深圳技术大学考试答题纸

(以论文、报告等形式考核专用)  
二○ 23 ～二○ 24 学年度第 1 学期

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 课程编号 | IB00123 | | 课程名称 | | 机器学习导论 | | 主讲教师 | | 王小航 | 评分 |  |
| 学 号 | 202100202016 | | 姓名 | 何锐生 | | 专业年级 | | 21级计算机科学与技术1班 | | | |
|  | | | | | | | | | | | |
| 教师评语： | | | | | | | | | | | |
| 题目： | | 基于机器学习的脑中风预测 | | | | | | | |  | |

摘要：随着医学数据的不断积累和机器学习技术的快速发展，预测脑中风成为了一个重要的研究领域。本研究旨在探讨不同机器学习模型在脑中风预测中的应用，并比较它们的性能。我们使用了包含性别、年龄、高血压、心脏病、婚姻状况、工作类型、居住类型、平均血糖水平、BMI指数和吸烟状况等10个特征的数据集。通过逻辑回归、线性支持向量机（Linear SVM）、非线性支持向量机（Non-linear SVM）、决策树和随机森林等多种机器学习模型进行训练和测试，并通过网格搜索和交叉验证调优模型参数。

关键词：脑中风预测、逻辑回归、线性SVM、非线性SVM、决策树、随机森林、网格搜索、交叉验证

**目录**

[1.数据集的背景描述 3](#_Toc155087797)

[2.方法和模型的介绍 3](#_Toc155087798)

[2.1逻辑回归 3](#_Toc155087799)

[2.2支持向量机 4](#_Toc155087800)

[2.3决策树 6](#_Toc155087801)

[2.4随机森林 7](#_Toc155087802)

[2.5模型评价指标 8](#_Toc155087803)

[2.5.1混淆矩阵 (Confusion Matrix) 8](#_Toc155087804)

[2.5.2准确性 (Accuracy) 8](#_Toc155087805)

[2.5.3召回率 (Recall) 9](#_Toc155087806)

[2.5.4精确度 (Precision) 9](#_Toc155087807)

[2.5.5特异性 (Specificity) 9](#_Toc155087808)

[2.5.6 F1分数 (F1 Score) 9](#_Toc155087809)

[2.6网格搜索 10](#_Toc155087810)

[2.7交叉验证 10](#_Toc155087811)

[2.8集成学习 11](#_Toc155087812)

[3.数据探索 12](#_Toc155087813)

[3.1特征及标签的介绍 12](#_Toc155087814)

[3.2密度图 13](#_Toc155087815)

[3.3热力图 15](#_Toc155087816)

[4.建模过程与结果 16](#_Toc155087817)

[4.1逻辑回归 16](#_Toc155087818)

[4.2线性SVM 17](#_Toc155087819)

[4.3非线性SVM 18](#_Toc155087820)

[4.4决策树 19](#_Toc155087821)

[4.5随机森林 21](#_Toc155087822)

[5.总结 22](#_Toc155087823)

# 1.数据集的背景描述

中风是一种医学疾病，由于流向大脑的血液不足导致细胞死亡。中风主要有两种类型：缺血性中风（缺乏血液流动导致）和出血性中风（出血导致）。两者都会导致大脑的某些部分停止正常运作。

中风的主要危险因素是高血压。其他危险因素包括高血胆固醇、吸烟、肥胖、糖尿病、以前的TIA、终末期肾病和心房颤动。

据统计，中风是全球死亡和残疾的重要原因之一，且中风患者数仍在不断增加。虽然中风对人类的危害极大，但幸运的是，中风在很大程度上是可以预防的。本文章的主要目的就是根据给定数据集的特征来探讨它们与中风的关系，以达到预测疾病的目标，为人类的健康贡献一份力。

# 2.方法和模型的介绍

## 2.1逻辑回归

逻辑回归是一种用于解决二分类问题的机器学习算法。它基于 logistic 函数（也称为 sigmoid 函数）进行建模，该函数可以将任意实数映射到一个介于0和1之间的概率值。逻辑回归常用于估计某个实例属于正类别的概率。

逻辑回归的原理如下：

①Logistic 函数是逻辑回归的核心。它的数学表达式为：

其中是输入的线性组合，而表示将映射到一个介于0和1之间的概率值。

②逻辑回归的模型预测概率如下：

如果大于等于某个阈值（通常为0.5），则模型预测为正类别；否则，预测为负类别。

③模型的参数通过最大似然估计等方法进行学习。

## 2.2支持向量机

支持向量机（Support Vector Machine，SVM）是一种强大的监督学习算法，用于分类和回归任务。其基本原理是找到一个超平面，将不同类别的样本分隔开，同时最大化样本到超平面的间隔。

支持向量机的主要原理如下：

①超平面：在二维空间中，超平面是一条直线；在三维空间中，超平面是一个平面；而在更高维度的空间中，超平面是一个超平面。SVM 的目标就是找到一个超平面，使得样本点离这个超平面的距离最远。

②在样本空间中，划分超平面可通过以下线性方程来描述：

其中𝒘=(𝑤\_1,𝑤\_2,…, 𝑤\_𝑑)为法向量，决定了超平面的方向；𝑏为位移项，决定了超平面与原点之间的距离，将超平面记为(𝒘,𝑏)。

假设超平面(𝒘,𝑏)能将训练样本正确分类，即对于∈𝐷，若=+1，则有 +𝑏>0；若=−1，则有+𝑏<0。令

③支持向量：距离最佳超平面最近的几个训练样本点使上式中的等号成立，它们被称为“支持向量”（support vector）。

④间隔：间隔是指支持向量到超平面的距离。SVM的优化目标是最大化间隔。

⑤硬间隔与软间隔： SVM可以处理线性可分问题（硬间隔）和线性不可分问题（软间隔）。软间隔允许一些样本点出现在错误的一侧，通过引入惩罚项来平衡最大化间隔和最小化分类错误之间的权衡。

在探究过程中，我使用了线性svm和非线性svm这两种支持向量机，接下来我将详细介绍一下这两种支持向量机：

①线性 SVM 适用于线性可分的情况，即数据集可以在特征空间中被一个超平面完美地分开。线性 SVM 的决策边界是一个线性函数，可以用来解决二分类和多分类问题。线性 SVM 的优势在于模型相对简单，训练速度较快，并且对于高维数据集表现良好。

线性 SVM 的数学表达式如下：

其中，是法向量（决策边界的法向量），是输入样本的特征向量， 是偏置项，是符号函数，用于判断样本位于超平面的哪一侧。

②非线性SVM用于处理数据集在特征空间中不是线性可分的情况。为了处理非线性关系，非线性SVM引入了核函数，将输入特征映射到高维空间，使得数据在高维空间中变得线性可分。

常见的核函数包括：

多项式核（Polynomial Kernel）：，其中 是常数， 是多项式的次数。

径向基函数核（Radial Basis Function, RBF或Gaussian Kernel）：，其中是带宽参数。

Sigmoid核：，其中 和 是参数。

非线性SVM的数学表达式如下：

其中， 是 Lagrange 乘子， 是支持向量， 是对应支持向量的类别标签。

非线性 SVM 的优势在于可以处理更加复杂的决策边界，但相应地，模型的复杂度和训练时间可能会增加。在选择核函数时，需要根据数据的特点和问题的需求来调整。

## 2.3决策树

决策树是一种基于树状结构进行决策的监督学习算法，主要用于分类和回归任务。

决策树的基本原理主要如下：

①树结构：决策树由树结构构成，包含节点和边。每个节点表示一个属性或特征，边表示在该属性上的取值。树的最顶部是根节点，最底部是叶节点。

②节点划分：决策树通过对数据集中的特征进行递归划分来生成树的结构。在每个节点上，选择一个特征进行划分，根据该特征的取值将数据集分成不同的子集。这个过程一直持续，直到满足某个停止条件，例如达到最大深度、节点上的样本数量小于某个阈值等。

③信息增益（ID3、C4.5）或基尼系数（CART）：决策树的关键是如何选择最优的特征进行划分。在分类问题中，常用的方法包括信息增益（用于ID3、C4.5算法）和基尼系数（用于CART算法）。这些方法衡量了选择不同特征后所获得的纯度或不确定性的减少。

④预测：对于新的样本，从根节点开始根据特征的取值依次遍历树的分支，最终到达叶节点。叶节点的类别或数值即为对新样本的预测结果。

⑤剪枝：为防止过拟合，决策树可能会在构建完成后进行剪枝。剪枝的目标是去除一些不必要的节点，以提高模型的泛化能力。

决策树的优点包括易于理解和解释、不需要对数据进行标准化、能够处理离散和连续型特征等。然而，决策树也容易过拟合，因此常常需要通过剪枝等手段来控制模型的复杂度。

## 2.4随机森林

随机森林是一种集成学习方法，通过组合多个决策树来进行分类和回归任务。

随机森林的基本原理主要如下：

①随机抽样：随机森林使用 Bootstrap 抽样技术，从原始数据集中有放回地随机抽取多个子样本，每个子样本用于训练一个单独的决策树。

②随机特征选择：在每个决策树的节点划分时，不是考虑所有的特征，而是从所有特征中随机选择一个子集。这样做的目的是增加树与树之间的差异性，提高整体模型的泛化性能。

③多个决策树的投票或平均：随机森林中的每个决策树都是独立地训练的。对于分类问题，多个树的投票决定最终的分类结果；对于回归问题，多个树的输出值取平均得到最终的预测结果。

④Bagging和过拟合的抑制：随机森林通过 Bagging的方式，降低了模型的方差，抑制了过拟合。每个决策树只看到部分数据，减小了单个决策树对噪声的敏感性。

⑤袋外估计：在每次抽样时，大约有 37% 的样本未被抽到，这些未被抽到的样本可以用于袋外估计，即在不使用交叉验证的情况下评估模型性能。

随机森林有以下三个优点：

①具有很好的泛化性能，对于高维数据和大量特征的情况表现较好。

②能够处理大规模数据集和具有高度非线性关系的问题。

③对于缺失值和不平衡的数据集具有较强的鲁棒性。

## 2.5模型评价指标

### 2.5.1混淆矩阵 (Confusion Matrix)

混淆矩阵是一个二维表格，用于比较模型的预测结果和实际类别。对于二分类问题，混淆矩阵如下：

|  |  |  |
| --- | --- | --- |
|  | Actual Positive | Actual Negative |
| Predicted Positive | TP | FP |
| Predicted Negative | FN | TN |

表1.混淆矩阵

其中：

TP (True Positive): 模型正确地预测为正类别的样本数。

TN (True Negative): 模型正确地预测为负类别的样本数。

FP (False Positive): 模型错误地将负类别预测为正类别的样本数。

FN (False Negative): 模型错误地将正类别预测为负类别的样本数。

### 2.5.2准确性 (Accuracy)

准确性是模型在所有样本中正确预测的比例，计算公式为：

准确性是一个总体性能指标，但在类别不平衡的情况下可能不够敏感。

### 2.5.3召回率 (Recall)

召回率是指在实际为正类别的样本中，模型正确预测为正类别的比例。计算公式为：

召回率衡量模型对正类别的敏感性。

### 2.5.4精确度 (Precision)

精确度是指在模型预测为正类别的样本中，有多少是真正的正类别。计算公式为：

精确度衡量了模型的预测准确性。

### 2.5.5特异性 (Specificity)

特异性是指在实际为负类别的样本中，模型正确预测为负类别的比例。计算公式为：

特异性衡量了模型对负类别的识别能力。

### 2.5.6 F1分数 (F1 Score)

F1 分数是精确度和召回率的调和平均值，它在处理类别不平衡问题时特别有用。计算公式为：

## 2.6网格搜索

网格搜索是一种通过在指定的超参数空间中系统地搜索来寻找最佳模型参数组合的方法。它的执行步骤大致如下：

①定义超参数空间：首先，需要明确定义要调整的模型超参数的范围。超参数是在模型训练之前需要设置的参数，例如学习速率、树的深度、正则化项等。

②创建参数组合：在超参数空间中，选择一组超参数的组合作为候选。这可以通过穷举法，也就是取每个超参数的可能取值的所有组合，或者通过随机抽样。

③模型训练和评估：对于每个超参数组合，使用训练集来训练模型，并使用验证集或交叉验证来评估模型性能。这可能涉及到多次训练和评估。

④选择最佳组合：根据模型在验证集上的性能，选择表现最好的超参数组合。通常，性能的度量指标可以是准确性、召回率、精确度等，具体取决于问题的性质。

## 2.7交叉验证

交叉验证是一种用于评估模型性能的统计学方法，它通过将数据集划分为训练集和测试集的不同子集，多次训练和测试模型，以减小因随机划分而引入的不确定性。

以下是交叉验证的基本原理：

①数据集划分：将原始数据集分为 k 个子集，其中 k 是交叉验证的折数。通常常用的 k 值是 5 或 10。

②模型训练和测试：对于每一次迭代（折），选择其中 k-1 个子集作为训练集，剩下的 1 个子集作为测试集。使用训练集来训练模型，然后在测试集上评估模型性能。

③重复：重复上述步骤 k 次，每次选择不同的测试集。这样，每个子集都会被用作测试集一次，而每个样本也会在训练集中出现 k-1 次。

④稳定性和泛化：交叉验证提供了对模型性能的更稳定和可靠的估计，因为它考虑了多个不同的训练集和测试集组合，减小了对单次划分的依赖。这对于泛化到未见过的数据的能力的估计更为准确。

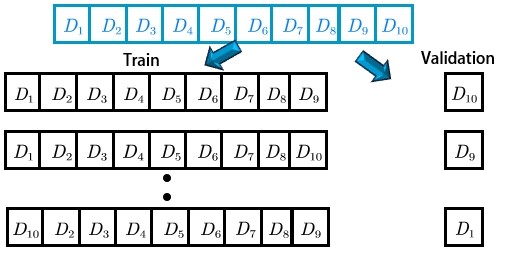


图1.十折交叉

## 2.8集成学习

集成学习是一种机器学习方法，通过结合多个模型的预测来改善整体性能，以期望相对于单个模型更好地泛化到未见过的数据。其基本原理是通过构建多个弱学习器（通常是高度关联但性能略好于随机猜测的模型），并将它们组合成一个强大的模型。

以下是集成学习的主要原理：

①弱学习器的构建：集成学习通常基于弱学习器，这些学习器可以略好于随机猜测，但不需要非常强大。这些弱学习器可以是决策树、线性模型、支持向量机等。

②多样性的引入：为了确保弱学习器之间的多样性，集成方法通常会使用不同的训练子集、不同的特征集，或者在模型构建过程中引入随机性。多样性有助于使得弱学习器在不同方面取得错误，从而提高整体性能。

③集成策略：集成学习采用不同的集成策略来组合弱学习器的预测结果。常见的集成方法包括投票法、平均法、堆叠法等。

投票法：多数投票决定最终的分类结果。可以是硬投票或软投票。

平均法：对多个模型的预测结果取平均值，适用于回归问题。

堆叠法：将多个模型的输出作为输入，训练一个元模型来组合它们的预测。

④降低过拟合风险 集成学习有助于降低过拟合的风险，因为弱学习器可能在训练集上过拟合，但通过组合多个模型，可以提高泛化性能。

⑤适用于不同数据分布：当数据集分布不均匀或者存在噪声时，集成学习能够减少个别模型的影响，从而提高整体的鲁棒性。

常见的集成学习方法包括随机森林（Random Forest）、、AdaBoost、Bagging 等。这些方法在实践中被广泛应用，能够有效提升模型的性能。

# 3.数据探索

## 3.1特征及标签的介绍

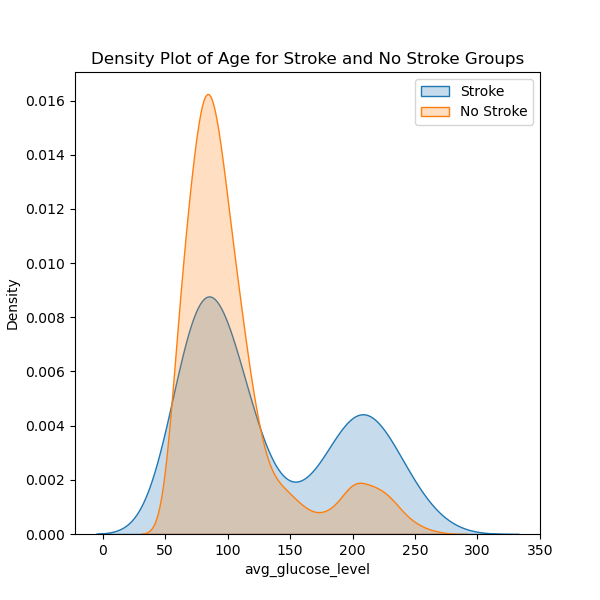
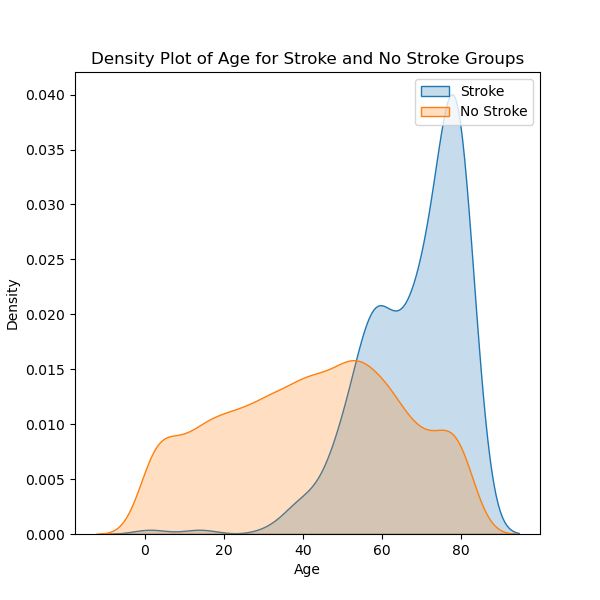
数据集来源于[和鲸社区](https://www.heywhale.com/mw/dataset/648d46ae9de7b81463a94e2b)的中风患者数据集。这个数据集由十个特征和一个标签组成。（见表2）

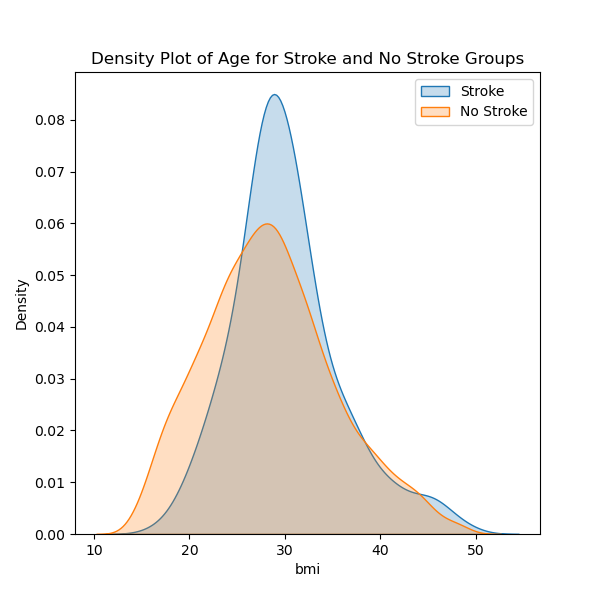
|  |  |  |
| --- | --- | --- |
| 编号 | 名称 | 说明 |
| 1 | gender | 特征：性别，数据类型为对象 |
| 2 | age | 特征：年龄，数据类型为浮点数 |
| 3 | hypertension | 特征：高血压，数据类型为整数（0或1） |
| 4 | heart\_disease | 特征：心脏病，数据类型为整数（0或1） |
| 5 | ever\_married | 特征：是否结过婚，数据类型为对象 |
| 6 | work\_type | 特征：是否工作，数据类型为对象 |
| 7 | Residence\_type | 特征：居住地区的类型，数据类型为对象 |
| 8 | avg\_glucose\_level | 特征：平均血糖水平，数据类型为浮点数 |
| 9 | bmi | 特征：身体质量指数（BMI），数据类型为浮点数 |
| 10 | smoking\_status | 特征：是否抽烟，数据类型为对象 |
| 11 | stroke | 标签：是否中风，数据类型为整数（0或1） |

表2.十个特征和一个标签

## 3.2密度图

经过统计，我们发现数据集并没有缺失值，因此我们不用进行缺失值的相关处理。根据表2，我们可以知道数据类型为浮点的特征有age、avg\_glucose\_level和bmi。因此，我们可以通过绘制它们的密度图来初步探索它们与中风之间的关系。

（a）年龄的密度图 （b）血糖水平的密度图



（c）bmi的密度图

图2

结合密度图，我们可以初步得出以下三个结论：

①中风人群在高龄人群中占比较高。低龄人群虽也有中风的案例，但是案例较少。（结合图2（a））

②中风人群的血糖水平分布范围较广，且相较于未中风人群，中风人群处于正常血糖水平的比例较低。已知正常空腹血糖浓度的预期值介于 70 mg/dL 到 100 mg/dL 之间。（结合图2（b））

③结合表3和图2（c）可知，中风人群处于正常bmi水平的比例相较于未中风人群要更低。

|  |  |
| --- | --- |
| BMI | 身体质量指数说明 |
| < 18.5 | 体重过轻 |
| 18.5 - 23.9 | 体重正常 |
| 24 - 27.9 | 超重 |
| > 28 | 肥胖 |

表3.BMI分类

## 3.3热力图

热力图是一种用颜色变化来表示矩阵数据的可视化技术。它通过使用不同颜色的方块或矩形来表示矩阵中的数值，使人们能够直观地看到数据之间的关系和模式。在统计学和数据分析中，热力图通常用于展示变量之间的相关性或距离。因此，我们可以通过热力图来初步分析各个特征之间的关系。（见图3）

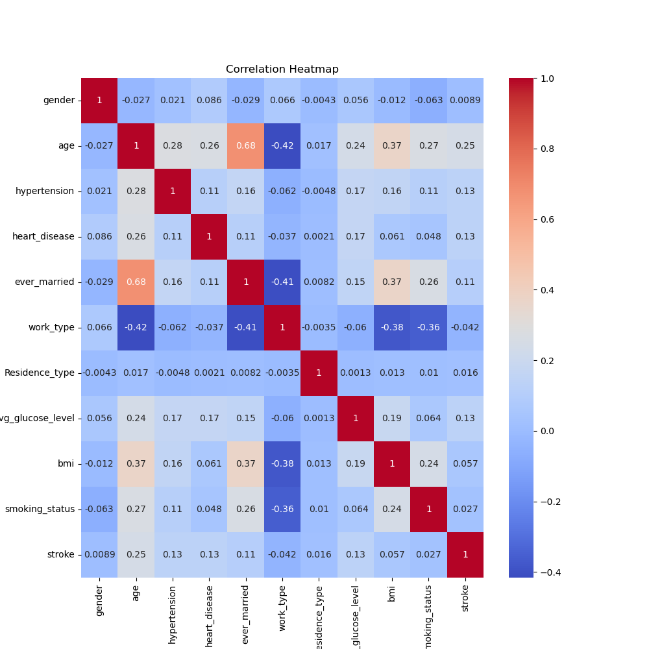


图3.热力图

通过绘制的热力图（图3），我们发现stroke标签与work\_type特征为负相关，和其余9个特征为正相关。

# 4.建模过程与结果

## 4.1逻辑回归

通过自定义逻辑回归的参数及其选择值（表4），我们可以使用网格搜索来找到一组较好的参数组合。通过绘制折线图（图4）我们可以发现参数组合为{'C': 0.01, 'class\_weight': 'balanced', 'max\_iter': 100}。有了较好的参数组合，我们可以训练出较好的模型。最终，我们通过较好的模型得出的评分结果如下：

①Accuracy: 0.75

②Recall: 0.7924528301886793

③Precision: 0.75

④Specificity: 0.7021276595744681

⑤f1: 0.7706422018348623

⑥conf\_matrix [[33 14]

[11 42]]

|  |  |  |
| --- | --- | --- |
| 超参数 | 超参数的选择值 | 超参数的含义 |
| C | [0.001, 0.01, 0.1, 1, 10, 100] | 正则化强度的倒数 |
| max\_iter | [100, 200, 300] | 最大迭代次数 |
| class\_weight | ['balanced'] | 类别权重 |

表4.逻辑回归的参数与选择值

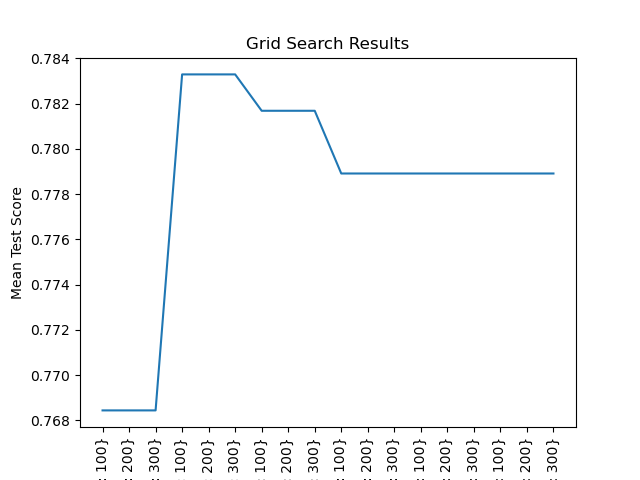


图4.逻辑回归折线图

## 4.2线性SVM

通过自定义线性SVM的参数及其选择值（表5），我们可以使用网格搜索来找到一组较好的参数组合。通过绘制折线图（图5）我们可以发现参数组合为{'C': 100, 'loss': 'hinge'}。有了较好的参数组合，我们可以训练出较好的模型。最终，我们通过较好的模型得出的评分结果如下：

①Accuracy: 0.72

②Recall: 0.7547169811320755

③Precision: 0.7272727272727273

④Specificity: 0.6808510638297872

⑤f1: 0.7407407407407407

⑥conf\_matrix [[32 15]

[13 40]]

|  |  |  |
| --- | --- | --- |
| 超参数 | 超参数的选择值 | 超参数的含义 |
| C | [100, 500, 1000, 2500, 5000, 7500, 10000] | 正则化参数 |
| loss | ['hinge', 'squared\_hinge'] | 损失函数 |

表5. 线性SVM的参数与选择值

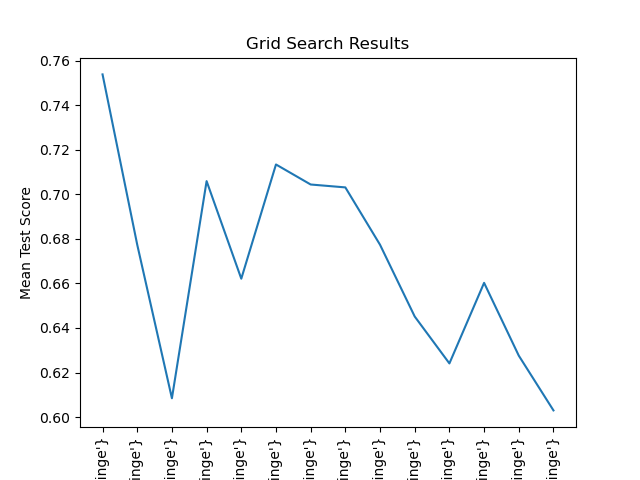


图5.线性SVM折线图

## 4.3非线性SVM

通过自定义非线性SVM的参数及其选择值（表6），我们可以使用网格搜索来找到一组较好的参数组合。通过绘制折线图（图6）我们可以发现参数组合为{'C': 50, 'coef0': 0, 'degree': 3}。有了较好的参数组合，我们可以训练出较好的模型。最终，我们通过较好的模型得出的评分结果如下：

①Accuracy: 0.66

②Recall: 0.6415094339622641

③Precision: 0.6938775510204082

④Specificity: 0.6808510638297872

⑤f1: 0.6666666666666666

⑥conf\_matrix [[32 15]

[19 34]]

|  |  |  |
| --- | --- | --- |
| 超参数 | 超参数的选择值 | 超参数的含义 |
| degree | [3, 4, 5] | 多项式核函数的次数 |
| coef0 | [0, 1, 2] | 'poly' 和 'sigmoid' 核的独立项 |
| C | [50, 75, 100] | 正则化参数 |

表6. 非线性SVM的参数与选择值

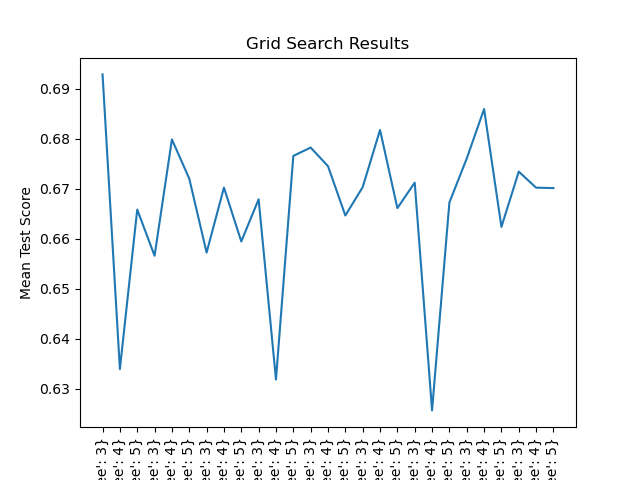


图6.非线性SVM折线图

## 4.4决策树

通过自定义决策树的参数及其选择值（表7），我们可以使用网格搜索来找到一组较好的参数组合。通过绘制折线图（图7）我们可以发现参数组合为{'max\_depth': 15, 'max\_leaf\_nodes': 50}。有了较好的参数组合，我们可以训练出较好的模型。最终，我们通过较好的模型得出的评分结果如下：

①Accuracy: 0.7

②Recall: 0.6792452830188679

③Precision: 0.7346938775510204

④Specificity: 0.723404255319149

⑤f1: 0.7058823529411764

⑥conf\_matrix [[34 13]

[17 36]]

|  |  |  |
| --- | --- | --- |
| 超参数 | 超参数的选择值 | 超参数的含义 |
| max\_depth | [10, 15, 20, 25, 30] | 最大深度 |
| max\_leaf\_nodes | [50, 100, 200, 500] | 最大叶子节点数 |

表7. 决策树的参数与选择值

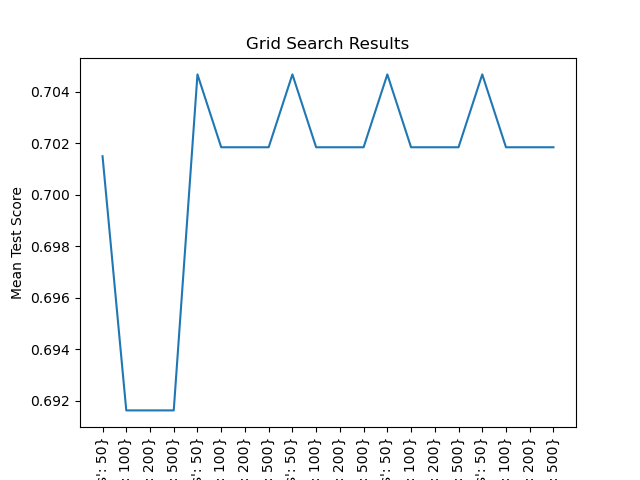


图7.决策树折线图

此外，通过开源图形可视化软件包graphviz，我们可以将我们的决策树模型可视化展示出来（图8）。

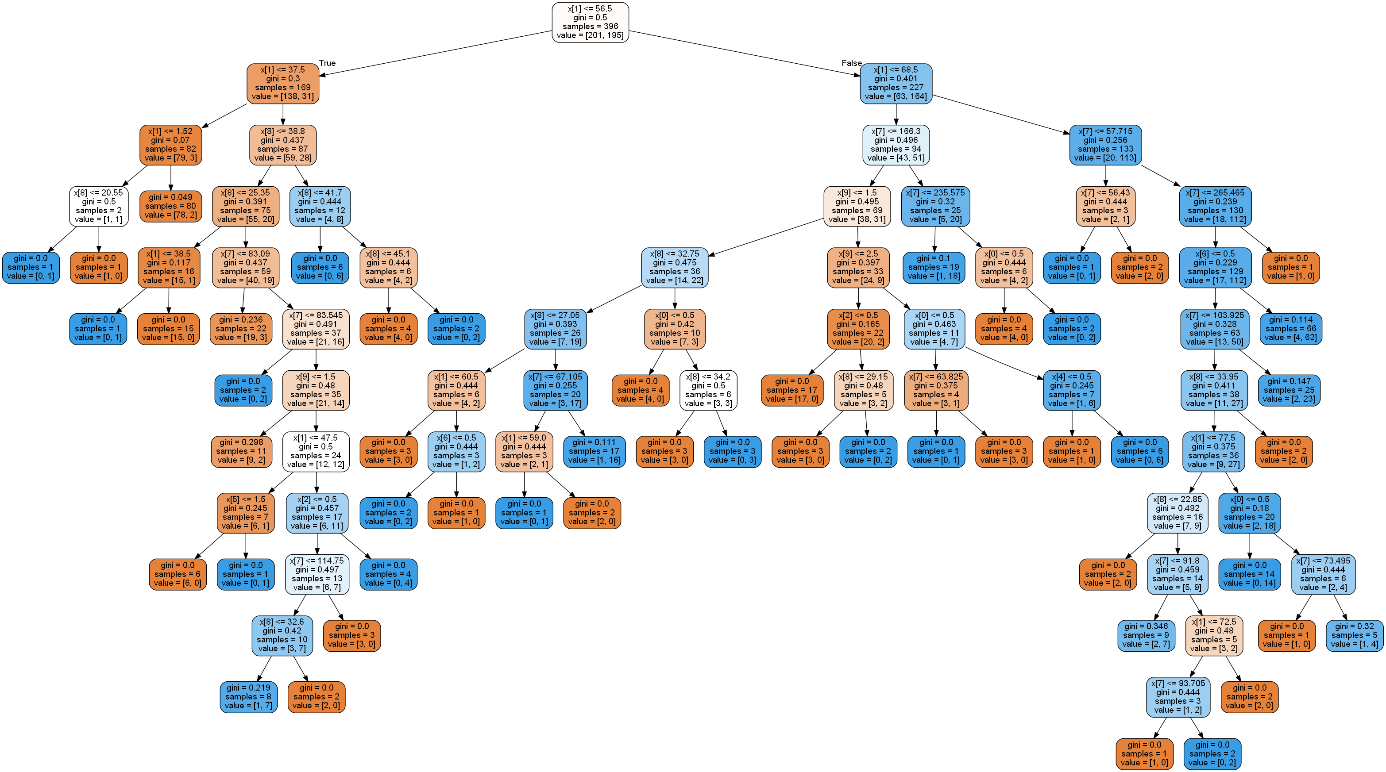


图8.可视化的决策树模型

## 4.5随机森林

通过自定义随机森林的参数及其选择值（表8），我们可以使用网格搜索来找到一组较好的参数组合。通过绘制折线图（图9）我们可以发现参数组合为{'max\_features': 'sqrt', 'n\_estimators': 5}。有了较好的参数组合，我们可以训练出较好的模型。最终，我们通过较好的模型得出的评分结果如下：

①Accuracy: 0.71

②Recall: 0.7735849056603774

③Precision: 0.7068965517241379

④Specificity: 0.6382978723404256

⑤f1: 0.7387387387387387

⑥conf\_matrix [[30 17]

[12 41]]

|  |  |  |
| --- | --- | --- |
| 超参数 | 超参数的选择值 | 超参数的含义 |
| n\_estimators | [1, 2, 5, 10, 15, 20] | 森林中树的数量 |
| max\_features | ['sqrt', 'log2'] | 每次划分时考虑的最大特征数 |

表8. 随机森林的参数与选择值

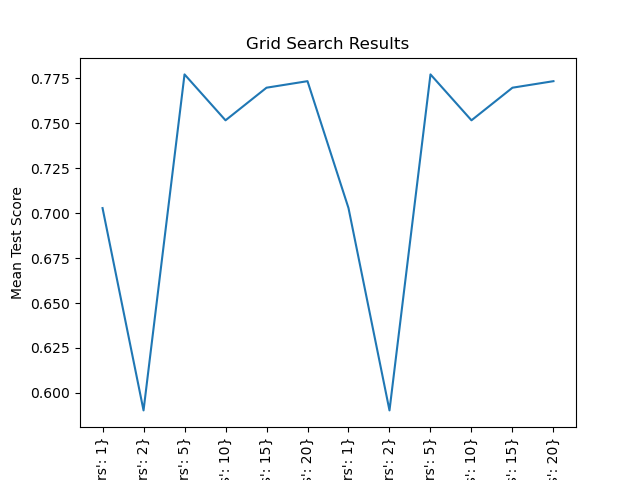


图9.随机森林折线图

# 5.总结

在本次研究中，我使用的数据集是一个有偏数据集。即中风样本数目远远小于未中风的样本数目。起初得到的效果并不好，因此我尝试了随机补丁、随机子空间和欠采样的方法。最终，我发现随机补丁和随机子空间的提升效果没有欠采样的效果理想，因此我最后使用欠采样的方法来处理数据集。

经过对各个模型的参数调优和在测试集上的预测，我们最终得到了以下的可视化结果：

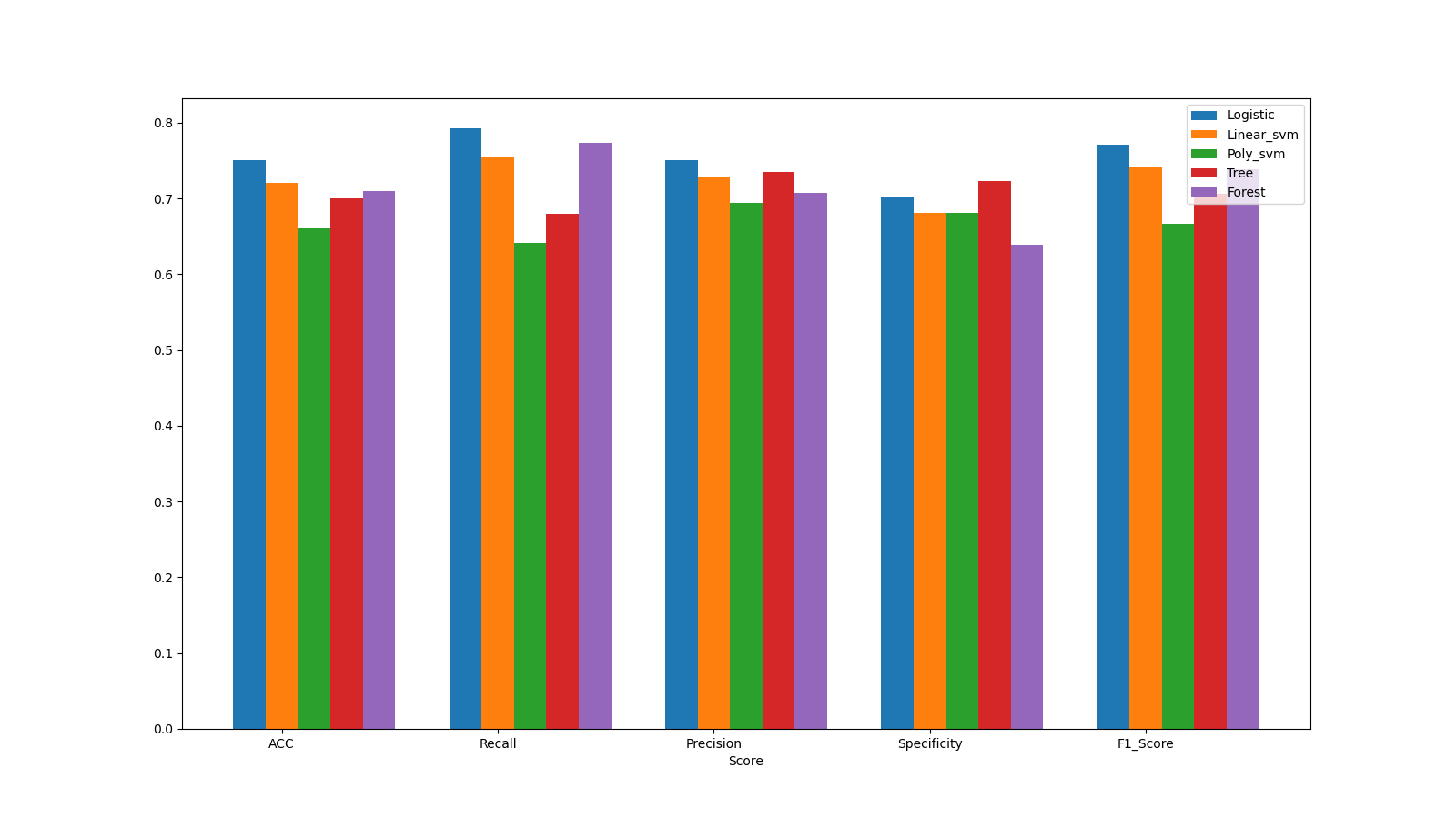


图10.可视化结果

我们可以清晰地观察到逻辑回归模型在除了特异性这一评价标准得分上是略低于决策树模型之外，其他的评价标准得分都是最高的。因此，我们将选取逻辑回归模型作为我们的最终模型。我认为逻辑回归模型的优势可能归因于其简单性和对线性关系的适应性。在各个评分标准下，逻辑回归的得分比较稳定。这对于解释模型的预测结果和为临床实践提供指导具有重要意义。

尽管其他模型如支持向量机、决策树和随机森林在某些方面表现良好，但逻辑回归模型在综合性能和可解释性上的优势使其成为脑中风预测的一个可行选择。

这一研究为脑中风预测提供了关于模型选择和应用的深入洞察，为未来的相关研究和医学实践提供了有益的建议。

参考文献：

[1] Smith, John A. "The use of machine learning methods in classification of pumpkin seeds." Journal of Agricultural Science, vol. 10, no. 3, Year, pp. 123-145.

[2]刘泽文. 基于机器学习的脑卒中复发预测模型研究. Diss. 湖南大学.

[3]饶夫阳等. "基于机器学习模型缺血性脑卒中1年死亡预测效果评价." 中国公共卫生 35.9(2019):5.