
第九届湖南省研究生数学建模竞赛承诺书

我们仔细阅读了湖南省高校研究生数学建模竞赛的竞赛规则。

我们完全明白，在竞赛开始后参赛队员不能以任何方式（包括电话、电子邮件、网上咨询等）与队外的任何人（包括指导教师）研究、讨论与赛题有关的问题。

我们知道，抄袭别人的成果是违反竞赛规则的，如果引用别人的成果或其他公开的资料（包括网上查到的资料），必须按照规定的参考文献的表述方式在正文引用处和参考文献中明确列出。

我们完全清楚，在竞赛中必须合法合规地使用文献资料和软件工具，不能有任何侵犯知识产权的行为。否则我们将失去评奖资格，并可能受到严肃处理。

我们郑重承诺，严格遵守竞赛规则，以保证竞赛的公正、公平性。如有违反竞赛规则的行为，我们将受到严肃处理。

我们授权湖南省研究生数学建模竞赛组委会，可将我们的论文以任何形式进行公开展示（包括进行网上公示，在书籍、期刊和其他媒体进行正式或非正式发表等）。

所属学校和学院（请填写完整的全名）：国防科技大学空天科学学院

参赛队员（打印后签名）：1. 姜瑞祺

2. 刘凌波

3. 徐子夜

指导教师或指导教师组负责人（打印后签名）：陈全

日期：2024 年 7 月 5 日

（请勿改动此页内容和格式。以上内容请仔细核对，如填写错误，论文可能被取消评奖资格。）

基于随机森林的数据分类模型研究

摘要

智能手机能够通过内置的运动传感器来对使用者日常生活的活动数据进行记录，通过对活动数据的分类解析可以计算出其日常热量消耗。本文针对活动数据分类问题建立了决策树和随机森林模型，解决了活动数据到活动状态的映射问题，并利用同样的算法模型探究人体年龄、身高等要素对所记录活动数据的影响。

针对问题一：本文建立了决策树模型。首先假定采样频率内加速度大小不变对附件 1 中所给加速度数据进行积分得到对应的速度，利用速度和和数据其他特征作为判别条件带入决策树模型进行分类。分类结果见表 1。

针对问题二：本文建立了随机森林模型。首先以问题 1 中建立的决策树模型为蓝本，建立包含 100 棵相同决策树的随机森林模型，每颗决策树依据判别条件对数据进行分类，最终已投票方式决定最后的分类结果。以附件 2 中所给的数据为训练集进行训练，再抽取 30% 的数据作为测试集对模型测试。最终得出问题 1 中建立模型对附件 2 中数据进行测试准确率为 63.5%，而问题 2 中建立的模型准确率为 95.297%。

依据问题 2 提出的随机森林模型对附件 3 中数据的分类结果见表 2。

针对问题三：本文建立了随机森林模型。对于探究年龄、身高、体重等因素对实验人员所测数据是否有影响，我们将 3 种因素与 12 种活动一一对应，并将这三种因素与附件所给出数据的直接联系，转变为三种因素与特征值之间的联系，并将这种联系绘制为 36 张图展示在支撑材料中。

对于实验人员身份的确定，问题 3 种采用与问题 2 种同样的随机森林模型，不同之处在于问题 2 种的模型是对 12 种活动的预测，而问题 3 中是对人员身份的预测。利用附件 2 中所给数据进行模型测试，得出模型分类正确率为 51.15%。

人员身份的预测结果见表 3。

最后，我们对提出的模型进行全面的评价：本文提出的决策树和随机森林模型广泛应用于分类问题，因而也适用于本题，在问题 1 和问题 2 中表现出色，贴合实际，能合理解决提出的问题，具有实用性强，算法效率高等特点。但是在问题 3 中表现一般，其原因在于选取的特征重要性不高，因而算法能力不强。

关键词：决策树 随机森林 分类

目录

基于随机森林的数据分类模型研究	II
摘要	II
1 问题综述	1
1.1 问题背景	1
1.2 问题提出	1
1.3 资料条件	1
2 模型假设与符号说明	2
2.1 模型基本假设	2
2.2 符号说明	2
3 问题分析	2
3.1 问题一的分析	2
3.2 问题二的分析	2
3.3 问题三的分析	3
4 模型 1 的建立与求解	4
4.1 问题 1 的模型建立与求解	4
4.1.1 模型准备	4
4.1.2 模型建立	4
4.1.3 模型求解	5
5 问题 2 的模型建立与求解	7
5.1 模型准备	7
5.2 模型建立	7
5.3 模型求解	8
6 问题 3 的模型建立与求解	10
6.1 模型建立	10
6.2 模型的求解	11
7 模型评价与改进	15
7.1 模型的优点	15
7.2 模型的不足	15
7.3 模型的改进	15
参考文献	15
附 录	16
附录 A: 主要程序/关键代码	16

1 问题综述

1.1 问题背景

目前人们使用的智能手机大多具有计算人体日常活动所消耗热量的功能，这种功能的实现主要依赖于手机软件对日常活动数据的收集，通过对使用者每天步行、骑行、爬高等活动持续时间、剧烈程度等因素分析得到使用者当天所消耗的热量值。

这种功能能够实现的关键在于手机内置的软硬件能够完整的收集使用者的活动数据并对其进行正确的分类，运动传感器即加速度计和陀螺仪就能够完成数据收集任务。加速度计能够记录手机在三个轴向的加速度变化情况，而陀螺仪能够记录手机绕三根轴的角速度变化情况。为了解释每组加速度和角速度背后表示的行为活动含义，需要建立数学模型来对数据进行分类识别，并确定当前使用者的活动状态。

1.2 问题提出

针对使用者活动状态分类以及使用者不同年龄下相同活动状态所对应的数据是否相同的问题，题目给出了一些携带传感器的实验人员经测试得到的 12 种日常活动状态数据，针对给出的这些数据解决一些以下 3 个问题：

- (1) 问题 1: 题目给出了三名实验人员每种运动状态的五组加速度和角速度的实验数据，但是这些数据并没有给出实验人员具体对应的活动状态，需要提出数学模型对这些数据进行分类，并给出分类结果。
- (2) 问题 2: 题目给出了 10 名试验人员 12 种活动状态的数据，每种活动状态有 5 组实验数据。与问题一中不同的是本问题中的活动状态均已被标记。现要求利用这些被标记的数据建立分类模型并解决以下两个问题：
 - 在忽略本题中标记数据的前提下利用建立的模型对其进行重新分类，并将分类得到的结果与并得到分类结果与问题一中的分类结果进行比较，分析建立模型针对不同活动状态的准确率。
 - 利用提出的判别模型对给出的试验人员的 30 次活动状态进行分类，并给出分类结果。
- (3) 问题 3: 本题给出了参与以上两个问题的 13 名实验人员的身体状态，包括年龄、身高、体重等数据。要求利用这些信息分析不同身体状态的实验人员在同一活动状态下得到的数据是否有差异，并利用分析结果给活动人员进行画像。同时在问题 2 中随机挑选 5 名实验人员给出其活动状态数据，要求判断这 5 位实验人员的身份。

1.3 资料条件

附件【1】提供了未作标记的三名实验人员的运动数据。

附件【2】提供了做出标记的十名实验人员的运动数据。

附件【3】提供了某一特定试验人员的 30 组运动数据。

附件【4】提供了问题一、二中参加试验的十三名实验人员的身体状态数据。

附件【5】提供了问题二中随机五位成员的运动数据。

2 模型假设与符号说明

2.1 模型基本假设

- (1) 假设在采样时间间隔内采集的加速度和角速度大小不变。
- (2) 假设实验人员身体状态对所采数据无影响。
- (3) 假设所用决策树的判别条件合适。

2.2 符号说明

本文定义了如下【数字】个使用次数较多的符号，其余符号在使用时注明。

表 1 符号说明

符号	含义	单位
$H(x)$	组合分类模型	/
h_i	单颗决策树	/
$I(\cdot)$	指示性函数	/
$\max$	取最大值	/
Y	输出函数	/
i	采样频率	Hz

3 问题分析

3.1 问题一的分析

本题以手机计算评估人体热量消耗为背景，介绍了人体活动数据的测量方法。要求利用附件一中给出的三名实验人员的活动数据特征进行分类，要求将每组数据对应的活动特征利用数学模型求解出来。

本文通过对所给出的数据进行观察分析，决定用积分法和决策树模型作为解决该问题的关键，对所给数据进行分类。对题目所给加速度数据进行一次积分，得到实验人员的速度大小，对加速度进行二次积分得到实验人员给定时间内为一大小，对题目所给的角速度进行一次积分得到转动方向大小。依据积分所得结果结合决策树对相应生活活动的判别条件共同得出分类结果。

3.2 问题二的分析

题目给出了 10 名实验人员的活动数据，并且标记了这些数据对应的活动状态，要求提取这些活动的特征，利用提取的特征建立判别模型，并实用建立的判别模型完成以下两个工作：

首先用建立的模型来重新给附件 2 中所给的活动数据进行分类，并将分类结果与问题一中分类结果作比较，分析二者的不同之处。

其次，将附件 3 中给出的三十组数据进行分类，并给出分类结果。

针对本问题，关键在于模型的建立。依据题目意思可知本题应该归类为分类问题，对于分类问题，本文采用随机森林模型来求解，随机抽取附件给出的一部分数据作为测

试数据集，在随机森林模型里的每颗决策树都按照问题 1 中建立的模型来进行分类，最终对所有决策树得到的分类结果进行投票，投票结果即为最终分类结果。

3.3 问题三的分析

题目提出了新的问题，将年龄、身高、体重三个因素引入到问题中来，力图探究实验人员的活动数据是否与以上三种因素有关。针对该问题，本文用两个步骤将题目所给的活动数据和这三种因素建立联系。第一步将年龄、身高、体重和 12 中活动状态中的每一个活动状态都建立联系。由于问题 1 和问题 2 中已经归类整理了可以区分这 12 种活动状态的特征值，而这些特征值都是依据附件中所给出的数据得来的，因此只要将年龄、身高、体重三个因素和这些特征值相联系就能够得到其与活动数据之间是否存在关联。

4 模型 1 的建立与求解

4.1 问题 1 的模型建立与求解

4.1.1 模型准备

通过对问题的分析和对附件 1 文件数据的观察，针对此问题判定应该选用决策树模型来求解。

本文首先利用决策树模型来对前 45 组数据进行判别分类，对剩余的 15 组数据利用积分来确定其判别条件。决策树求解的关键在于判别条件的选取，本文一部分直观的给出了判别条件，并依据这些条件对数据进行分类。决策树可以以树形数据结构来展示决策规则和树状模型，作为一种归纳学习算法，其重点是将看似无序、杂乱的已知实例，通过某种技术手段将它们转化成可以预测未知实例的树状模型，每一条从根结点（对最终分类结果贡献最大的属性）到叶子结点（最终分类结果）的路径都代表一条决策的规则^[1]。

4.1.2 模型建立

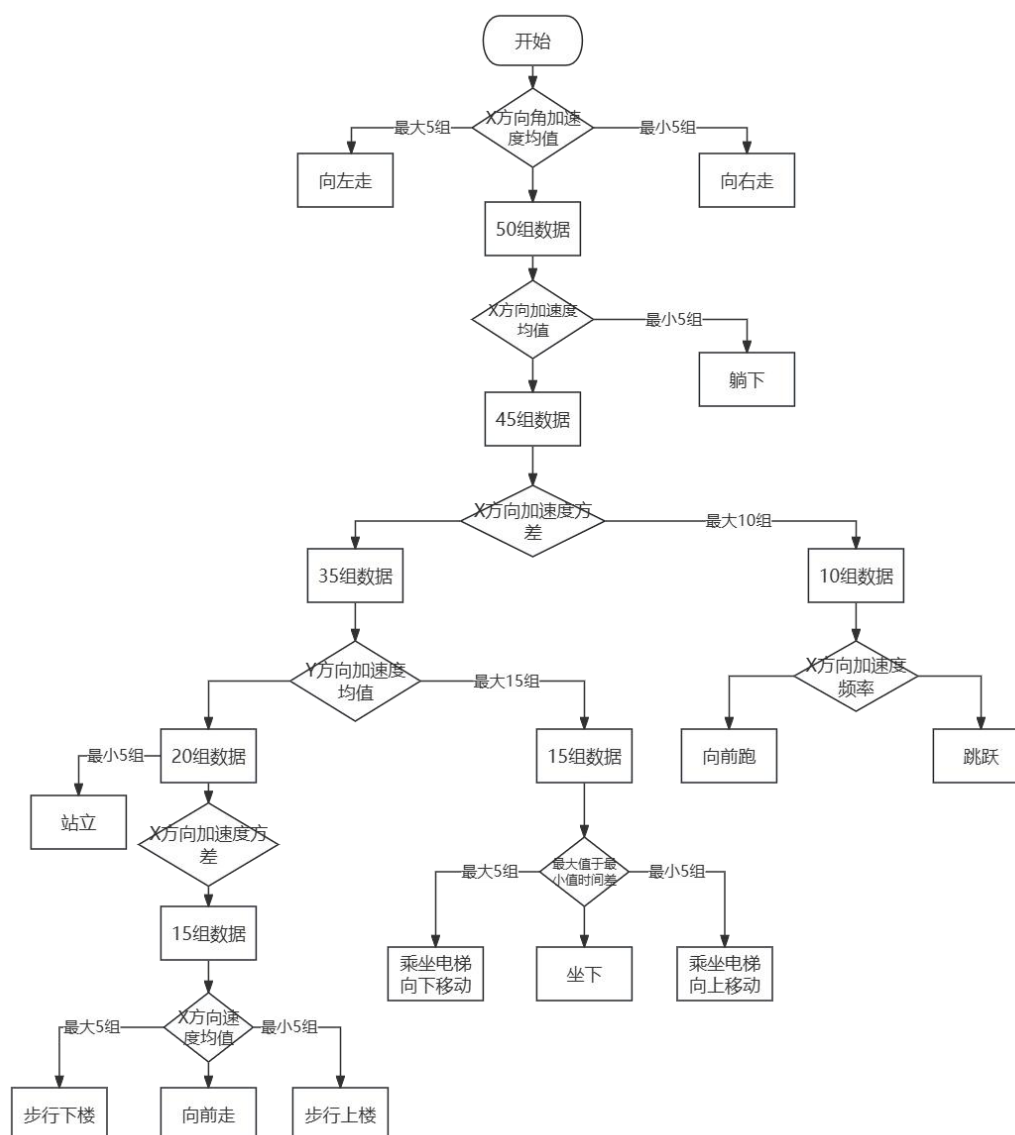


图 1 决策流程图

本题的模型建立基于上述决策流程图，其基本步骤如下所示：

Step 1: 以 X 方向上角加速度均值作为判别条件，可将均值最大和均值最小的五组数据完成分类，根据生活经验，其活动分别为向左走和向右走。剩余的五十组数据进行下一轮分类判别。

Step 2: 以 X 方向上加速度均值作为判别条件，可将均值最小的五组数据完成分类，根据一般生活经验，该组数据对应的活动为躺下。剩余的 45 组数据进行下一轮判别。

Step 3: 以 X 方向上加速度方差作为判别条件，将方差最大的十组数据和剩余的三十五组数据分开，分别进行下一轮判别。

Step 4: 对 X 方向加速度方差最大的十组数据进行分类，分类的判别条件为 X 方向的加速度频率，根据频率差异可将数据分类为向前跑和跳跃两类活动状态。

Step 5: 对剩余的 35 组数据进行分类，分类的判别条件为 Y 方向上的加速度均值，将其分为加速度均值最大的 15 组和另外 20 组数据，分别进行下一轮判别。

Step 6: 对于 Y 方向加速度最大的 15 组数据分类的判别条件为数据处于最大值和处于最小值间的时间间隔，间隔最小的五组数据分类为乘坐电梯向上移动，间隔最大的五组数据分类为乘坐电梯向下移动，其余五组数据分类为坐下。

Step 7: 对于剩下的 20 组数据进行分类，分类的判别条件为 X 方向加速度方差，方差最小的五组数据为站立。

对于剩下的十五组数据的判别条件本文采用积分的形式来确定，通过对 X 方向上加速度进行积分得到对应数据的速度大小，以速度大小对数据对应的活动状态进行分类。若已积分形式进行求解，则需满足在采样频率内加速度大小不变，以采样点采集的数据作为被积函数。

根据数学方法有：

$$f(x) = \sum_{1}^n \int_0^i d(x)$$

其中：

$f(x)$ 是关于 x 的函数，指实验人员在相应坐标上的速度。

$d(x)$ 指所在采样点的加速度大小，假定其在采样频率内数值大小不变。

n 指累加数目，在本文中为每种运动状态的组数。

i 指采样频率。

根据上式可得出剩余 15 组数据的在 X 方向上的速度大小，即可进行下一步分类。

Step 8: 对剩余的十五组数据进行分类，其行为活动的判别依据为在 X 方向上的速度大小，速度最大的数据其行为活动分类为步行下楼，速度最小的数据其行为活动分类为步行上楼，速度居中的数据其行为活动分类为向前走。

4.1.3 模型求解

依据上述步骤即可利用决策树模型对所给出的 60 组数据对应的行为活动进行分类，以下展示分类结果：

表 1: 问题 1 结果

分类	Person1	Person2	Person3
第 1 类	SY21,SY29	SY3.SY20	SY8,SY23
	SY48,SY53.SY56	SY38,SY51,SY60	SY39,SY42,SY52
第 2 类	SY14,SY17	SY16 ,SY21	SY9,SY14
	SY31,SY37,SY52	SY29,SY40,SY43	SY30,SY37,SY43
第 3 类	SY22,SY32	SY31,SY32	SY3,SY32
	SY34,SY35,SY40	SY39,SY52,SY59	SY50,SY54,SY55
第 4 类	SY15,SY23	SY2,SY10	SY4,SY5
	SY27,	SY13,SY26,SY47	SY10,SY41,SY53
第 5 类	SY19,SY26	SY1,SY4	SY18,SY22
	SY43,SY44,SY50	SY48,SY49,SY50	SY57,SY58,SY59
第 6 类	SY3,SY6	SY8,SY11	SY1,SY6
	SY11,SY16,SY55	SY22,SY55,SY56	SY17,SY21,SY33
第 7 类	SY4,SY13	SY5,SY9	SY7,SY15
	SY25,SY30,SY57	SY36,SY41,SY54	SY26,SY40,SY45
第 8 类	SY12,SY45	SY14,SY17	SY16,SY20
	SY51,SY54,SY60	SY18,SY45,SY53	SY28,SY38,SY46
第 9 类	SY9,SY10	SY6,SY15	SY2,SY12
	SY18,SY24,SY39	SY25,SY32,SY58	SY34,SY47,SY48
第 10 类	SY5,SY8	SY12,SY23	SY13,SY19
	SY42,SY47,SY58	SY28,SY37,SY46	SY36,SY49,SY60
第 11 类	SY2,SY20	SY19,SY27	SY11,SY24
	SY33,SY41,SY59	SY30,SY33,SY44	SY44,SY51,SY56
第 12 类	SY1,SY7	SY7,SY24	SY25,SY27
	SY28,SY36,SY38	SY34,SY42,SY57	SY29,SY31,SY35

5 问题 2 的模型建立与求解

5.1 模型准备

针对本题所提出要求，将其归类为分类问题。本文选用随机森林模型来解决这一问题。要建立随机森林模型就要对决策树进行阐释，下面首先介绍本题决策树的建立。

决策树作为一种单分类器，可以通过对数据进行建模来得到。其结构为树状结构，具有可快速分类，并能够多次使用的优点。其工作原理为在根节点出依据信息增益最大原则选择出判别条件，从而分裂出属性特征不同的两个叶节点。叶节点重复同样的步骤，直到所得叶节点满足分类要求为止^[2]。

由于决策树是一种单一的分类器模式，它得到的解往往是局部最优而非全局最优，而且训练时容易出现过拟合的情况，为了解决这些问题，在此基础上提出了随机森林模型。随机森林是综合考虑多个决策树而形成的一种集成分类器模型，它不仅用于分类还可以解决回归问题^[3]。随机森林的投票决策过程如下：

$$H(x) = \arg \max_y \sum_{i=1}^k I(h_i(x) = Y)$$

其中：

$H(x)$ 表示组合分类模型。

h_i 表示单棵决策树。

Y 为输出变量。

$I(\cdot)$ 为指示性函数。

算法根据最大投票法则判断得票数最多的一类作为最后的分类结果。

5.2 模型建立

针对问题 2，本文提出了如下模型：

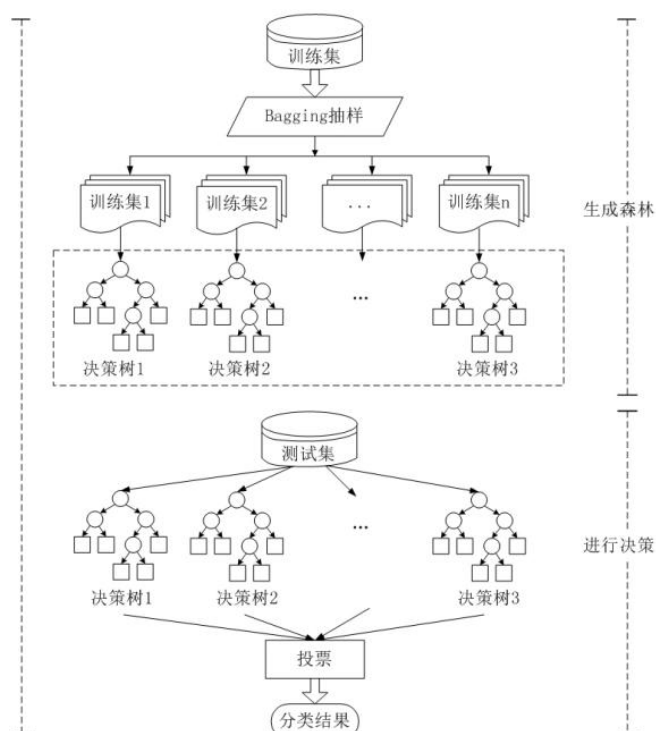


图 2 随机森林模型示意图

该模型建立共分为两部分，其工作原理如下：

生成森林：

在对测试数据集进行测试之前首先要进行模型的训练，本文选取 100 组数据集生成 100 组决策树，每个训练集中包含所有的十名实验人员所测得的活动数据。对于每组训练集对应的决策树均与问题一中提出的模型一致。其特征数据分别为：X 方向角加速度平均值、X 方向加速度平均值、X 方向加速度方差、X 方向加速度频率、Y 方向加速度平均值、X 方向加速度极差、X 方向 1s 内加速度最大值与最小值的时间差和 X 方向速度平均值，每次选择其中一个特征来进行分类，直至所有数据均已完成分类，其中每个特征重要性分别为：1.1546、0.72816、0.91181、0.84348、1.4087、0.88054、0.58269。每棵决策树都遵从上述分类步骤。

进行决策：

在所有十名试验人员的数据集中随机选取 30% 作为测试数据集，对测试数据集利用上述训练完成的模型进行测试，模型中同样包含 100 棵决策树，选取的特征数和训练集一致，分类方法一致。最终对 100 棵树的分类结果进行投票，投票得出的结果即为最终数据对应的活动状态。

5.3 模型求解

(1) 利用问题 1 中的模型对附件 2 中数据进行重新分类，分类得到的结果与附件 2 进行比对的到的准确率为 63.5%。

利用问题 2 中建立的随机森林模型对该数据进行分类得到了如下结果：

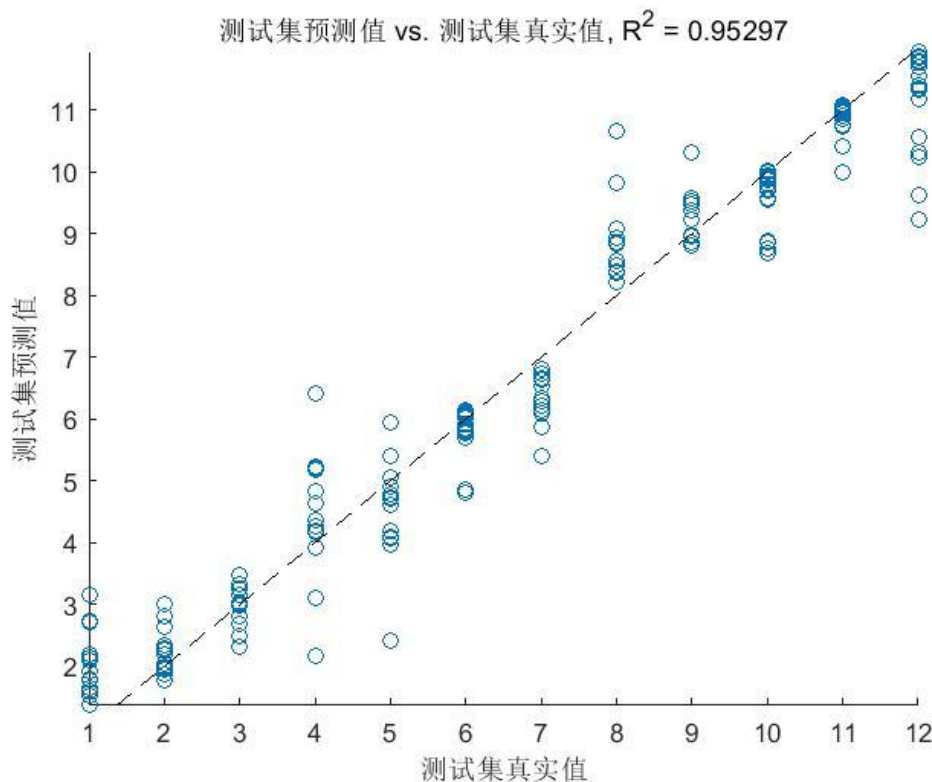


图 3 模型准确率

通过将模型预测结果和附件 2 中标记的结果进行一一比对，得到如图 3 所示的对比图。其中 R^2 为 0.95297，表明模型预测准确程度已经十分接近测试集的真实值，且明显高于利用第一问中所建立的决策树的模型准确率。

原因分析：

因为随机森林选取了部分数据建立了多个决策树，即使有个别决策树会因为异常值的影响导致预测不准确，但预测结果是参考多个决策树得到的结果，降低了异常值带来的影响，因此随机森林模型所做的分类准确率较单棵决策树更高。

(2) 附件 3 中分类结果如下表所示

表 2：问题 2 结果

活动类型	判别状态
SY1	6
SY2	2
SY3	6
SY4	9
SY5	6
SY6	10
SY7	2
SY8	6
SY9	6
SY10	10
SY11	10
SY12	6
SY13	4
SY14	3
SY15	3
SY16	2
SY17	4

活动类型	判别状态
SY18	6
SY19	8
SY20	9
SY21	6
SY22	2
SY23	9
SY24	6
SY25	2
SY26	10
SY27	8
SY28	6
SY29	6
SY30	6

6 问题 3 的模型建立与求解

6.1 模型建立

对于问题 3，本文首先要找到年龄、身高、体重三种特征和题中所给数据中存在的联系。为了建立这种联系本文分两部分建立映射关系。第一步是年龄、身高、体重三种特征和 12 种活动状态一一连线，第二步是在这种关系中将问题 1 和问题 2 种归类整理的特征值随年龄、身高、体重的变化表示出来，将其与活动数据之间的联系转化为与特征值之间的联系。

针对本题所建立的模型依然为随机森林模型，和问题 2 的区别在于模型预测的动作转换为预测实验人员的身份。我们首先对模型进行训练，训练集的数据为附件 2 中 10 人的已经被标记过的数据，判别条件同样为 X 方向角加速度平均值、X 方向加速度平均值、X 方向加速度方差、X 方向加速度频率、Y 方向加速度平均值、X 方向加速度极差、X 方向 1s 内加速度最大值与最小值的时间差和 X 方向速度平均值。模型的建立仍然有生成森林和进行决策两部分。

生成森林：

利用附件 2 中进行标记的十名实验人员的活动数据进行训练，找到 12 种活动状态数据和具体人员之间的对应关系。

进行决策：

将附件 5 中数据作为模型测试集，每一棵决策树均根据以上八种判别条件进行判别分类，分裂结果为附件 2 中人员的身份。最终人员身份的确定由全体决策树共同投票决定，显示投票率高的即为预测结果。

对于建立的模型利用附件 2 中数据进行测试得到准确率如下图所示：

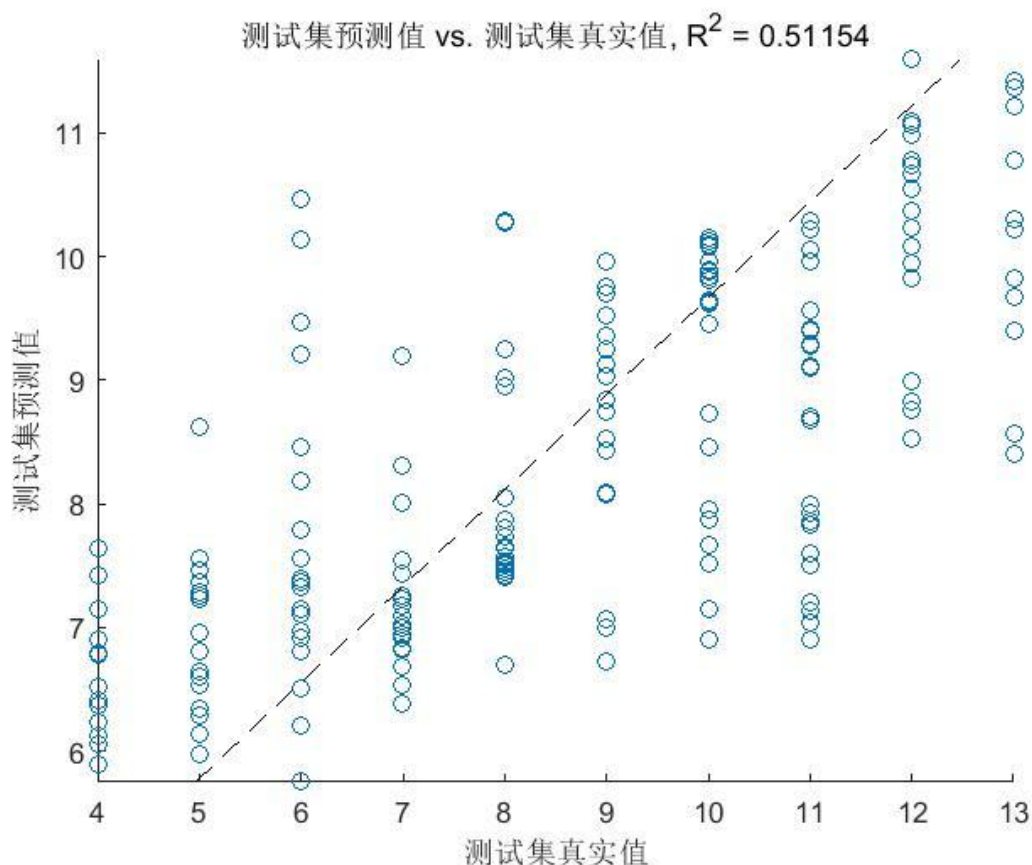


图 4 模型准确率

6.2 模型的求解

本题中年龄、身高、体重因素和活动数据之间的关系被总结为 36 张图，现取三种要素中具有代表性的图片进行解释，其余图片见支撑材料。

图片的横坐标表示年龄、身高或体重中的一种因素，纵坐标为问题 1 中找到的八种特征值归一化后的数值。

可见图中共有八条折线，每一条折线对应其中一种特征值，折线的变化趋势即为特征值随三种要素变化的趋势，同时也可以间接反映三种要素和 12 中活动数据之间的联系。

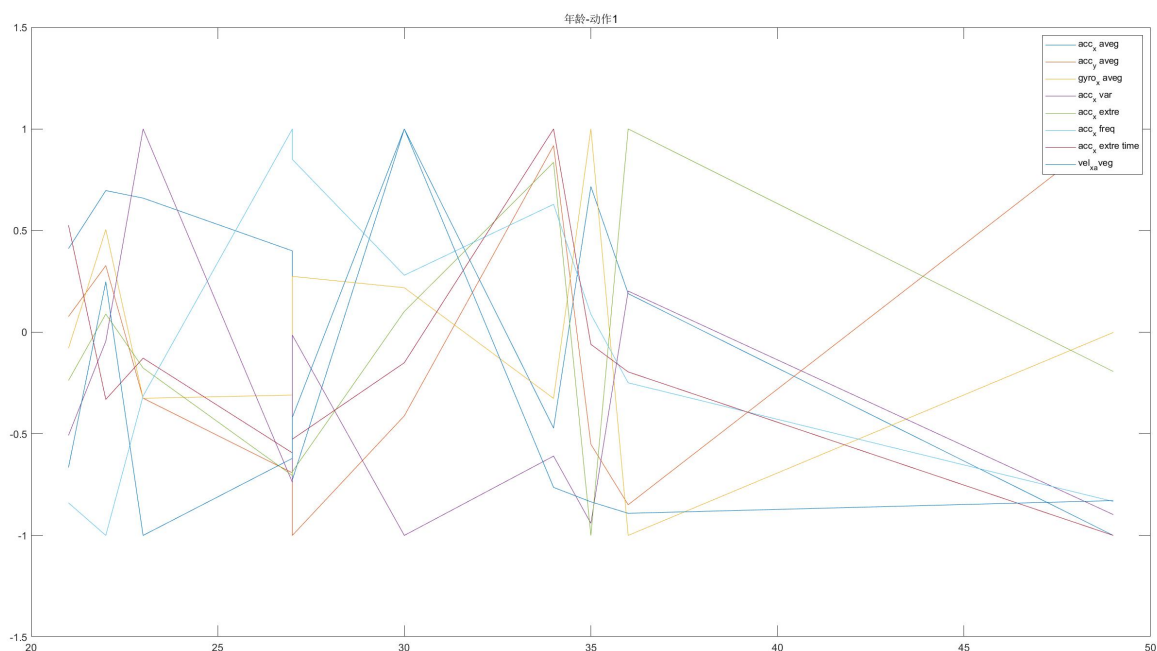


图 5 在同一动作下年龄与特征值间的关系

在图 5 中可以观察到年龄与动作 1 向前走之间的关系，在年龄达到 35 岁之前八种特征值的折线和年龄变化没有表现出明显的关联性，但是在年龄超过 35 岁之后，八种特征值与其联系均呈现明显的强线性关系。

例如 X 方向加速度方差和 X 方向最大值与最小值之间的时间间隔在年龄超过 35 岁之后便开始呈现数值增大的趋势，而其它的速度、加速度等的均值在数值上的表现却呈现出减小的趋势。这些显现表明实验人员在 35 岁之后对于向前走这一活动状态身体机能逐步下降，导致反映活动状态的数据中加速度和角速度的基础数值下降。

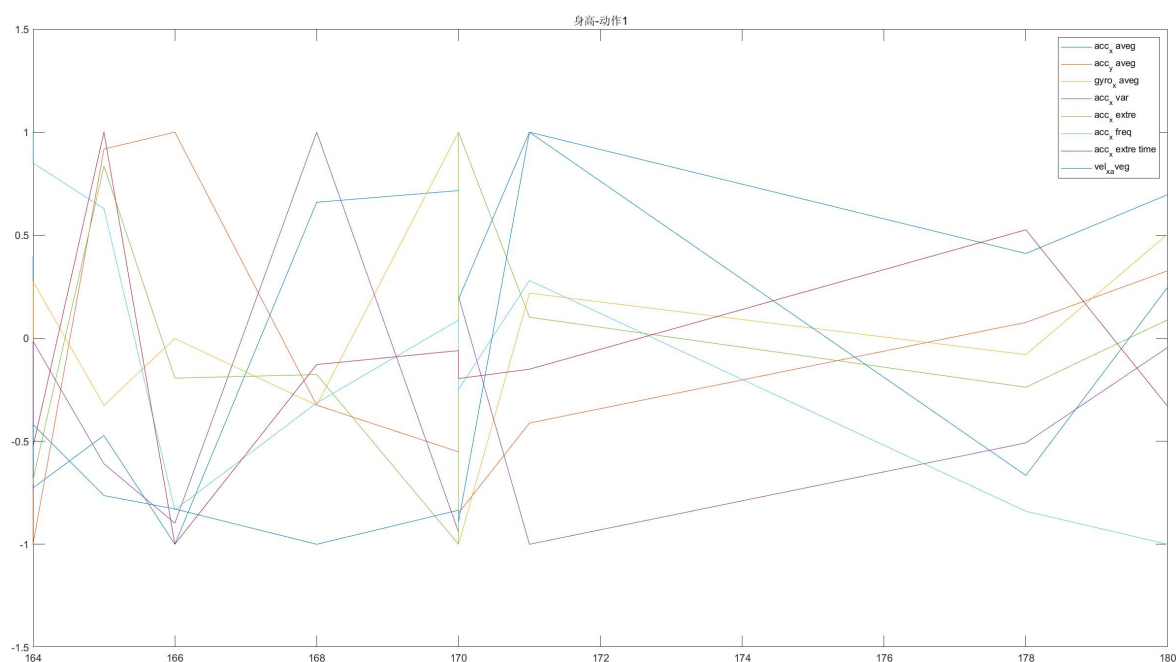


图 6 在同一动作下身高与特征值之间联系

在图 6 中可以观察到身高与动作 1 向前走之间的联系。在身高达到 170cm 之前，各特征值与身高之间的线性关系不明显，随着身高的提高各特征值可能增大或者减小。

但是当身高处于 170cm 至 178cm 之间时八项特征值与身高之间呈现强线性关系，具体表现为特征值 2 (Y 方向加速度平均值)、特征值 4 (X 方向加速度傅里叶变换后频率)、特征值 7 (x 方向加速度方差) 在数值上均表现出增大的趋势，而其他特征值表现出减小趋势。但是以上 8 中特征值在实验人员身高超过 178cm 之后数值增减趋势均呈现出于之前相反的现象。

这种现象表明实验人员身高不断增大时，当其处于 170 至 178cm 之间与各活动数据之间的关系和超过 178cm 之后活动数据的关系成相反的状态。

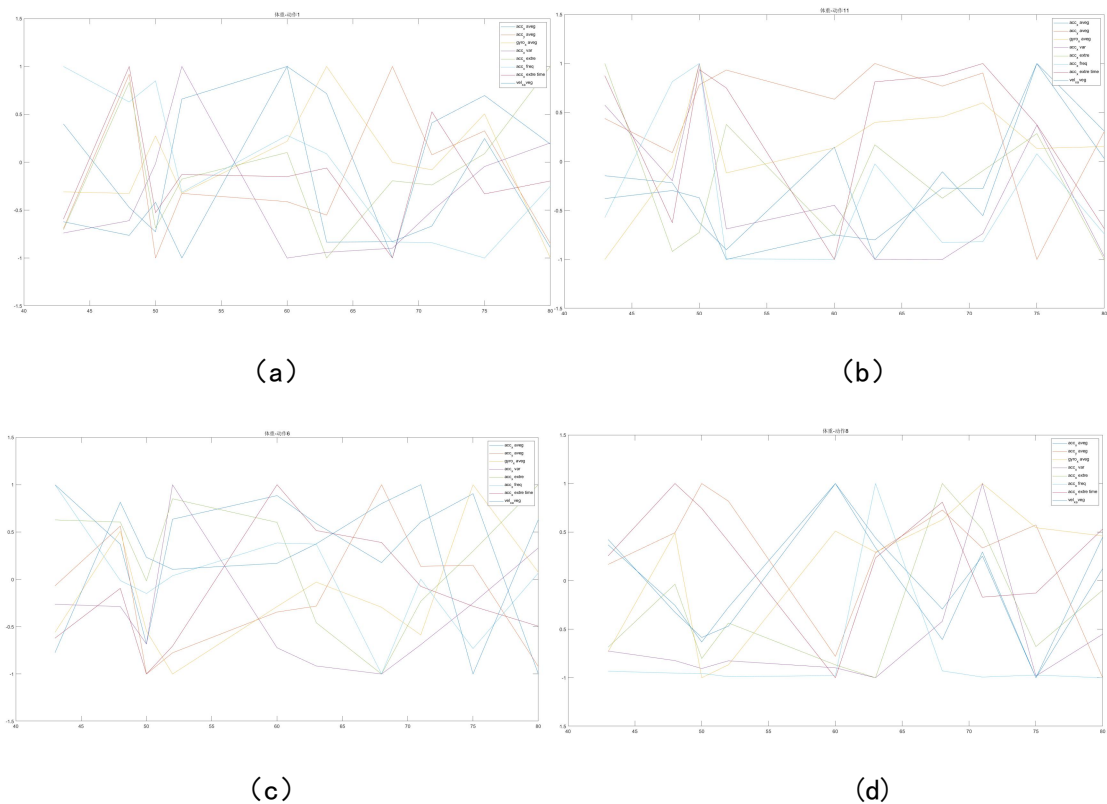


图 7 在同一动作下体重与特征值之间关系

如图 7 所示的四张图片展示了体重与 12 中活动状态间具有代表性的样例，可以观察在个活动状态下随着体重的不断变化计算得到的八种特征值的样条曲线也在不断变化，而这种变化间没有明显的线性关系，即在体重整体范围或在某一区间内没有哪一种特征值的样条曲线是有规律变化的。

由此我们推测，不同的体重范围都会有与之相关的活动数据，但是在这些数据不在数值上具有连续性，因为无法简单的将活动状态与活动数据进行关联。

对于附件 5 给出的五位隐藏身份的实验人员，本文同样用随机森林模型对其身份进行预测，结果显示 Unknow1 有 7 个特征预测为人员 7，5 个特征预测其为人员 9；

Unknow2 有 7 个特征预测其为人员 7，4 个特征预测其为人员 8，1 个特征预测其为人员 6；

Unknow3 有 5 个特征预测其为人员 7，4 个特征预测其为人员 8，3 个特征预测其为人员 6；

Unknow4 有 2 个特征预测其为人员 11，4 个特征预测其为人员 8，5 个特征预测其为人员 9，1 个特征预测其为人员 7；

Unknow5 有 5 个特征预测其为人员 10，4 个特征预测其为人员 11，1 个特征预测其为人员 9，2 个特征预测其为人员 10.

将以上数据进行汇总分析得到如下结果：

表 3：问题 3 结果

活动类型	判别结果
Unknow1	10
Unknow2	7
Unknow3	8
Unknow4	9
Unknow5	11

7 模型评价与改进

7.1 模型的优点

- (1) 问题 1 中提出的决策树模型其优点在于效率较高，具有一次建立可以重复使用的特点，在问题 2 中就利用这一特点对快速构建了随机森林模型。并且能够处理多类别的分类任务在本文中进行了十二种活动状态的分类，除此之外还能够处理混合类型的特征。
- (2) 问题 2 和 3 中提出的随机森林模型相较于决策树模型更具优势，其能够降低异常值带来的负面影响，随机森林模型通过构建多颗决策树，并将这些决策树得到的分类结果进行投票得到最终分类结果能够有效的避免因异常值带来的分类不准确的问题。
- (3) 决策树和随机森林模型发展已经较为成熟对于本文提出的分类问题具有很高的适应程度。

7.2 模型的不足

- (4) 问题 1 中提出的决策树模型容易出现过拟合现象，且当处理的数据关联性过强的时候往往处理的不够准确。
- (5) 当处理的数据样本数量不一致时，在决策树当中，信息增益的结果偏向于那些具有更多数值的特征。

7.3 模型的改进

经过阅读参考文献，提出了以下可能的改进措施：

1. 利用主成分分析和线性判别分析分别对原始特征空间作变换，得到的分类结果更准确。
2. 通过扩展特征空间使得属性子集分割更彻底，可以提高随机森林的泛化性能。

参考文献

- [1] 赵振武,李雪琴,贾朋霖.融合贝叶斯优化随机森林的机场旅客风险评估研究[J/OL].安全与环境学报,1-12[2024-07-10].
- [2] 王克忠,谢添,李梅,等.基于数值样本和随机森林分类器的岩爆风险快速预测代理模型[J].清华大学学报(自然科学版),2024,64(07):1203-1214.
- [3] 王郭艳,林建平,王阳.基于可解释随机森林的场地放大系数估算方法研究[J].河南城建学院学报,2024,33(03):29-36.张良均.Python 数据分析与挖掘实战[M].机械工业出版社,2016

附录

附录 A: 主要程序/关键代码

操作系统: window11

编程语言: MATLAB R2021a

代码清单 1

```
clc;clear;
P=ones(60,1);
SY=[1;10;11;12;13;14;15;16;17;18;19;
    2;20;21;22;23;24;25;26;27;28;29;
    3;30;31;32;33;34;35;36;37;38;39;
    4;40;41;42;43;44;45;46;47;48;49;
    5;50;51;52;53;54;55;56;57;58;59;
    6;60;7;8;9;];
feature_P=zeros(60,8);

path='D:\学习\数学建模\2024 湖南省省研究生赛\A 题\附件 1\Person3\';
namelist = dir([path,'*.xlsx']);
L = length(namelist);
appendix1_P = cell(1,L);
for i = 1:L
    filename{i} = [path,namelist(i).name];
    appendix1_P{1,i} = xlsread(filename{i});%读取 excel 文件
    namelist(i).name;
end
feature_P=[P,SY,feature_P];
for i=1:size(appendix1_P,2)
    a=cell2mat(appendix1_P(1,i));
    feature_P(i,3)=mean(a(:,1));

    feature_P(i,4)=mean(a(:,2));

    feature_P(i,5)=mean(a(:,4));

    feature_P(i,6)=var(a(:,4));

    feature_P(i,7)=max(a(:,1))-min(a(:,1));

    y=a(:,1)';
    if rem(size(y,2),2)~=0
        y(1)=[];
```

```

end

y=y-mean(y);

%对信号进行 fft 变换, 获取信号频率分布

L=length(y);           % 采样长度 应该是偶数
Fs=100;                 % 采样频率
f=Fs*(0:(L/2))/L;

% 傅里叶变换
S=y(1:L);
Y = fft(S);
P2 = abs(Y/L);
P1 = P2(1:L/2+1);
P1(2:end-1) = 2*P1(2:end-1);
[M,l]=max(P1);
feature_P(i,8)=f(l);

X=a(:,1);
while rem(length(X),100)~=0
    X(1)=[];
end
X=reshape(X,100,length(X)/100);
X=sum(X,1);
[m,i_max]=max(X);
[m,i_min]=min(X);
feature_P(i,9)=i_max-i_min;

vx=zeros(size(a,2),1);
for j=1:size(a,2)-1
    vx(j+1)=vx(j)+a(j,1)*0.01;
end
feature_P(i,10)=mean(vx);
end

feature_P3=sortrows(feature_P,2);
file='feature_P3.mat';
save(file,'feature_P3')

```

代码清单 2

```

clc;clear;
load('feature_P3.mat')
feature_P=feature_P3;

un_classify=feature_P;

un_classify=sortrows(un_classify,5);
classify=[un_classify(1:5,:),[3;3;3;3;3];un_classify(56:60,:),[2;2;2;2;2]];
un_classify(56:60,:)=[];un_classify(1:5,:)=[];

un_classify=sortrows(un_classify,3);
classify=[classify;un_classify(1:5,:),[10;10;10;10;10]];
un_classify(1:5,:)=[];

un_classify=sortrows(un_classify,6);
jump_run=un_classify(36:45,:);
un_classify(36:45,:)=[];
jump_run=sortrows(jump_run,8);
classify=[classify;jump_run(1:5,:),[6;6;6;6;6];jump_run(6:10,:),[7;7;7;7;7]];

un_classify=sortrows(un_classify,4);
up_down_sit=un_classify(21:35,:);
un_classify(21:35,:)=[];
up_down_sit=sortrows(up_down_sit,9);
classify=[classify;up_down_sit,[11;11;11;11;11;8;8;8;8;8;12;12;12;12;12]];

un_classify=sortrows(un_classify,6);
classify=[classify;un_classify((1:5),,:),[9;9;9;9;9]];
un_classify((1:5),,:)=[];

un_classify=sortrows(un_classify,10);
classify=[classify;un_classify,[4;4;4;4;4;1;1;1;1;5;5;5;5;5]];
un_classify=[];
classify_P3=classify;
file='classify_P3.mat';
save(file,'classify_P3')

```

代码清单 3

```

clc;clear;
P=ones(60,1);
SY=[10;10;10;10;10;
    11;11;11;11;11;

```

```

12;12;12;12;12;
1;1;1;1;1;
2;2;2;2;2;
3;3;3;3;3;
4;4;4;4;4;
5;5;5;5;5;
6;6;6;6;6;
7;7;7;7;7;
8;8;8;8;8;
9;9;9;9;9;
feature_P=zeros(60,8);

```

代码清单 4

```

clc;clear
load ('feature_Person4.mat')
load ('feature_Person5.mat')
load ('feature_Person6.mat')
load ('feature_Person7.mat')
load ('feature_Person8.mat')
load ('feature_Person9.mat')
load ('feature_Person10.mat')
load ('feature_Person11.mat')
load ('feature_Person12.mat')
load ('feature_Person13.mat')
feature=[feature_Person4;
         feature_Person5;
         feature_Person6;
         feature_Person7;
         feature_Person8;
         feature_Person9;
         feature_Person10;
         feature_Person11;
         feature_Person12;
         feature_Person13];
x = feature(:,3:end);
y = feature(:,2);
% 将数据集按 30%分成训练集和测试集
cv = cvpartition(size(x,1),'HoldOut',0.3);
idx = cv.test;
Xtrain = x(~idx,:);
Ytrain = y(~idx,:);
Xtest = x(idx,:);
Ytest = y(idx,:);
% 使用 TreeBagger 函数训练随机森林分类器
ntrees = 100; % 树的数量
B = TreeBagger(ntrees,Xtrain,Ytrain,'Method','regression','OOBPrediction','On','OOBPredictorImportance','on');
Ypred =predict(B,Xtest);

```

```

% Ytest = table2array(Ytest);

% 评估模型性能
rmse = sqrt(mean((Ytest - Ypred).^2));
r_squared = corrcoef(Ytest, Ypred).^2;

% 显示评估结果
disp(['均方根误差 (RMSE) : ', num2str(rmse)]);
disp(['R²值: ', num2str(r_squared(1, 2))]);
importance = B.OOBPermutedPredictorDeltaError; % 重要性
disp(['特征重要性: ', num2str(importance)]);

% 绘制测试集预测图
figure
scatter(Ytest, Ypred)
hold on
plot(xlim, ylim, '--k')
xlabel('测试集真实值');
ylabel('测试集预测值');
xlim([min(Ytest) max(Ytest)])
ylim([min(Ypred) max(Ypred)])
title(['测试集预测值 vs. 测试集真实值, R² = ', num2str(r_squared(1, 2))]);
file='B.mat';
save(file,'B')

```

代码清单 5

```

clc;clear
load ('feature_Person4.mat')
load ('feature_Person5.mat')
load ('feature_Person6.mat')
load ('feature_Person7.mat')
load ('feature_Person8.mat')
load ('feature_Person9.mat')
load ('feature_Person10.mat')
load ('feature_Person11.mat')
load ('feature_Person12.mat')
load ('feature_Person13.mat')
feature={feature_Person4;
         feature_Person5;
         feature_Person6;
         feature_Person7;
         feature_Person8;
         feature_Person9;
         feature_Person10;
         feature_Person11;
         feature_Person12;
         feature_Person13};
all_classify=[];

```

```

for i=1:length(feature)

    feature_P=cell2mat(feature(i));

    un_classify=feature_P;
    classify=[];

    un_classify=sortrows(un_classify,5);
    classify=[un_classify(1:5,:), [3;3;3;3;3];un_classify(56:60,:), [2;2;2;2;2]];
    un_classify(56:60,:)=[];un_classify(1:5,:)=[];

    un_classify=sortrows(un_classify,3);
    classify=[classify;un_classify(1:5,:), [10;10;10;10;10]];
    un_classify(1:5,:)=[];

    un_classify=sortrows(un_classify,6);
    jump_run=un_classify(36:45,:);
    un_classify(36:45,:)=[];
    jump_run=sortrows(jump_run,8);

    classify=[classify;jump_run(1:5,:), [6;6;6;6;6];jump_run(6:10,:), [7;7;7;7;7]];

    un_classify=sortrows(un_classify,4);
    up_down_sit=un_classify(21:35,:);
    un_classify(21:35,:)=[];
    up_down_sit=sortrows(up_down_sit,9);
    classify=[classify;up_down_sit, [11;11;11;11;11;8;8;8;8;8;12;12;12;12;12]];

    un_classify=sortrows(un_classify,6);
    classify=[classify;un_classify((1:5),,:), [9;9;9;9;9]];
    un_classify((1:5),,:)=[];

    un_classify=sortrows(un_classify,10);
    classify=[classify;un_classify, [4;4;4;4;4;1;1;1;1;1;5;5;5;5;5]];
    un_classify=[];
    all_classify=[all_classify;classify];

end
percent=0;
for i=1:(size(all_classify,1)/5)
    tag=all_classify((i*5-4):(i*5),2);
    tbl=tabulate(tag);
    percent=percent+(tbl(end,3)/100)*(1/120);
end

```


代码清单 6

```
clc;clear
SY=[1;10;11;12;13;14;15;16;17;18;19;
    2;20;21;22;23;24;25;26;27;28;29;
    3;30;4;5;6;7;8;9];
feature_P=zeros(30,8);
path='D:\学习\数学建模\2024 湖南省省研究生赛\A 题\附件 3\';
namelist = dir([path,'*.xlsx']);
L = length(namelist);
appendix1_P = cell(1,L);
for i = 1:L
    filename{i} = [path,namelist(i).name];
    appendix1_P{1,i} = xlsread(filename{i});%读取 excel 文件
    namelist(i).name;
end
feature_P=[SY,feature_P];
for i=1:size(appendix1_P,2)
    a=cell2mat(appendix1_P(1,i));
    feature_P(i,2)=mean(a(:,1));

    feature_P(i,3)=mean(a(:,2));

    feature_P(i,4)=mean(a(:,4));

    feature_P(i,5)=var(a(:,4));

    feature_P(i,6)=max(a(:,1))-min(a(:,1));

    y=a(:,1)';
    if rem(size(y,2),2)~=0
        y(1)=[];
    end
    y=y-mean(y);
    %对信号进行 fft 变换, 获取信号频率分布
    L=length(y);           % 采样长度 应该是偶数
    Fs=100;                 % 采样频率
    f=Fs*(0:(L/2))/L;
    % 傅里叶变换
    S=y(1:L);
    Y = fft(S);
    P2 = abs(Y/L);
    P1 = P2(1:L/2+1);
    P1(2:end-1) = 2*P1(2:end-1);
    [M,I]=max(P1);
```

```

feature_P(i,7)=f(l);

X=a(:,1);
while rem(length(X),100)~=0
    X(1)=[];
end
X=reshape(X,100,length(X)/100);
X=sum(X,1);
[m,i_max]=max(X);
[m,i_min]=min(X);
feature_P(i,9)=i_max-i_min;

vx=zeros(size(a,2),1);
for j=1:size(a,2)-1
    vx(j+1)=vx(j)+a(j,1)*0.01;
end
feature_P(i,9)=mean(vx);
end
feature=sortrows(feature_P,1);
load('B.mat')
Xtest=feature(:,2:end);
Ypred =predict(B,Xtest);
Ypred=round(Ypred);
an=[feature(:,1),Ypred];

```

代码清单 7

```

clc;clear

load ('feature_Person4.mat')
load ('feature_Person5.mat')
load ('feature_Person6.mat')
load ('feature_Person7.mat')
load ('feature_Person8.mat')
load ('feature_Person9.mat')
load ('feature_Person10.mat')
load ('feature_Person11.mat')
load ('feature_Person12.mat')
load ('feature_Person13.mat')

```

```

feature=[feature_Person4;

        feature_Person5;

        feature_Person6;

        feature_Person7;

        feature_Person8;

        feature_Person9;

        feature_Person10;

        feature_Person11;

        feature_Person12;

        feature_Person13];

feature=[feature,zeros(600,3)];

for i=1:size(feature,1)

    switch feature(i,1)

        case 4

            feature(i,11:13)=[27,164,43];

        case 5

            feature(i,11:13)=[23,168,52];

        case 6

            feature(i,11:13)=[27,164,50];

        case 7

            feature(i,11:13)=[35,170,63];

        case 8

            feature(i,11:13)=[30,171,60];

        case 9

            feature(i,11:13)=[22,180,75];

        case 10

            feature(i,11:13)=[49,166,68];

        case 11

            feature(i,11:13)=[21,178,71];

        case 12

            feature(i,11:13)=[34,165,48];

        case 13

            feature(i,11:13)=[36,170,80];

    end

end

feature=sortrows(feature,2);

```

```

feature_aveg = zeros((size(feature,1)/5),13);

for k = 1:(size(feature,1)/5)

    j = feature(k*5-4:k*5,:);

    feature_aveg(k,:) = mean(j);

end

% title('walk left:age-acc_xtime average')

% saveas(gcf,'walkleftage-acc_xtimeaverage.png')

% feature_for_norm = feature_aveg(:,3:10);

% [feature_norm,feature_norm_1] = mapminmax(feature_for_norm);

% feature_norm = feature_norm';

% feature_aveg(:,3:10) = feature_norm;

% figure(2)

% feature2=feature_aveg(1:10,:);

% plot(feature2(:,11),feature2(:,9),'*');

figure_title_reason = ['ÄËÄä';'Éíß';'İâÖØ'];

%                                     figure_title_act =

['İöÇ°×ß';'İö×6×ß';'İöÖÖ×ß';'½ĐĐİÄ¥';'½ĐĐİÄÄ¥';'İöÇ°ÄÜ';'İöÖ¼';'×øİÄ';'Ö¼ÄÇ';'İÉİÄ';'³É×øμçİÝİöÉİÖÆİ';'³É

×øμçİÝİöİÄÖÆİ'];

for i = 1:12

    feature_for_norm = feature_aveg(i*10-9:i*10,3:10);

    [feature_norm,feature_norm_1] = mapminmax(feature_for_norm);

    feature_norm = feature_norm';

    feature_aveg(i*10-9:i*10,3:10) = feature_norm;

    for k = 11:13

        figure((k-10)*100+i)

        set(gcf,'outerposition',get(0,'screensize'));

        plot_feature = feature_aveg(i*10-9:i*10,:);

        plot_feature=sortrows(plot_feature,k);

        for j = 3:10

            plot(plot_feature(:,k),plot_feature(:,j))

            hold on

        end

        title([figure_title_reason(k-10,:),'-İ×÷',int2str(i)])

        set(gca,'YLim',[-1.5 1.5]);%XÖáμÄËÿ¼ÝİÖÊ¾·İİ§

        legend('acc_x_aveg','acc_y_aveg','gyro_x_aveg','acc_x_var','acc_x_extre','acc_x_freq','acc_x_extre

time','vel_x_aveg');

```

```

saveas(gcf,['C:\Users\86137\Desktop\pÄi¹²Òµ´óÑ$ËÿÑ$½²´Ä£\code\i¼/E¬\png\','figure_title_reason(k-10,:),'-Π×
÷',int2str(i),'png'])

saveas(gcf,['C:\Users\86137\Desktop\pÄi¹²Òµ´óÑ$ËÿÑ$½²´Ä£\code\i¼/E¬\fig\','figure_title_reason(k-10,:),'-Π×
÷',int2str(i),'fig'])

    end
end

```

代码清单 8

```

clc;clear
load ('feature_Person4.mat')
load ('feature_Person5.mat')
load ('feature_Person6.mat')
load ('feature_Person7.mat')
load ('feature_Person8.mat')
load ('feature_Person9.mat')
load ('feature_Person10.mat')
load ('feature_Person11.mat')
load ('feature_Person12.mat')
load ('feature_Person13.mat')
feature=[feature_Person4;
         feature_Person5;
         feature_Person6;
         feature_Person7;
         feature_Person8;
         feature_Person9;
         feature_Person10;
         feature_Person11;
         feature_Person12;
         feature_Person13];
x = feature(:,2:end);
y = feature(:,1);
% 将数据集按 30%分成训练集和测试集
cv = cvpartition(size(x,1),'HoldOut',0.3);
idx = cv.test;
Xtrain = x(~idx,:);
Ytrain = y(~idx,:);
Xtest = x(idx,:);
Ytest = y(idx,:);
% 使用 TreeBagger 函数训练随机森林分类器
ntrees = 100; % 树的数量
B = TreeBagger(ntrees,Xtrain,Ytrain,'Method','regression','OOBPrediction','On','OOBPredictorImportance','on');
Ypred = predict(B,Xtest);
% Ytest = table2array(Ytest);

```

```

% 评估模型性能
rmse = sqrt(mean((Ytest - Ypred).^2));
r_squared = corrcoef(Ytest, Ypred).^2;

% 显示评估结果
disp(['均方根误差 (RMSE) : ', num2str(rmse)]);
disp(['R²值: ', num2str(r_squared(1, 2))]);

importance = B.OOBPermutedPredictorDeltaError; % 重要性
disp(['特征重要性: ', num2str(importance)]);

% 绘制测试集预测图
figure
scatter(Ytest, Ypred)
hold on
plot(xlim, ylim, '--k')
xlabel('测试集真实值');
ylabel('测试集预测值');
xlim([min(Ytest) max(Ytest)])
ylim([min(Ypred) max(Ypred)])
title(['测试集预测值 vs. 测试集真实值, R^2 = ', num2str(r_squared(1, 2))]);

file='B1.mat';
save(file,'B')

```

代码清单 9

```

clc;clear
SY=[10;11;12;1;2;3;4;5;6;7;8;9];
feature_P=zeros(12,8);
path='D:\学习\数学建模\2024 湖南省省研究生赛\A 题\附件 5\unknown5\';
namelist = dir([path, '*.xlsx']);
L = length(namelist);
appendix1_P = cell(1,L);
for i = 1:L
    filename{i} = [path,namelist(i).name];
    appendix1_P{1,i} = xlsread(filename{i});%读取 excel 文件
    namelist(i).name;
end
feature_P=[SY,feature_P];
for i=1:size(appendix1_P,2)
    a=cell2mat(appendix1_P(1,i));
    feature_P(i,2)=mean(a(:,1));

    feature_P(i,3)=mean(a(:,2));

```

```

feature_P(i,4)=mean(a(:,4));

feature_P(i,5)=var(a(:,4));

feature_P(i,6)=max(a(:,1))-min(a(:,1));

y=a(:,1)';
if rem(size(y,2),2)~=0
    y(1)=[];
end
y=y-mean(y);
%对信号进行 fft 变换, 获取信号频率分布
L=length(y);           % 采样长度 应该是偶数
Fs=100;                 % 采样频率
f=Fs*(0:(L/2))/L;
% 傅里叶变换
S=y(1:L);
Y = fft(S);
P2 = abs(Y/L);
P1 = P2(1:L/2+1);
P1(2:end-1) = 2*P1(2:end-1);
[M,l]=max(P1);
feature_P(i,7)=f(l);

X=a(:,1);
while rem(length(X),100)~=0
    X(1)=[];
end
X=reshape(X,100,length(X)/100);
X=sum(X,1);
[m,i_max]=max(X);
[m,i_min]=min(X);
feature_P(i,8)=i_max-i_min;

vx=zeros(size(a,2),1);
for j=1:size(a,2)-1
    vx(j+1)=vx(j)+a(j,1)*0.01;
end
feature_P(i,9)=mean(vx);
end
feature=sortrows(feature_P,1);

```

```
load('B1.mat')
Xtest=feature;
Ypred =predict(B,Xtest);
Ypred=round(Ypred);
an=[feature(:,1),Ypred];
file='unknow5.mat';
save(file,'an')
```