，现在也都在寻求治疗它的方法。目前，治疗抑郁症手段主要包括抗抑郁药物和中医针刺。本次项目拟通过脑电信号直接评估抑郁症，以便更好的解决这方面的难题。

目前全世界对于抑郁症的关注度也很高，其中也不乏有希望脑电信号拟合得到有关病症的数据，让脑电波像血糖一样容易理解。其目的是开发出抑郁和焦虑的客观测量值，以用于支持诊断、治疗和治疗抑郁症。美国谷歌 Amber 团队试图将机器学习技术与脑电图(EEG)相结合，以测量大脑的电活动。当然，X LAB 并不是第一个将机器学习算法应用于脑电图读数的公司。在去年 4 月发表的一篇论文中，IBM 的研究人员声称已经开发出一种算法，可以对癫痫发作进行分类，准确率高达 98.4%。事实上，脑电图已被广泛用于研究吞咽、分类精神状态、诊断神经精神疾病，如神经源性疼痛和癫痫，以及分类情绪。

本次项目任务：(1)分类问题：根据脑电指标直接评估患者是否抑郁。(2) 拟合问题：根据脑电指标直接拟合 HAMD-17 分值。

3. 实验方法

3.1 数据预处理

3.1.1. 数据探索

本次实验的数据集“data_depression.xlsx”包含两个数据表单：“抑郁脑电图”和“针刺效益、前后脑电图”。表单“抑郁症脑电图”字段有姓名、性别、年龄、HAMD 评分、以及针刺前脑电节律波幅指数和节律指数。表单“针刺效益、前后脑电图”字段主要有姓名、针刺前脑电指标和针刺后脑电指标。根据 17-item HAMD 抑郁症评价量表，字段 HAMD 评分是患者抑郁症程度评价。当 HAMD-17 量表评分<=17 分时为正常；当 HAMD-17 量表评分>17 分时为抑郁症。

对于节律指数的数据源描述性统计如下表。由于总共有 76 个属性，只列出部分结果。

	0	1	2	3	4	5	6	7	8	9	...
count	92.000000	92.000000	92.000000	92.000000	92.000000	92.000000	92.000000	92.000000	92.000000	92.000000	...
mean	69.003261	35.840217	40.268478	21.192391	64.526087	31.931522	37.963043	19.454348	43.670652	30.498913	...
std	46.165827	19.601094	18.311246	11.630712	44.933932	14.804424	13.796844	9.186079	32.932967	15.112563	...
min	16.000000	9.100000	12.000000	8.400000	10.600000	8.900000	12.600000	8.400000	8.700000	7.500000	...
25%	35.100000	23.600000	28.475000	13.675000	29.075000	21.875000	28.050000	13.675000	22.450000	20.100000	...
50%	57.850000	32.550000	36.300000	17.850000	57.100000	30.100000	36.000000	17.300000	37.500000	28.750000	...
75%	86.850000	39.700000	48.250000	26.825000	85.950000	37.100000	46.725000	23.750000	53.375000	37.125000	...
max	230.400000	131.000000	109.000000	82.100000	229.100000	98.800000	73.600000	61.700000	203.700000	76.400000	...

8 rows × 76 columns

图 1 节律指数数据源描述图

3.1.2.数据预处理

在本次实验给出的数据中，使用 pandas 中的 isnull()函数判断数据集不存在缺失值。对于分类任务，从数据集中读取 HAMD 评分和δ节律、θ节律 、α节律、 β(LF)节律四个节律指数对应的导联数据。同时将 HAMD 评分中小于等于 17 分的标签设为 0，大于 17 分时设为 1。

由于有 72 个特征，接着对数据进行特征选择。本次实验使用 sklearn 中的 SelectKBest 方法进行特征选择和降维。使 72 个特征降维处理成 10 个特征。

3.2 算法介绍

分类：

3.2.1 SVM

支持向量机（support vector machines, SVM）是一种二分类模型，其主要思想为找到空间中的一个更够将所有数据样本划开的超平面，并且使得样本集中所有数据到这个超平面的距离最短。

SVM 的基本模型是定义在特征空间上的间隔最大的线性分类器，间隔最大使它有别于感知机，除此之外还包括核技巧，这使它成为实质上的非线性分类器。SVM 的的学习策略就是间隔最大化，可形式化为一个求解凸二次规划的问题，也等价于正则化的合页损失函数的最小化问题。SVM 的的学习算法就是求解凸二次规划的最优化算法。

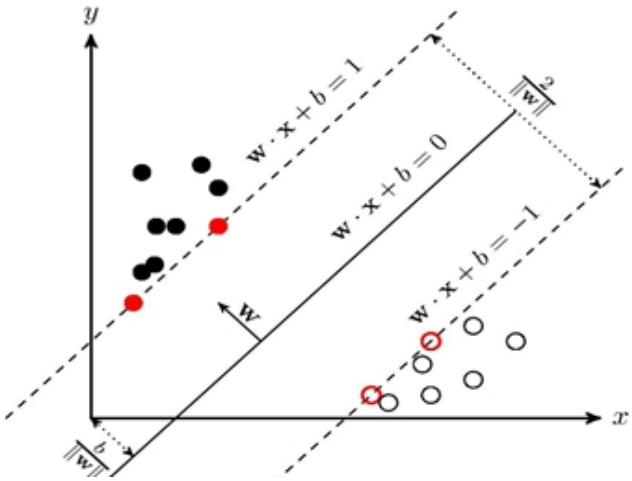
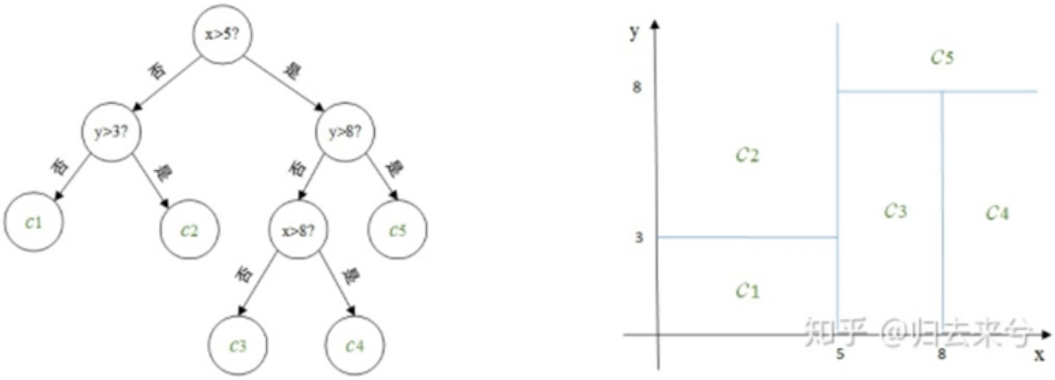


图 2 SVM 模型

拟合：

3.2.2 决策树回归：

回归树就是将特征空间划分成若干单元，每一个划分单元有一个特定的输出。
因为每个结点都是“是”和“否”的判断，所以划分的边界是平行于坐标轴的。
对于测试数据，我们只要按照特征将其归到某个单元，便得到对应的输出值。



3.2.3 线性回归

线性回归假设目标值与特征之间线性相关，即满足一个多元一次方程。通过构建损失函数，来求解损失函数最小时的参数 w 和 b。如下公式：

$$\hat{y} = wx + b$$

3.2.3KNN 回归

通过找出一个样本的 k 个最近邻居，通过一定的处理（如将这些邻居的属性的平均值）赋给该样本，这样就可以得到该样本对应属性的值。

3.3 评价指标

3.3.1 分类任务评价指标

对于实验中的分类任务使用的评价指标有：混淆矩阵、精确率、召回率、F1 值、ROC 曲线的 AUC 值。其中精确率、召回率、F1 值可以从混淆矩阵中直接计算得到。

3.3.1 confusion matrix

表 1 为混淆矩阵的结构，其中行表示数据在模型上的预测类别（predicted class/predicted condition），列表示数据的真实类别（actual class/true condition）。通过混淆矩阵，我们可以很直观地看清一个模型在各个类别（positive 和 negative）上分类的情况。在混淆矩阵中，TP 是指：真实类别为 positive，模型预测的类别也为 positive；FP 是指：预测为 positive，但真实类别为 negative，真实类别和预测类别不一致；FN 是指：预测为 negative，但真实类别为 positive，真实类别和预测类别不一致；TN 是指：真实类别为 negative，模型预测的类别也为 negative。

表 1：混淆矩阵

		Actual class	
		positive class	negative class
Predicted class	positive class	True Positive(TP)	False Positive(FP)
	negative class	False Negative(FN)	True Negative(TN)

3.3.2 精确率（查准率）和召回率

精确率表示在预测为 positive 的样本中真实类别为 positive 的样本所占比例。其计算公式为：$\text{precision} = \frac{TP}{TP + FP}$ 。

positive class 的召回率表示在真实为 positive 的样本中模型成功预测出的样本所占比例。其计算公式为：$\text{recall} = \frac{TP}{TP + FN}$

3.3.3 F1 值

F1 值就是精确率和召回率的调和平均值，使用 F1 值可以对 Precision 和 Recall 进行整体评价。其计算公式为：$F_1 = \frac{2}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}}$ 。

3.3.4 ROC 曲线的 AUC 值

AUC（Area Under Curve）被定义为 ROC 曲线下与坐标轴围成的面积，显然这个面积的数值不会大于 1。又由于 ROC 曲线一般都处于 y=x 这条直线的上方，所以 AUC 的取值范围在 0.5 和 1 之间。AUC 越接近 1.0，检测方法真实性越高;等于 0.5 时，则真实性最低，无应用价值。

其中，ROC 曲线全称为受试者工作特征曲线 （receiver operating characteristic curve），它是根据一系列不同的二分类方式（分界值或决定阈），以真阳性率（敏感性）为纵坐标，假阳性率（1-特异性）为横坐标绘制的曲线。

从 AUC 判断分类器（预测模型）优劣的标准：

1. AUC = 1，是完美分类器，采用这个预测模型时，存在至少一个阈值能得出完美预测。绝大多数预测的场合，不存在完美分类器。

2. 0.5 < AUC < 1，优于随机猜测。这个分类器（模型）妥善设定阈值的话，能有预测价值。

3. AUC = 0.5，跟随机猜测一样（例：丢铜板），模型没有预测价值。

4. AUC < 0.5，比随机猜测还差；但只要总是反预测而行，就优于随机猜测。

3.3.1 拟合任务评价指标

1.SSE(误差平方和)

计算公式如下：

$$SSE = \sum (Y\_actual - Y\_predict)^2$$

同样的数据集的情况下，SSE 越小，误差越小，模型效果越好

缺点： SSE 数值大小本身没有意义，随着样本增加，SSE 必然增加，也就是说，不同的数据集的情况下，SSE 比较没有意义

4. 结果与讨论

4.1 实验环境

本次实验使用的环境为：python3.6.8，pandas0.25.0，numpy1.17.0，matplotlib3.1.1，scikit-learn0.21.3，scipy-1.4.1，graphviz-0.14.2，openpyxl 3.0.5

4.2 实验数据

使用网格搜索后找到较优的参数

4.2.1 SVM 实验数据

SVM 方法模型使用了 sklearn 中的 SVC 函数，选择的类型是监督学习，选择的核函数为 rbf，惩罚参数 C 为 10，核函数参数 gamma 为 0.5，调参方法采用网格搜索。具体的参数如下所示：

```
SVC(C=10, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma=0.5, kernel='rbf',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
```

4.2.2 决策树回归实验数据

```
DecisionTreeRegressor(ccp_alpha=0.0, criterion='mse', max_depth=1,
                      max_features=None, max_leaf_nodes=None,
                      min_impurity_decrease=0.0, min_impurity_split=None,
                      min_samples_leaf=1, min_samples_split=2,
                      min_weight_fraction_leaf=0.0, presort='deprecated',
                      random_state=None, splitter='best')
```

4.2.3 线性回归实验数据

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

4.2.4 knn 回归实验数据

```
KNeighborsRegressor(algorithm='auto', leaf_size=30, metric='minkowski',
                    metric_params=None, n_jobs=None, n_neighbors=6, p=2,
                    weights='uniform')
```

4.3 实验结果

4.3.1 SVM 实验结果

svc 模型的得分为 0.7143。使用 roc 曲线的方式来评价该分类模型，其 auc 最终结果为 0.76。其混淆矩阵和 ROC 曲线如下图所示。

	precision	recall	f1-score	support
0	0.74	0.89	0.81	19
1	0.60	0.33	0.43	9
accuracy			0.71	28
macro avg	0.67	0.61	0.62	28
weighted avg	0.69	0.71	0.69	28

auc: 0.7602339181286549

图 1 SVM 混淆矩阵

图 2 svm 的 roc 曲线

为检验模型的稳定性，使用 10 折交叉验证，其最终平均得分为 0.698。如下图为交叉验证结果可视化图。

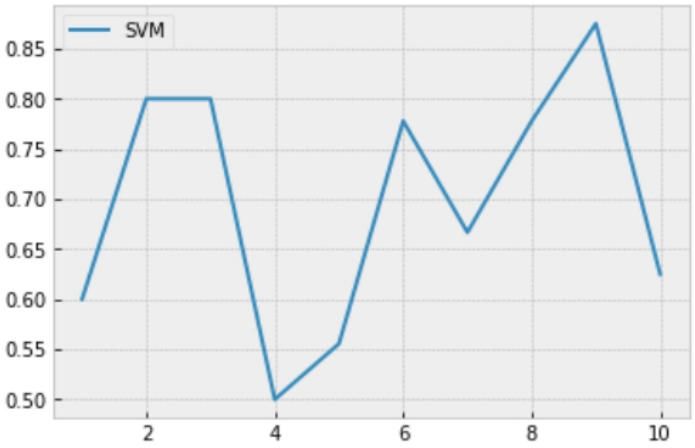
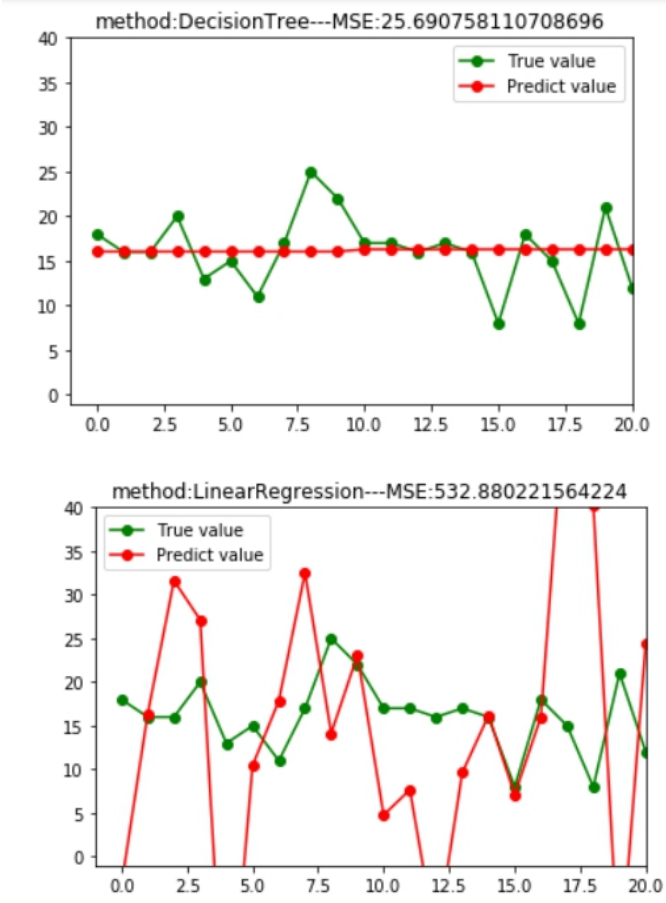
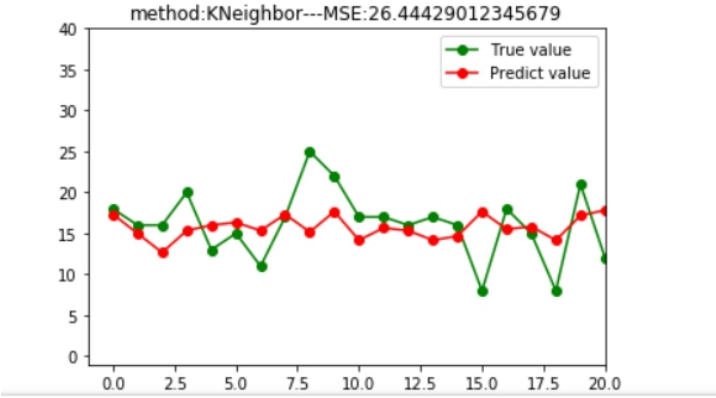


图 3 SVM 交叉验证得分

4.3.2 回归





4.4 结果讨论

分类模型上:

SVM 结果讨论：本次实验使用 SVM 方法分类效果较好，其 AUC 最终得分有 0.76，但其模型的准确率并不高，这可能与模型选择的特征及数据集的大小有关。

回归模型上:

参考 MSE 的值，预测的值与真实的值的曲线，发现 KNN 模型训练集的拟合效果较好，MSE 值(均方误差)=26.44

问题和优化:

①样本数据集太小，特征数又多，即使采用交叉验证与集成学习的方法都没办法提升模型拟合效果都不够理想

②数据集中噪声数据，我们没有发现并进行清洗范围的预测

5. 总结

1.通过这次的项目实验，我们小组熟悉了 sklearn 中的基本分类算法的使用，熟悉了机器学习的整个流程，包括数据预处理，训练模型，模型评估，调整模型，保存模型。在实验的过程中当然遇到了不少的问题，但通过查资料和小组群里的讨论基本都解决了。

2.我们小组成员进一步地了解了具体的各个算法的基本原理，通过动手实验，我们加深了各个算法的理解。

参考文献

1. 《scikit-learn Machine Learning in Python》 https://scikit-learn.org/stable/
2. 《回顾及总结--评价指标》
https://blog.csdn.net/chocolate_chuqi/article/details/81112051
3. 《谷歌发布 Amber 项目，用 AI 分析脑电波诊断治疗抑郁症》
https://www.jiqizhixin.com/articles/2020-11-04-11