

所属类别	2023 年“华数杯”全国大学生数学建模竞赛	参赛编号
本科组		CM2301435

## 母亲身心健康对婴儿成长的预测和优化模型

### 摘要

本文针对母亲的身心健康对婴儿成长的影响的问题，通过母亲的身心健康指标如何对婴儿的行为特征和睡眠质量产生影响进行分析，建立了决策树模型和线性优化模型将母亲的身心健康对婴儿成长的影响的指标进行量化并更为直观地可视化，并针对改善 238 号婴儿的行为特征和睡眠质量提出了对母亲心理健康的最佳治疗方案。这对于缓解母亲的产后抑郁和新生儿的茁壮成长有着重要的意义。

在建立模型之前，我们先查阅了大量的相关资料，理解了各个指标的含义，并通过 TOPSIS[1] 对各个指标进行统一的正向化和标准化处理，使得每个指标都有统一的评判标准。

**针对问题一：**由于大部分变量数据并不符合正态分布，并且有序数类型变量，故而问题一采用**斯皮尔曼相关分析**的方法来分析母亲身体和心理指标对婴儿行为特征和睡眠质量的影响。最终说明了母亲的心理健康指标对婴儿的行为特征和睡眠质量都有较为显著的影响；而身体指标对婴儿的行为特征和睡眠质量影响都较小。

**针对问题二：**因为最后涉及预测问题，并且母亲的各个身体和心理指标之间对于婴儿的行为特征和睡眠质量的影响会有较大差异，所以优先考虑**回归模型、决策树、XGboost** 等机器学习算法，选取准确度高的模型进行预测。最终成功建立了母亲的身体和心理指标对婴儿行为特征的影响的模型，并给出了最后 20 个婴儿行为特征分类的预测结果。

**针对问题三：**由于 CBTS、EPDS、HADS 的治疗费用相对于患病程度的变化率均与治疗费用呈正比，进而确定了在使婴儿行为特征达到要求的前提下，以最低成本治疗母亲心理健康疾病的目标函数，然后将母亲的心理指标进行**加权量化**，再求出安静型和中等型婴儿需要对应的母亲的心理指标的**阈值**，最后建立**整数线性规划模型**求出**最优解**。最终针对 238 号样本给出了具有普适性的治疗方案。

**针对问题四：**首先由 **k-means 聚类**判断婴儿的睡眠质量，然后尝试多个模型建立婴儿综合睡眠质量与母亲身体和心理健康指标的联系，最后选取**决策树模型**进行最后 20 个婴儿综合睡眠质量的预测。最终成功将前 390 个婴儿的综合睡眠质量分为了优、中、差四类，并成功对最后 20 个婴儿的综合睡眠质量进行了预测。

**针对问题五：**问题五是问题三在结合了问题四之后的一个延伸，依然可以判断为整数线性规划求最优解的问题。采用与问题三同样的思路求解后，得出了通过治疗母亲心理健康疾病改善婴儿睡眠质量的模型，与问题三的模型进行对比后，发现只有问题三的治疗方案可以既满足使婴儿行为特征为安静型，又使婴儿的睡眠质量为优，故最终采用问题三的治疗方案，不需要调整。

关键词：TOPSIS，斯皮尔曼相关分析，决策树模型，整数线性规划，K-means 聚类

## 一、问题重述

### 1.1 问题背景

随着社会的发展，母亲的身心健康对婴儿成长的影响愈发引起人们的关注，母亲是婴儿的物质保障和情感寄托，也即母亲的一言一行均会对婴儿的认知、情感、社会行为等方面产生深远影响<sup>[2]</sup>。

科学家们通过各类身体与心理的指标，试图对母亲身心健康对婴儿成长的影响建立相关模型，并希望通过最优的方案来减轻此类影响。<sup>[3][4][5][6]</sup>但该类问题的影响因素较多，参数之间也存在或多或少的关联，直接进行分析所造成的误差将巨大。因此，本文通过科学地设置相应的权重，对数据进行处理，更精准地建立模型，以期减少母亲的身心健康对婴儿成长所产生的负面影响，并给出最优解。

### 1.2 问题提出

问题一：探究母亲的身体指标和心理指标对婴儿的行为特征和睡眠质量是否存在某种规律。

问题二：根据婴儿行为问卷，将婴儿的行为特征分为安静型、中等型、矛盾型三类，建立婴儿的行为特征与母亲的身体指标与心理指标的关系模型，对缺失的 20 组婴儿的行为特征信息进行预测并判断其属于的类型。

问题三：在治疗效果最好，花费治疗费用最少的条件下，根据给出的表格及 CBTS、EPDS、HADS 治疗费用相对于患病程度的变化率与治疗得用成正比的条件，建立模型判断一个行为特征为矛盾型的婴儿（编号 238）最少花费多少治疗费用，才能使婴儿的行为特征从矛盾型转变为中等型。

问题四：通过研究婴儿的整晚睡眠时间、睡醒次数、入睡方式的关系，对婴儿的睡眠治疗进行优、良、中、差的划分，并建立婴儿综合睡眠治疗与母亲的身体指标、心理指标的关联模型，并预测最后 20 组婴儿的综合睡眠质量。

问题五：根据问题三所得最优解的方法，判断对 238 号婴儿的睡眠质量评级为优的质量方案是否需要调整，该如何调整。

## 二、问题分析

### 2.1 问题一的分析

根据题意，需要对母亲的身体指标与婴儿的行为特征、母亲的身体指标对婴儿的睡眠状况、母亲的心理指标对婴儿的行为特征、母亲的心理指标对婴儿的睡眠质量之间的规律进行探究。

本文先对数据进行预处理，在剔除无效数据的同时正向化、标准化数据，并对数据进行可视化处理，判断数据特征，并据此来选用适合采用的相关分析的方法，建立母亲的身体指标与婴儿的行为特征、母亲的身体指标对婴儿的睡眠状况、母亲的心理指标对婴儿的行为特征、母亲的心理指标对婴儿的睡眠质量的四个分析模型，并进一步给出分析结果。

### 2.2 问题二的分析

由于题目根据婴儿行为问卷，将婴儿的行为特征分为安静型、中等型、矛盾型三类，且最后需要对缺失的 20 组婴儿的行为特征信息进行预测。

针对第一个子问题，由于母亲的各个身体和心理指标之间对于婴儿的行为特征和睡眠质量的影响有较大差异，可以认为选择母亲的各个指标作为判断婴儿行为特征的信息熵差别较大，对比使用回归模型、决策树<sup>[7]</sup>、XGboost<sup>[8]</sup>等机器学习算法，本文根据不同的机器学习算法，对婴儿的行为特征独立进行分类，并与题目所给数据进行对比，判断使用哪一个模型的准确率最高，对比可得决策树这一机器学习算法准确率最高，因此本文选用其作为解决问题二的主要模型。

针对第二个子问题，本文将数据的前百分之八十用来建立决策树模型，将数据的后百分之二十用来验证，建立决策树模型，并根据该模型的预测结果，以表格的形式给出缺失的 20 组婴儿的行为特征信息。

## 2.3 问题三的分析

根据题意可以判断这是一道整数线性规划求最优解的问题，且题目给出了 CBTS、EPDS、HADS 治疗费用相对于患病程度的变化率与治疗得用成正比的条件下，可以很容易地求出三种治疗费用的线性方程，约束条件为治疗效果达到要求，目标函数为治疗费用最少。再求出安静型和中等型婴儿需要对应的母亲的心理指标的阈值，权重和阈值根据问题二所建立的模型可以给出，最后进行整数规划求最优解，得出编号 238 治疗费用最少，治疗效果最好的方案。

## 2.4 问题四的分析

根据问题所给条件，可以先行判断婴儿的综合睡眠质量，由于题目已经明确给出是分为四类，优先对样本进行 **k-means 聚类分析**，但由于婴儿的入睡方式无法量化，也无法区分其优劣，故最后对婴儿综合睡眠质量进行评价时可以选择将其忽略。

聚类完成后，接下来的解题思路参照问题二，尝试不同的机器学习算法，将婴儿的睡眠质量分为优、良、中、差，建立婴儿综合睡眠治疗与母亲的身体指标、心理指标的关联模型，找出最优的模型进行预测，预测结果由表格给出。

## 2.5 问题五的分析

问题五是问题三的一个延伸，可判断为整数线性规划求最优解的问题，但由于有两个指标，且其对应的权重不同，权重最大的变量都是 EPDS 系数。

因而可以选择按照问题三找约束条件与目标函数，求解睡眠治疗评级为优的阈值的思路，先不考虑婴儿的行为特征而仅考虑婴儿的睡眠质量，将所求出的最优解的两个方案进行对比，选择既满足使婴儿行为特征为安静型，又满足使婴儿综合睡眠质量为优的治疗方案作为最佳方案。

# 三、模型假设

- (1) 假设各个受访者与其婴儿的指标相互独立。
- (2) 假设除编码无意义和违背常理的数据外，其余数据均无误，不作为异常值处理。
- (3) 假设所有受访者均认真且诚实地填写了问卷。
- (4) 假设除附件给出的指标之外，其余未提到的因素对所探究的问题都没有影响。

## 四、符号说明

符号	含义
$\hat{x}_x$	处理后的变量
$\rho_s$	斯皮尔曼相关系数
$H(X_i)$	变量 $X_i$ 的信息熵
$W(X_i)$	变量 $X_i$ 的权重
$Th_x$	阈值

## 五、模型的建立和求解

### 5.1 数据预处理

为了使建立的模型更加准确，达到剔除无效数据的同时正向化、标准化数据。本文先对题所给数据的前 390 组进行缺失值检验，结果为无缺失值，无需补充。其次，本文检测了题目所给数据是否存在异常值，结果为编号 180 的婴儿整晚睡眠时间异常、编号为 231、306、43、95、134、196、301、308、355 的母亲婚姻状况异常，将以上异常数据一并剔除。

此外，本文用 TOPSIS（优劣解距离法）对数据标准化处理，统一指标为 0-1，0 为最劣，1 为最优。经过调查文献可得，母亲的年龄，以 25-29 岁为最优区间<sup>[9]</sup>；母亲的妊娠时间，38-40 周为最优区间。睡眠时间，以 10-12 小时为最优区间；睡醒次数，以尽可能少为最优区间。对数据进行 TOPSIS 标准化处理：

记 25 岁为下限  $a_1$ ，29 岁为上限  $b_1$ ，记  $Mage$  为样本中离最优区间距离的最大值，则有：

$$Mage = \max\{x_i - a_1, b_1 - x_i\} \quad (1)$$

也即处理后的年龄为：

$$\hat{x}_{age} = \begin{cases} 1 - \frac{a_1 - x_{agei}}{Mage}, & x_{agei} < a_1 \\ 1, & a_1 \leq x_{agei} \leq b_1 \\ 1 - \frac{x_{agei} - b_1}{Mage}, & x_{agei} > b_1 \end{cases} \quad (2)$$

对母亲的妊娠时间做同样的处理，得到  $\hat{x}_{pregnancy}$

剔除异常值后，母亲的婚姻状况只剩 1 和 2，已婚优于未婚，故：

$$x_{marriage-1} = \hat{x}_{marriage} \quad (3)$$

母亲的教育程度同样也是越高越好，故：

$$\hat{x}_{edu} = \frac{(x_{edu}-1)}{4} \quad (4)$$

母亲的分娩方式中自然分娩要优于剖宫产：

$$\hat{x}_{\text{deliver}} = \begin{cases} 1, x_{\text{deliver}} = 1 \\ 0, x_{\text{deliver}} = 2 \end{cases} \quad (5)$$

母亲的心理指标都是 0-30，但均为极小型指标，因此所有的心理指标均为

$$\hat{x}_{p_i} = \frac{(30 - x_{p_i})}{30} \quad (6)$$

将婴儿的行为特征编码：安静型编码为 1，中等型编码为 2，矛盾型编码为 3；将整晚睡眠时间格式处理：将整晚睡眠时间的(时:分:秒)格式直接转化为小时。

## 5.2 问题一的模型的建立与求解

### 5.2.1 数据初步可视化

为了直观的感受母亲的身体指标和心理指标的数据情况，本文将题目给出的数据进行了初步的可视化，得到以下图 1。

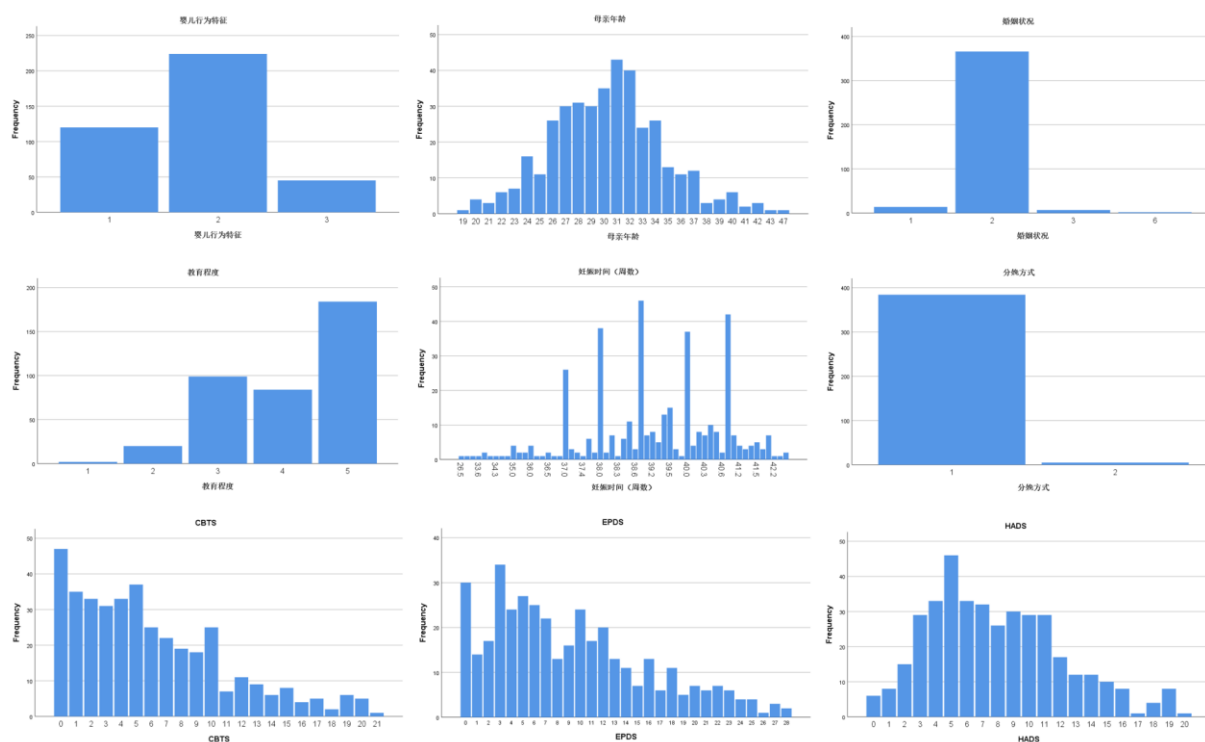


图 1 母亲身体指标与心理指标数

图 1 中，按左到右，从上到下的顺序排序，为婴儿行为特征、母亲年龄、婚姻状况、教育程度、妊娠时间（周数）、分娩方式、CBTS、EPDS、HADS。

婴儿行为特征中，安静型有 150 人，矛盾型有 45 人，中等型有 226 人，中等型所占比例是最大的；母亲年龄的数据近似服从正太分布，其中年龄为 31 岁的占比最大；婚姻状况已婚占比最大；教育程度为研究生的占比最大；妊娠时间的数据较为分散，但其绝大部分是在 37-41 周；分娩方式以自然分娩为主；CBTS（是分娩相关创伤后应激障碍（CB-PTSD）问卷，分数越高表示症状越明显。CB-PTSD 的症状包括反复回忆分娩过程、避免与分娩有关的刺激、情感麻木、易激惹、失眠等，会对女性的身心健康和社交生活造成严重影响。）的值基本服从值越大，人数越少的规律；EPDS（是爱丁堡产后抑郁量表（Edinburgh Postnatal Depression Scale）的缩写，是一种常用的心理量表。该量表旨在帮助评估产后抑郁症状的严重程度，包括心情低落、失眠、食欲改变等常见症

状。得分越高，表明症状越严重。) 的值基本服从值越大，人数越少的规律；HADS（是医院焦虑抑郁量表（Hospital Anxiety and Depression Scale）的缩写，是一种广泛使用的心理测量工具。该量表被用于评估患者在医院中是否存在焦虑和抑郁症状。得分越高，表明症状越严重。) 的值主要集中在 3-11。

5.2.2 斯皮尔曼相关分析<sup>[10]</sup>

在母亲的五个生理指标和三个心理指标中，仅有母亲年龄这一个指标服从正太分布，因此本题主要采用的方法是斯皮尔曼相关分析，而不采用皮尔逊相关检验。选用斯皮尔曼相关分析的原因还有该方法适合用于序数类型变量与数值类型变量之间的分析，且所给出的数据中有较多的异常值，而斯皮尔曼相关系数对异常值不敏感。

斯皮尔曼相关分析的计算公式如下：

$$\rho_s = 1 - \frac{6 \sum D_i^2}{n(n^2-1)}$$

(7)

其中， $D_i$  表示第*i*个数据对的位次之差， $n$  为总的观测样本数

5.2.3 母亲的生理指标与婴儿行为特征相关分析

通过程序对母亲的生理指标与婴儿行为特征进行斯皮尔曼分析。

表 1 母亲的生理指标与婴儿行为特征的斯皮尔曼分析

母亲生理指标与婴儿行为特征的斯皮尔曼分析						
	母亲年龄	婚姻状况	教育程度	妊娠时间	分娩方式	婴儿行为特征
母亲年龄	1					
婚姻状况	0.054	1				
教育程度	0.171**	0.096	1			
妊娠时间	-0.119*	-0.051	0.029	1		
分娩方式	0.037	0.006	-0.016	-0.176**	1	
婴儿行为特征	-0.096	-0.019	0.056	0.002	0.004	1

( \*相关性在 0.05 水平上显著(双尾)， \*\*相关性在 0.01 水平上显著(双尾)。 )

在斯皮尔曼相关分析的结果中，母亲的五个生理指标对婴儿的行为特征都没有显著性影响，但正如前文所言，母亲的年龄近似于正态分布，而婴儿的行为特征也可以近似为正态分布，因此对母亲年龄和婴儿行为特征单独进行皮尔逊相关分析。结果如下：

表 2 母亲生理指标与婴儿行为特征的皮尔逊相关分析

母亲生理指标与婴儿行为特征的皮尔逊相关分析		
	母亲年龄	婴儿行为特征
母亲年龄	1	
婴儿行为特征	-0.114*	1

( \*相关性在 0.05 水平上显著(双尾) )

从皮尔逊相关分析中可以认为母亲年龄对婴儿行为特征有较为显著的影响。即在母亲生理特征的五个指标中，只有母亲年龄对婴儿行为特征有显著影响。

5.2.3 母亲的心理指标与婴儿行为特征相关分析

通过程序对母亲的心理指标与婴儿行为特征进行斯皮尔曼分析。

表 3 母亲心理指标与婴儿行为特征的斯皮尔曼相关分析

母亲心理指标与婴儿行为特征的斯皮尔曼相关分析				
	CBTS	EPDS	HADS	婴儿行为特征
CBTS	1			
EPDS	0.781**	1		
HADS	0.711**	0.784**	1	
婴儿行为特征	0.126*	0.145**	0.129*	1

( \*相关性在 0.05 水平上显著(双尾), \*\*相关性在 0.01 水平上显著(双尾)。 )

分析结果可得, 母亲的心理指标 (GBTS、EPDS、HADS) 均对婴儿的行为特征有较为显著的影响, 其中 EPDS 的影响最为显著, 值达到 0.781。

5.2.4 母亲的物理指标与婴儿睡眠质量相关分析

通过程序对母亲的物理指标与婴儿睡眠质量进行斯皮尔曼分析。

表 4 母亲身体指标与婴儿睡眠质量的斯皮尔曼分析

母亲身体指标与婴儿睡眠质量的斯皮尔曼分析								
	母亲年龄	婚姻状况	教育程度	妊娠时间	分娩方式	整晚睡眠时间	睡醒次数	入睡方式
母亲年龄	1							
婚姻状况	0.054	1						
教育程度	0.171**	0.096	1					
妊娠时间	-0.119*	-0.051	0.029	1				
分娩方式	0.037	0.006	-0.016	-0.176	1			
整晚睡眠时间	0.014	-0.028	-0.056	0.074	0.016	1		
睡醒次数	0.037	-0.034	0.085	0.091	-0.062	-0.309**	1	
入睡方式	-0.08	-0.103*	-0.05	0.051	0.064	0.234**	-0.26**	1

( \*相关性在 0.05 水平上显著(双尾), \*\*相关性在 0.01 水平上显著(双尾)。 )

从结果可以看出, 在母亲的物理指标中只有婚姻状况对入睡方式有较为显著的影响, 其余指标均不显著。

5.2.5 母亲的心理指标与婴儿睡眠质量相关分析

通过程序对母亲的心理指标与婴儿睡眠质量进行斯皮尔曼分析。

表 5 母亲心理指标与婴儿睡眠质量的斯皮尔曼相关分析

母亲心理指标与婴儿睡眠质量的斯皮尔曼相关分析						
	CBTS	EPDS	HADS	整晚睡眠时间	睡醒次数	入睡方式
CBTS	1					
EPDS	0.781**	1				
HADS	0.711**	0.784**	1			
整晚睡眠时间	-0.129*	-0.167**	-0.117*	1		
睡醒次数	0.076	0.115*	0.065	-0.309	1	
入睡方式	0.046	0.006	0.06	0.234	-0.26	1

( \*相关性在 0.05 水平上显著(双尾), \*\*相关性在 0.01 水平上显著(双尾)。 )

其中, CBTS、EPDS、HADS 都对婴儿的整晚睡眠时间和睡醒次数有较为显著的影响, 其中 EPDS 的影响最为显著; 而只有 EPDS 对入睡方式有较为显著的影响; 则无法说明母亲的物理指标对婴儿的入睡方式有影响。

5.2.6 总结

从总体上来说，母亲的心理指标对婴儿的行为特征和睡眠质量（其中睡眠质量不包括入睡方式）都有显著影响；母亲的身体指标虽然个别指标对婴儿的行为特征和睡眠质量有一定影响，但若将其作为一个整体，则无法说明母亲的身体指标对婴儿的行为特征有显著影响。

从现实来看，母亲对婴儿的情绪传递方面：母亲的情绪状态可能会通过各种方式传递给婴儿。例如，心理状态良好的母亲可能积极地与婴儿互动，如唱歌、讲故事、玩游戏等，这有助于婴儿的社交技能和认知能力的发展，同时也有助于建立稳定的睡眠规律。反之，如果母亲的情绪状态稳定且积极，婴儿也会更加安静和快乐，这有助于提高婴儿的睡眠质量；母婴互动方面：母亲的心理状态会影响她与婴儿的互动方式。例如，心理状态良好的母亲更可能积极地与婴儿互动，如唱歌、讲故事、玩游戏等，这有助于婴儿的社交技能和认知能力的发展，同时也有助于建立稳定的睡眠规律。反之，心理状态不佳的母亲可能会减少与婴儿的互动，这可能会影响婴儿的社交和认知发展，同时也可能打乱婴儿的睡眠规律；母亲的抚养方式方面：母亲的心理状态也会影响她的抚养方式。例如，心理状态良好的母亲更可能采取积极的抚养方式，如定期喂养、定时换尿布、保持良好的睡眠习惯等，这有助于婴儿的健康发展和良好的睡眠质量。反之，心理状态不佳的母亲可能会忽视这些重要的抚养行为，这可能会对婴儿的健康和睡眠质量产生负面影响。

5.3 问题二的模型的建立与求解

5.3.1 子问题一的模型选择

本文对比分析了 XGBoost、随机森林、ET、朴素贝叶斯、K 最近邻、逻辑回归、决策树、支持向量机这八个机器学习算法，得到如下表格：

表 6 模型准确率比较表

模型比较分析	
模型	准确率
Decision Tree Classifier	0.64
XGBoost Classifier	0.59
Random Forest Classifier	0.63
Extra Trees Classifier	0.49
GaussianNB	0.49
K Neighbors Classifier	0.47
Logistic Regression	0.62
SVC	0.62

由上表可以对比得出，决策树的模型准确率最高。此外，决策树还能处理多类型的数据，可使用白盒模型，也能进行特征选择。因此，决策树模型更为适合解答该题。

5.3.2 子问题一的模型分析

根据问题分析及问题一数据情况，母亲的身体指标和心理指标对婴儿的行为特征影响的显著性差别较大，因此，本文将其认为选择母亲的各个指标作为判断婴儿行为特征的信息熵差别较大，并根据模型选择得出的数据，采用决策树模型，该方法的应用环境是在复杂的决策情况中，需要多层次多阶段的决策，其实就是由决策点、策略点（事件点）及结果构成的树形图，通过图解方式求解在不同条件下各类方案的效益值，比较后



做出决策。使用该模型可以较为准确地表征母亲心理与身体指标与婴儿行为特征之间的关系。

将百分之八十的数据用来建模，百分之二十的数据用来验证，建立决策树模型，如下：

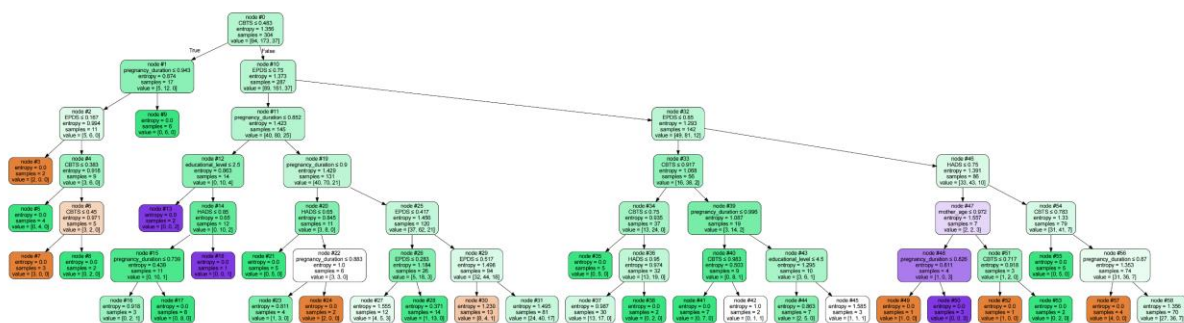


图 2 问题二决策树模型

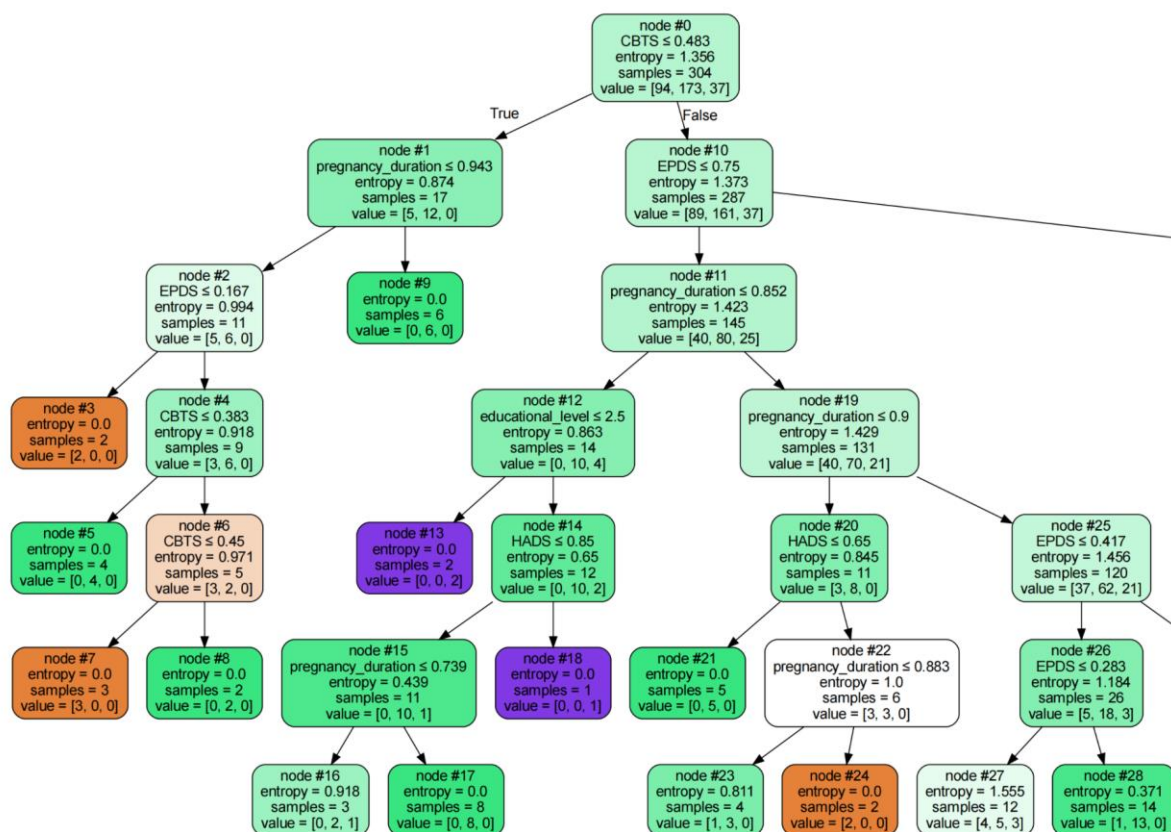


图 3 问题二决策树模型放大图 1

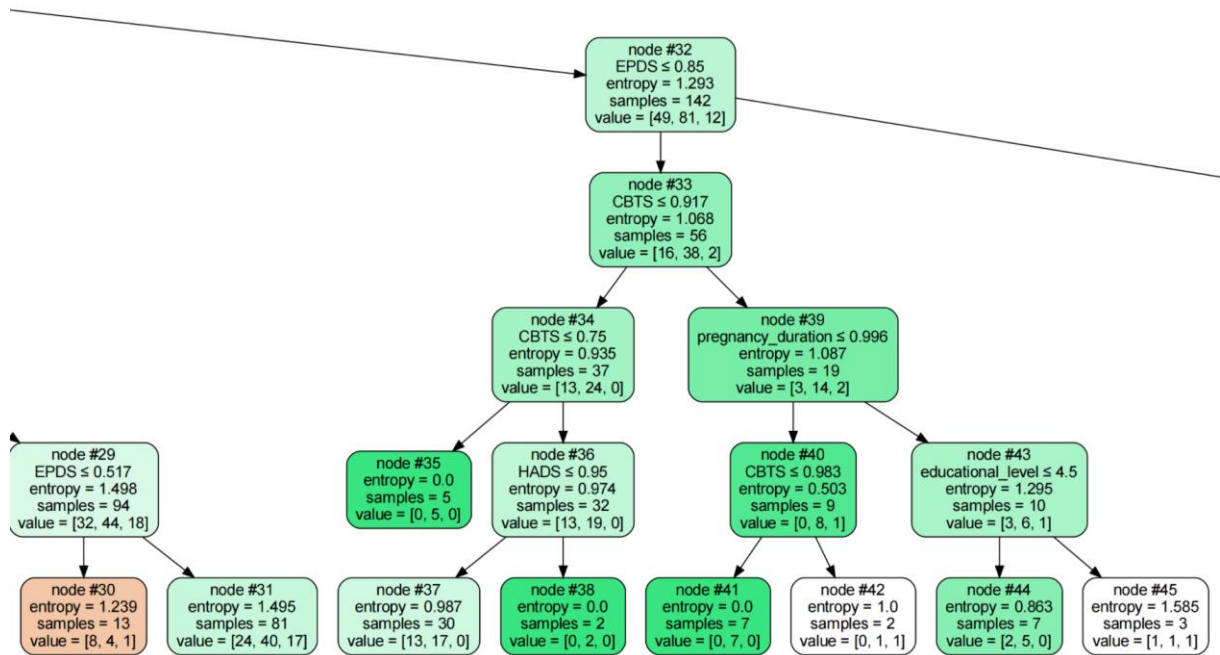


图 4 问题二决策树模型放大图 2

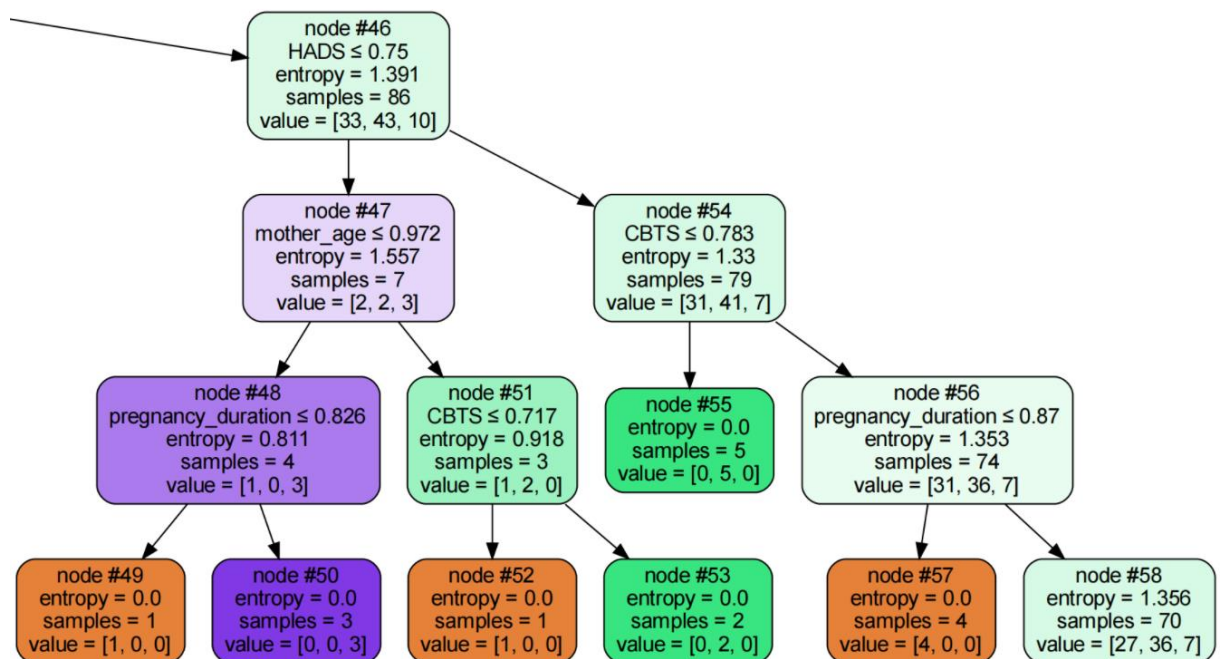


图 5 问题二决策树模型放大图 3

第一行是节点编号，第二行是分类依据，第三行是信息熵，第四行是样本数量，第五行是不同类别的个数。

详细数据见附录。

### 5.3.3 子问题二的求解结果

根据以上所构建的决策树模型，通过 Python 程序对后二十组婴儿行为特征进行预测，得到如下预测结果，具体代码在附录 2。

表 7 后二十组婴儿行为特征预测结果

编号	婴儿行为特征
391	安静型
392	中等型
393	中等型
394	中等型
395	中等型
396	安静型
397	中等型
398	中等型
399	中等型
400	中等型
401	中等型
402	中等型
403	中等型
404	中等型
405	中等型
406	中等型
407	中等型
408	中等型
409	中等型
410	中等型

5.4 问题三的模型的建立与求解

5.4.1 权重与阈值求解

通过治疗母亲的心理指标，让婴儿的行为特征评级为 2 或 1，以决策树中第 j 个分支所用到的第 i (i=1,2,3) 个 (  $X_1$ =CBTS,  $X_2$ =EPDS,  $X_3$ =HADS, 对应的得分为  $P(X_i)$  心理指标作为决策特征时的信息熵  $H(X_{i,j})$  作为母亲的心理指标对婴儿的行为特征的影响的权重系数，即：

$$W(X_i)=\frac{\sum_j \frac{H(X_{ij})}{2^j}}{\sum_i \sum_j \frac{H(X_{ij})}{2^j}} \tag{8}$$

计算结果如下：

$$W(x_1)=37.6\% \text{ 、 } W(x_2)=38.1\% \text{ 、 } W(x_3)=24.3\%$$

母亲的心理指标综合得分  $P_p$  的计算公式为：

$$\sum_i P(X_i)W(X_i) \quad (9)$$

因此，238 号的母亲心理指标综合评分应为  $P_p = 18.40$ 。

在不考虑母亲身体指标的情况下，仅靠改善母亲的心理健康水平让婴儿的行为特征改善为中等型或安静型，需要更加严格地规定母亲心理综合评分的阈值，故我们从 0 节点开始依次寻找，当且仅当信息熵<sub>[11]</sub>为 0 时才认为对母亲心理指标的改善可以使婴儿从矛盾性转变为中等型或安静型，即以节点内只有中等型的组所对应的心理指标综合评分作为中等型的阈值，记为  $Th_m = P_p(\text{note}\#n)$ ；只有安静型的组所对应的心理指标的综合评分作为安静型的阈值，记为  $Th_q = P_p(\text{note}\#n)$ （如有多个节点符合要求，取组内人数最多的组所对应的心理指标综合评分作为阈值；若没有节点完全符合要求，则选择该类型所占比重最大的点）。

由以上公式得安静型对应的指标阈值为： $Th_q = P_p(\text{note}\#6) = 5.98$ ；等型对应的指标阈值为： $Th_m = P_p(\text{note}\#35) = 9.78$

#### 5.4.2 整数线性规划

因为患病程度与治疗费率成正比，设 3 个指标需要降低的分数为  $\Delta d_1$ 、 $\Delta d_2$ 、 $\Delta d_3$ ，治疗所需费用  $f_i = k_i \Delta d_i + b_i$ ，解得  $k_1 = 870.67$ ， $k_2 = 695$ ， $k_3 = 2440$ ， $b_1 = 200$ ， $b_2 = 500$ ， $b_3 = 300$ 。

$$\begin{cases} f_1 = 870.67\Delta d_1 + 200 \\ f_2 = 695\Delta d_2 + 500 \\ f_3 = 2440\Delta d_3 + 300 \end{cases} \quad (11)$$

先对母亲心理指标建立一个综合评分标准，求出 3 个成分所占的比重，通过整数线性规划求最优解，目标方程为：

$$\text{Min}(870.67\Delta d_1 + 200 + 659\Delta d_2 + 500 + 2440\Delta d_3 + 300) \quad (12)$$

约束条件为：

$$\text{s. t.} \begin{cases} \sum_i \Delta d_i W(X_i) \geq 18.40 - Th \\ \Delta d_1, \Delta d_2, \Delta d_3 \geq 0 \\ \Delta d_1 \leq 15, \Delta d_2 \leq 22, \Delta d_3 \leq 18 \end{cases} \quad (13)$$

计算结果为：若要使婴儿行为特征转变为中等型，则需要治疗改善的心理指标为：

①  $\Delta d_1 = 1$ ， $\Delta d_2 = 22$ ， $\Delta d_3 = 0$ ，所需费用：16860.67 元，记为治疗方案 1。

若要使婴儿的行为特征转变为安静型，则需要治疗改善的心理指标为：

②  $\Delta d_1 = 11$ ， $\Delta d_2 = 22$ ， $\Delta d_3 = 0$ ，所需费用：24775.37 元，记为治疗方案 2。

### 5.5 问题四的模型的建立与求解

#### 5.5.1 K-means 聚类

对已有的睡眠质量进行评定所采用的模型为 K-means 聚类 (k=4)，这是聚类的经典算法，接受参数 k，以此为簇质心，将样本的每一点与每一簇质心计算距离，依据此距离对样本进行分配，再将每次簇的质心更改为该簇内所有点的平均值，中心思想为以空间中 k 个点为中心进行聚类，对最靠近他们的对象归类，通过迭代的方法，主次更新各聚类中心的值，直到得到最好的聚类结果。

表 8 初始聚类中心

初始聚类中心				
聚类变量/聚类组	1	2	3	4
整晚睡眠时间	11	7	5	12
睡醒次数	10	2	7	0
入睡方式	1	5	1	1

表 9 聚类过程

聚类过程				
聚类次数/聚类组	1	2	3	4
1	3.692	2.487	2.754	2.362
2	0.944	0.209	1.165	0.246
3	0.793	0.199	0.801	0.126
4	0.356	0.177	0.547	0.043
5	0.105	0.135	0.333	0
6	0	0.238	0.549	0.078
7	0.128	0.116	0.236	0.062
8	0.26	0.097	0.48	0.176
9	0.07	0	0.387	0.212
10	0.236	0.058	0.28	0.095
11	0.134	0.037	0.306	0.211
12	0.043	0.06	0.11	0.119
13	0.069	0.017	0.082	0.044
14	0.158	0.027	0.103	0.014
15	0.1	0.056	0.035	0.029
16	0.108	0.057	0	0
17	0	0	0	0

表 10 聚类结果

聚类结果				
聚类变量/聚类组	1	2	3	4
整晚睡眠时间	9.35	9.64	9.65	11.27
睡醒次数	4	2	1	0
入睡方式*	2	4	1	4

\*入睡方式对婴儿睡眠质量指标几乎没有影响，在评定等级时忽略

由聚类结果，可以把睡眠时间小于等于 9.35、睡醒次数大于等于 4 的婴儿的综合睡眠质量定为差，编号为 1；把整晚睡眠时间在 9.35 到 9.64、睡醒次数在 2 到 4 之间的婴儿睡眠质量定为中，编号为 2；把整晚睡眠时间在 9.65 到 11.27、睡醒次数为 1 的婴

儿睡眠质量定为良，编号为 3；把整晚睡眠时间大于 11.27、睡醒次数为 0 的婴儿睡眠质量定为优，编号为 4。

### 5.5.2 子问题一模型构建

同问题二理，对其建立决策树模型，如下：

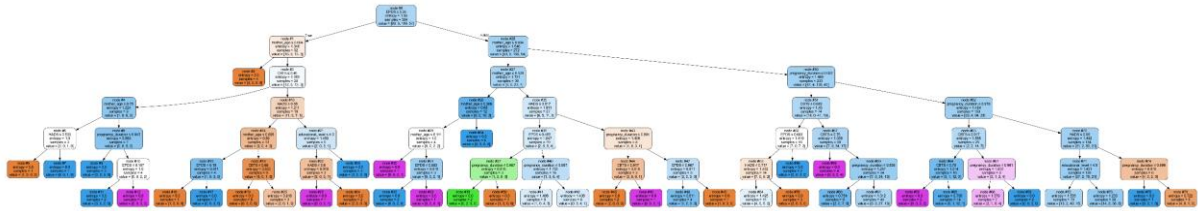


图 6 问题四决策树模型

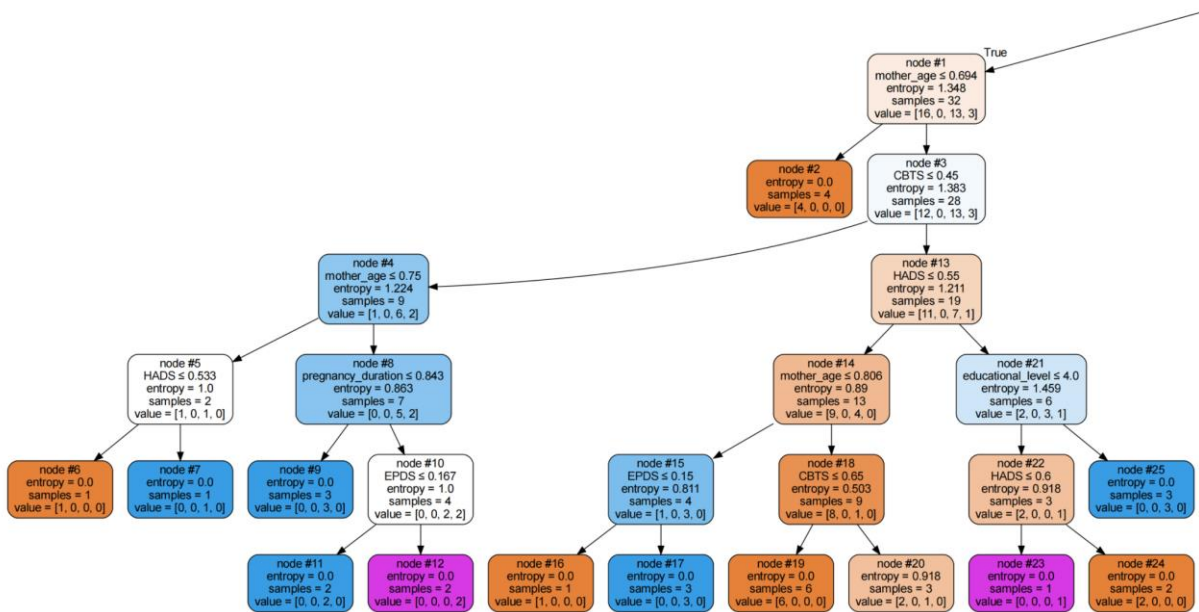


图 7 问题四决策树模型扩大图 1



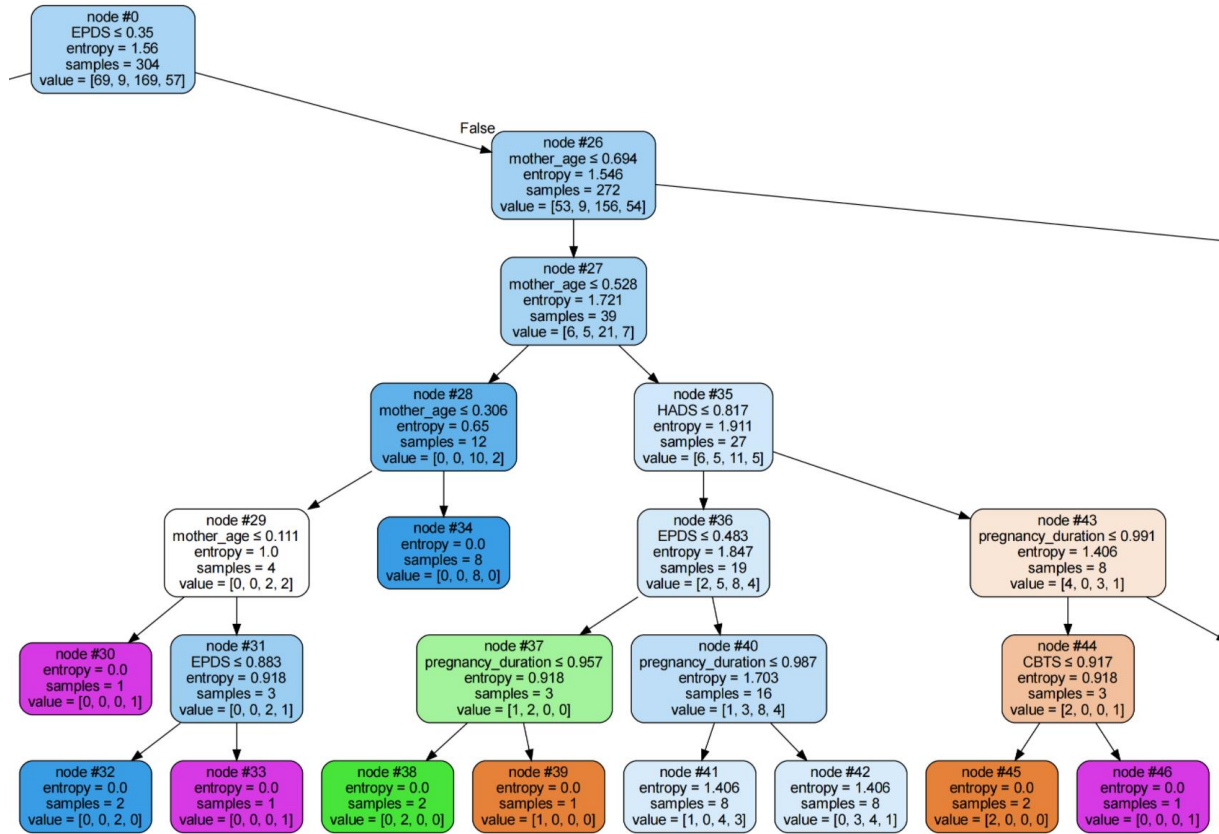


图 8 问题四决策树模型扩大图 2

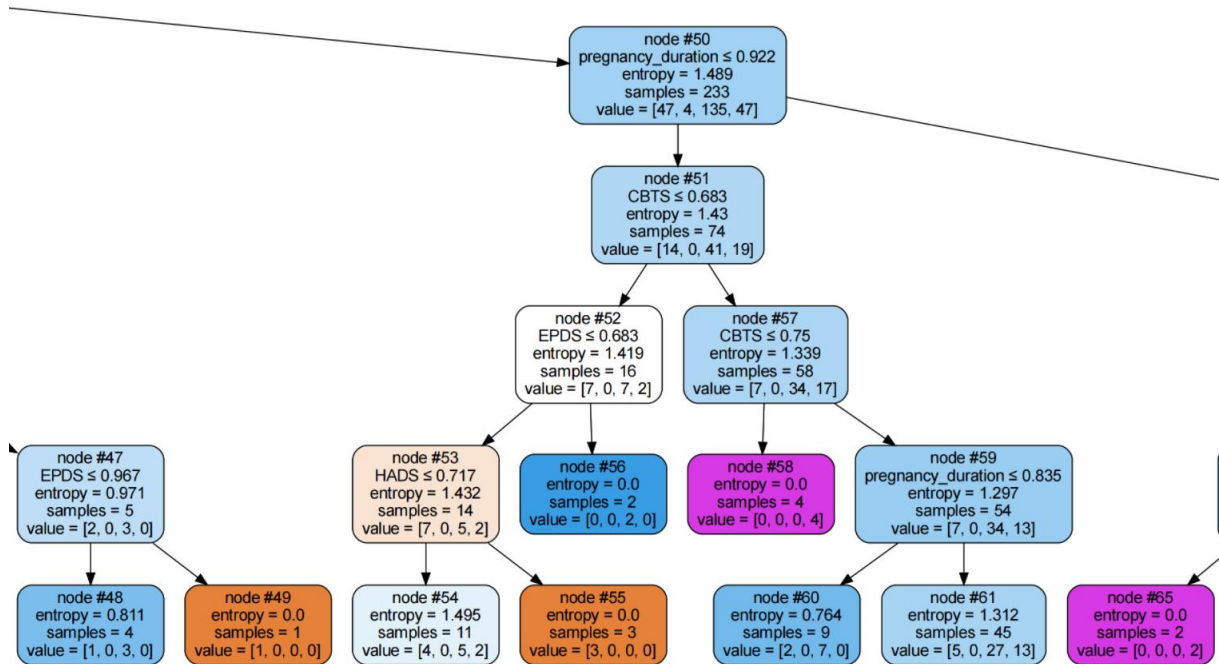


图 9 问题四决策树模型扩大图 3

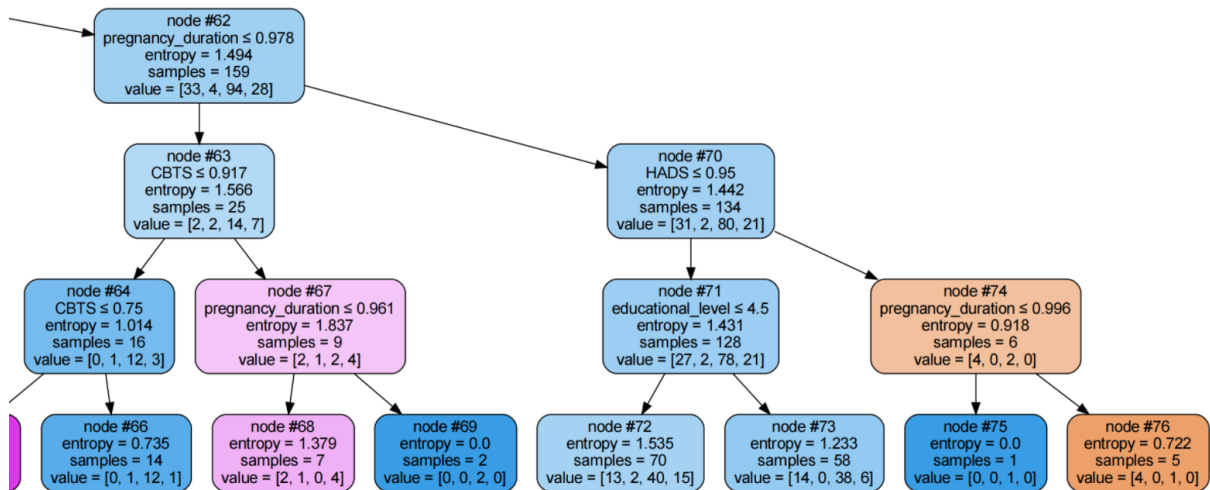


图 10 问题四决策树模型扩大图 4

第一行是节点编号，第二行是分类依据，第三行是信息熵，第四行是样本数量，第五行是不同类别的个数。详细数据见附录。

### 5.5.3 子问题二求解结果

根据以上所构建的决策树模型，对后二十组婴儿睡眠质量进行预测，得到如下预测结果：

表 11 后二十组婴儿综合睡眠质量预测结果

编号	综合睡眠质量
391	良
392	优
393	良
394	优
395	良
396	良
397	优
398	良
399	良
400	良
401	差
402	良
403	良
404	良
405	优
406	良
407	良
408	良
409	良
410	良



## 5.6 问题五的模型的建立与求解

在第三题的基础上，让 238 号的睡眠质量也为优，综合第三、第四题的模型，用与第三题同样的方法进行量化，设综合睡眠质量为差，中，良，优的的阈值为  $Th_1$ ， $Th_2$ ， $Th_3$ ， $Th_4$ 。 $Th_4 = P_p(\text{note}\#65) = 5.28$ 、 $W'(x_1) = 25.3\%$ 、 $W'(x_2) = 57.1\%$ 、 $W'(x_3) = 17.6\%$

由于母亲的 3 个心理指标对婴儿行为特征和综合睡眠质量的影响所占的比重不同，且  $Th_4 < Th_q$ ，所以治疗策略可能需要调整。

若仅仅改善母亲心理指标使婴儿的睡眠质量为优，目标方程与问题三相同，而约束条件变为：

$$\text{s. t.} \begin{cases} \sum_i \Delta d_i W'(X_i) \geq 18.40 - Th \\ \Delta d_1, \Delta d_2, \Delta d_3 \geq 0 \\ \Delta d_1 \leq 15, \Delta d_2 \leq 22, \Delta d_3 \leq 18 \end{cases} \quad (14)$$

若要使婴儿的综合睡眠质量转变为优，则需要治疗改善的心理指标为：

③  $\Delta d_1 = 3$ ， $\Delta d_2 = 22$ ， $\Delta d_3 = 0$ ，所需费用：18602.01 元，记为治疗方案 3。

但若采用治疗方案 3，则无法满足使婴儿的行为特征转变为安静型；但反过来，若采用治疗方案 2，则既可以满足使婴儿的睡眠质量转变为安静型，又可以满足使婴儿的综合睡眠质量转变为优，故而最终仍然采取方案 2，不需要调整。

## 六、模型的评价、改进与推广

### 6.1 模型的评价

#### 6.1.1 模型的优点

##### (1) TOPSIS

TOPSIS 是一种多属性决策分析方法，具有直观性高、计算简单、适用性广、考虑了所有属性、结果可靠、灵活性等优点。在本文中。TOPSIS 主要对数据标准化处理，统一指标为 0-1，0 为最劣，1 为最优。使得模型所建立的结果更为准确。

##### (2) 斯皮尔曼相关分析

斯皮尔曼相关分析是一种非参数统计方法，主要用于衡量两个变量之间的等级相关性，具有非参数方法、对异常值不敏感、可以处理定序数据、可以衡量非线性关系、易于理解和计算等优点，在本文中，题目所给的数据仅仅只有少部分存在正太分布，因此，选用斯皮尔曼相关分析也更加恰当。

(3) 决策树模型是常用的机器学习模型，其主要的优点为能处理多类型的数据，可使用白盒模型，也能进行特征选择。本文决策树模型主要用于建立母亲身心健康和婴儿行为特征及睡眠质量的联系。

(4) 整数线性规划<sup>[12]</sup>是一种优化技术，其模型简洁，可提供全局最优解，能处理非连续问题，本文主要用其来求最佳方案。

(5) K-means 聚类是一种常用的无监督学习方法，主要优点包括：计算效率高、可解释性强、适用于各种数据类型。本文主要用 K-means 聚类做分类，其实现效果好。

### 6.1.2 模型的缺点

对于决策树模型，由于数据集相对较少并且母亲的身体指标与婴儿的行为特征和综合睡眠质量的关系都比较低，所以在部分指标作为决策特征进行决策时，会对预测的准确率产生一定的影响。

在求解最优治疗方案时，默认 CBTS、EPDS、HADS 三个心理指标是相互独立的，其实不然，从问题 1 的相关性分析中我们可以看出，这三个指标之间有着较强的联系，在治疗时对一个指标的改善在很大程度上也会造成另一个指标的改善，但由于我们没有问卷各个题项的具体数据，所以很难对此做出量化，只能问题三与问题五认为是整数线性规划求最优解的问题。

### 6.2 模型的改进

可通过多种模型的融合，设置评价系数，构成一个庞大的综合评价模型，使得模型的结果更为准确。此外，还可获取原问卷的具体数据，量化去处理数据，构建更准确的模型。

### 6.3 模型的推广

本文建立的整数线性规划模型对于整数线性规划的问题具有一定的参考价值，可推广至工厂进材、货物运输、商品采购等相关领域的研究。

本文建立的决策树模型，实用性强，可推广至两个自变量影响一个因变量的情况，如：化学实验、天气模型等问题中。

## 参考文献

- [1]农静雅,戴钰莹,汤曼等.基于优劣解距离法、秩和比法和线性插值法的我国儿童保健工作质量综合评价[J].中国初级卫生保健,2023,37(01):56-60.
- [2]Ono, S (Ono, Sae); Ogi, H (Ogi, Hiroto); Ogawa, M (Ogawa, Masato); Nakamura, D(Nakamura, Daisuke), Nakamura, T (Nakamura, Teruhiko); Izawa, KP (Izawa, Kazuhiro P),Impact of Parents' Comprehensive Health Literacy on BMlin Children: A Multicenter Cross-Sectional Study in Japan, BMC PUBLIC HEALTH, DOI: 10.1186/s12889-021-10864-z, APR 24 2021.
- [3]赵茂矩,徐秀莲,赵萍,等. 母亲因素对婴儿早期教养影响的观察[J]. 中国妇幼保健,2005,20(22):2915-2918. DOI:10.3969/j.issn.1001-4411.2005.22.013.
- [4]陈桂香. 母亲因素对婴儿早期教养的影响分析[J]. 中国医药指南,2012,10(18):583-584. DOI:10.3969/j.issn.1671-8194.2012.18.450.
- [5]何满芬. 家庭养育指导对母亲育儿效能及产后抑郁情绪的影响[D]. 湖南:湖南师范大学,2016.
- [6]姜艳蕊,孙莞琦,王广海,等. 孕晚期母亲睡眠与婴儿出生早期睡眠的相关性研究[C]. //中国医师协会第七届睡眠医学学术年会论文集. 2016:419-419.
- [7]苗夺谦,王珏. 基于粗糙集的多变量决策树构造方法[J]. 软件学报, 1997, 8(6):7.DOI:CNKI:SUN:RJXB.0.1997-06-003.
- [8]Chen T, He T, Benesty M .xgboost: Extreme Gradient Boosting[J]. 2016.
- [9]李龙, 女性最佳生育年龄, [https://m.baidu.com/bh/m/detail/qz\\_12809595231795052401,2023.08.05](https://m.baidu.com/bh/m/detail/qz_12809595231795052401,2023.08.05)
- [10] Fieller, E.C.; Hartley, H.O.; Pearson, E.S. (1957) Tests for rank correlation coefficients. I. Biometrika 44, pp. 470–481
- [11]王国胤,于洪,杨大春.基于条件信息熵的决策表约简[J]. 计算机学报, 2002, 25(7):8.DOI:10.3321/j.issn:0254-4164.2002.07.013.
- [12]燕子宗.求解整数线性规划问题的一种新算法[J].长江大学学报 A(自然科学版), 2007.

## 附录 1 决策树表

表 12 问题二决策树模型表格数据

问题二决策树模型表格数据				
	classification_basis	entropy	samples	values
0	CBTS<=0.483	entropy=1.356	samples=304	value=94,173,37
1	pregnancy_duration<=0.943	entropy=0.874	samples=17	value=5,12,0
2	EPDS<=0.167	entropy=0.994	samples=11	value=5,6,0
3		entropy=0.0	samples=2	value=2,0,0
4	CBTS<=0.383	entropy=0.918	samples=9	value=3,6,0
5		entropy=0.0	samples=4	value=0,4,0
6	CBTS<=0.45	entropy=0.971	samples=5	value=3,2,0
7		entropy=0.0	samples=3	value=3,0,0
8		entropy=0.0	samples=2	value=0,2,0
9		entropy=0.0	samples=6	value=0,6,0
10	EPDS<=0.75	entropy=1.373	samples=287	value=89,161,37
11	pregnancy_duration<=0.852	entropy=1.423	samples=145	value=40,80,25
12	educational_level<=2.5	entropy=0.863	samples=14	value=0,10,4
13		entropy=0.0	samples=2	value=0,0,2
14	HADS<=0.85	entropy=0.65	samples=12	value=0,10,2
15	EPDS<=0.533	entropy=0.439	samples=11	value=0,10,1
16		entropy=0.918	samples=3	value=0,2,1
17		entropy=0.0	samples=8	value=0,8,0
18		entropy=0.0	samples=1	value=0,0,1
19	pregnancy_duration<=0.9	entropy=1.429	samples=131	value=40,70,21
20	HADS<=0.65	entropy=0.845	samples=11	value=3,8,0
21		entropy=0.0	samples=5	value=0,5,0
22	pregnancy_duration<=0.883	entropy=1.0	samples=6	value=3,3,0
23		entropy=0.811	samples=4	value=1,3,0
24		entropy=0.0	samples=2	value=2,0,0
25	EPDS<=0.417	entropy=1.456	samples=120	value=37,62,21
26	EPDS<=0.283	entropy=1.184	samples=26	value=5,18,3
27		entropy=1.555	samples=12	value=4,5,3
28		entropy=0.371	samples=14	value=1,13,0
29	EPDS<=0.517	entropy=1.498	samples=94	value=32,44,18
30		entropy=1.239	samples=13	value=8,4,1
31		entropy=1.495	samples=81	value=24,40,17
32	EPDS<=0.85	entropy=1.293	samples=142	value=49,81,12
33	CBTS<=0.917	entropy=1.068	samples=56	value=16,38,2
34	CBTS<=0.75	entropy=0.935	samples=37	value=13,24,0
35		entropy=0.0	samples=5	value=0,5,0
36	HADS<=0.95	entropy=0.974	samples=32	value=13,19,0
37		entropy=0.987	samples=30	value=13,17,0
38		entropy=0.0	samples=2	value=0,2,0

39	pregnancy_duration<=0.996	entropy=1.087	samples=19	value=3,14,2
40	CBTS<=0.983	entropy=0.503	samples=9	value=0,8,1
41		entropy=0.0	samples=7	value=0,7,0
42		entropy=1.0	samples=2	value=0,1,1
43	educational_level<=4.5	entropy=1.295	samples=10	value=3,6,1
44		entropy=0.863	samples=7	value=2,5,0
45		entropy=1.585	samples=3	value=1,1,1
46	HADS<=0.75	entropy=1.391	samples=86	value=33,43,10
47	mother_age<=0.972	entropy=1.557	samples=7	value=2,2,3
48	HADS<=0.65	entropy=0.811	samples=4	value=1,0,3
49		entropy=0.0	samples=1	value=1,0,0
50		entropy=0.0	samples=3	value=0,0,3
51	CBTS<=0.717	entropy=0.918	samples=3	value=1,2,0
52		entropy=0.0	samples=1	value=1,0,0
53		entropy=0.0	samples=2	value=0,2,0
54	CBTS<=0.783	entropy=1.33	samples=79	value=31,41,7
55		entropy=0.0	samples=5	value=0,5,0
56	pregnancy_duration<=0.87	entropy=1.353	samples=74	value=31,36,7
57		entropy=0.0	samples=4	value=4,0,0
58		entropy=1.356	samples=70	value=27,36,7

表 13 问题四决策树模型表格数据

问题四决策树模型表格数据				
	classification_basis	entropy	samples	values
0	CBTS<=0.483	entropy=1.356	samples=304	value=94,173,37
1	pregnancy_duration<=0.943	entropy=0.874	samples=17	value=5,12,0
2	EPDS<=0.167	entropy=0.994	samples=11	value=5,6,0
3		entropy=0.0	samples=2	value=2,0,0
4	CBTS<=0.383	entropy=0.918	samples=9	value=3,6,0
5		entropy=0.0	samples=4	value=0,4,0
6	CBTS<=0.45	entropy=0.971	samples=5	value=3,2,0
7		entropy=0.0	samples=3	value=3,0,0
8		entropy=0.0	samples=2	value=0,2,0
9		entropy=0.0	samples=6	value=0,6,0
10	EPDS<=0.75	entropy=1.373	samples=287	value=89,161,37
11	pregnancy_duration<=0.852	entropy=1.423	samples=145	value=40,80,25
12	educational_level<=2.5	entropy=0.863	samples=14	value=0,10,4
13		entropy=0.0	samples=2	value=0,0,2
14	HADS<=0.85	entropy=0.65	samples=12	value=0,10,2
15	EPDS<=0.533	entropy=0.439	samples=11	value=0,10,1
16		entropy=0.918	samples=3	value=0,2,1
17		entropy=0.0	samples=8	value=0,8,0
18		entropy=0.0	samples=1	value=0,0,1
19	pregnancy_duration<=0.9	entropy=1.429	samples=131	value=40,70,21

20	HADS<=0.65	entropy=0.845	samples=11	value=3,8,0
21		entropy=0.0	samples=5	value=0,5,0
22	pregnancy_duration<=0.883	entropy=1.0	samples=6	value=3,3,0
23		entropy=0.811	samples=4	value=1,3,0
24		entropy=0.0	samples=2	value=2,0,0
25	EPDS<=0.417	entropy=1.456	samples=120	value=37,62,21
26	EPDS<=0.283	entropy=1.184	samples=26	value=5,18,3
27		entropy=1.555	samples=12	value=4,5,3
28		entropy=0.371	samples=14	value=1,13,0
29	EPDS<=0.517	entropy=1.498	samples=94	value=32,44,18
30		entropy=1.239	samples=13	value=8,4,1
31		entropy=1.495	samples=81	value=24,40,17
32	EPDS<=0.85	entropy=1.293	samples=142	value=49,81,12
33	CBTS<=0.917	entropy=1.068	samples=56	value=16,38,2
34	CBTS<=0.75	entropy=0.935	samples=37	value=13,24,0
35		entropy=0.0	samples=5	value=0,5,0
36	HADS<=0.95	entropy=0.974	samples=32	value=13,19,0
37		entropy=0.987	samples=30	value=13,17,0
38		entropy=0.0	samples=2	value=0,2,0
39	pregnancy_duration<=0.996	entropy=1.087	samples=19	value=3,14,2
40	CBTS<=0.983	entropy=0.503	samples=9	value=0,8,1
41		entropy=0.0	samples=7	value=0,7,0
42		entropy=1.0	samples=2	value=0,1,1
43	educational_level<=4.5	entropy=1.295	samples=10	value=3,6,1
44		entropy=0.863	samples=7	value=2,5,0
45		entropy=1.585	samples=3	value=1,1,1
46	HADS<=0.75	entropy=1.391	samples=86	value=33,43,10
47	mother_age<=0.972	entropy=1.557	samples=7	value=2,2,3
48	HADS<=0.65	entropy=0.811	samples=4	value=1,0,3
49		entropy=0.0	samples=1	value=1,0,0
50		entropy=0.0	samples=3	value=0,0,3
51	CBTS<=0.717	entropy=0.918	samples=3	value=1,2,0
52		entropy=0.0	samples=1	value=1,0,0
53		entropy=0.0	samples=2	value=0,2,0
54	CBTS<=0.783	entropy=1.33	samples=79	value=31,41,7
55		entropy=0.0	samples=5	value=0,5,0
56	pregnancy_duration<=0.87	entropy=1.353	samples=74	value=31,36,7
57		entropy=0.0	samples=4	value=4,0,0
58		entropy=1.356	samples=70	value=27,36,7

---

## 附录 2 程序源码

相关系数分析:

```
import scipy.stats
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

plt.rcParams['font.sans-serif'] = ['SimHei']
plt.rcParams['axes.unicode_minus'] = False

data = pd.read_csv('./C_Python.csv')
# data = pd.DataFrame(data,columns=['CBTS','EPDS','HADS','infant_behavioral'])
data = data[:-20]

correlation_matrix = data.corr(method='spearman')
print(correlation_matrix)

plt.figure(figsize=(10,8))
sns.heatmap(correlation_matrix,annot=True,cmap='coolwarm',linewidths=0.5)
plt.title('Spearman Correlation Heatmap')

CBTS = np.array(data['CBTS'])
EPDS = np.array(data['EPDS'])
HADS = np.array(data['HADS'])
ib = np.array(data['infant_behavioral'])

s_CBTS = scipy.stats.spearmanr(CBTS,ib)
s_EPDS = scipy.stats.spearmanr(EPDS,ib)
s_HADS = scipy.stats.spearmanr(HADS,ib)

p_CBTS = scipy.stats.pearsonr(CBTS,ib)
p_EPDS = scipy.stats.pearsonr(EPDS,ib)
p_HADS = scipy.stats.pearsonr(HADS,ib)

print(s_CBTS)
print(s_EPDS)
print(s_HADS)
print('-----')
print(p_CBTS)
print(p_EPDS)
print(p_HADS)
```

第二题决策树模型:

```

# DecisionTree
import warnings
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import graphviz
from sklearn.metrics import confusion_matrix
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.metrics import accuracy_score, classification_report
from sklearn.tree import plot_tree
from sklearn.tree import export_graphviz

warnings.filterwarnings('ignore')

plt.figure(figsize=(10,6),dpi=300)

df = pd.read_csv('./C_Python.csv')

def fun(value,lb,ub,M):
    if value < lb:
        return 1.0-(lb-value)/M
    elif value > ub:
        return 1.0-(value-ub)/M
    else:
        return 1.0

def fun2(value,sum):
    value = value / sum
    if(value==0):
        return 0
    else:
        return np.log(value) * value

# TOPSIS
class TOPSIS():
    def __init__(self,data:pd.DataFrame):
        self.data = data
        # 母亲最佳生育年龄
        self.ma_lb = 25
        self.ma_ub = 29
        self.ma_M = max(self.ma_lb-
self.data['mother_age'].min(),self.data['mother_age'].max()-self.ma_ub)

```



```

        # 母亲最佳妊娠时间
        self.pd_lb = 38
        self.pd_ub = 40
        self.pd_M = max(self.pd_lb-
self.data['pregnancy_duration'].min(),self.data['pregnancy_duration'].max()-
self.pd_ub)

        # 婴儿最佳睡眠时间
        self.st_lb = 10
        self.st_ub = 12
        self.st_M = max(self.st_lb-
self.data['sleeping_time'].min(),self.data['sleeping_time'].max()-self.st_ub)

# 数据预处理
def process(self):
    # 删除 id
    self.data = self.data.drop(columns='id')

    # 对全部异常值的处理
    # 婚姻状态大于 2
    self.data =
self.data.drop(self.data[self.data['marriage_status']>2].index)
    # 填充后 20 行空缺值
    self.data = self.data.fillna(value=0)
    # 睡眠时间超过正常值
    self.data = self.data.drop(self.data[(self.data.sleeping_time<1) &
(self.data.sleeping_way>0)].index)

    # 区间型指标的正向化处理
    # 母亲生育年龄
    self.data['mother_age'] = self.data['mother_age'].apply(lambda
x:fun(x,self.ma_lb,self.ma_ub,self.ma_M))

    # 母亲妊娠时间
    self.data['pregnancy_duration'] =
self.data['pregnancy_duration'].apply(lambda
x:fun(x,self.pd_lb,self.pd_ub,self.pd_M))

    ## 独热编码
    ## 婚姻状态
    # dummies =
pd.get_dummies(self.data['marriage_status'],prefix='marriage_status')
    # self.data = self.data.drop(columns='marriage_status')
    # self.data = pd.concat([self.data,dummies],axis=1)

```

```

        # # 婴儿性别
        # dummies =
pd.get_dummies(self.data['infant_gender'],prefix='infant_gender')
        # self.data = self.data.drop(columns='infant_gender')
        # self.data = pd.concat([self.data,dummies],axis=1)

        # # 受教育程度
        # dummies =
pd.get_dummies(self.data['educational_level'],prefix='educational_level')
        # self.data = self.data.drop(columns='educational_level')
        # self.data = pd.concat([self.data,dummies],axis=1)

        # # 分娩方式
        # dummies =
pd.get_dummies(self.data['delivery_way'],prefix='delivery_way')
        # self.data = self.data.drop(columns='delivery_way')
        # self.data = pd.concat([self.data,dummies],axis=1)

        # # 婴儿年龄
        # dummies =
pd.get_dummies(self.data['infant_age'],prefix='infant_age')
        # self.data = self.data.drop(columns='infant_age')
        # self.data = pd.concat([self.data,dummies],axis=1)

        # # 睡眠方式
        # dummies =
pd.get_dummies(self.data['sleeping_way'],prefix='sleeping_way')
        # self.data = self.data.drop(columns='sleeping_way')
        # self.data = pd.concat([self.data,dummies],axis=1)

        # 心理指标归一化
self.data['CBTS'] = self.data['CBTS'].apply(lambda x:(30-x)/30)
self.data['EPDS'] = self.data['EPDS'].apply(lambda x:(30-x)/30)
self.data['HADS'] = self.data['HADS'].apply(lambda x:(30-x)/30)

        return self.data

def tocsv(self,coloum_name:str):
    res = np.array(self.data[coloum_name])
    res = res.reshape((-1,1))
    res = pd.DataFrame(res)
    res.to_csv('./C_'+coloum_name+'.csv',index=False)

```

```

TOPSIS_data = TOPSIS(df)
data = TOPSIS_data.process()
data_known = data[:-20]
data_predict = data[-20:]

feature_names = data_known[[
    'mother_age',
    'pregnancy_duration',
    'marriage_status',
    'educational_level',
    'CBTS',
    'EPDS',
    'HADS']]
label_name = data_known['infant_behavioral']
X_var = feature_names.values
Y_var = label_name.values

x_train,x_test,y_train,y_test =
train_test_split(X_var,Y_var,test_size=0.2,random_state=0)

# 使用决策树
dtc_model = DecisionTreeClassifier(criterion='entropy',max_depth=6)
dtc_model.fit(x_train,y_train)
y_test_pred = dtc_model.predict(x_test)
print('Accuracy of the dtc_model is {:.0%}'.format(accuracy_score(y_test,
y_test_pred)))
print(classification_report(y_test,y_test_pred))

# 尝试使用其他模型
# xgboost
from xgboost import XGBClassifier
xgb_model = XGBClassifier()
y_train_xgb = np.array(y_train-1)
y_test_xgb = np.array(y_test-1)
xgb_model.fit(x_train,y_train_xgb)
y_test_pred = xgb_model.predict(x_test)

print('Accuracy of the xgb_model is {:.0%}'.format(accuracy_score(y_test_xgb,
y_test_pred)))
print(classification_report(y_test_xgb,y_test_pred))

# # 尝试 xgboost 网格搜索
# # 定义参数的取值范围
# learning_rate = [0.01,0.015,0.025,0.05,0.1]

```

```

# subsample = [0.5,0.6,0.7,0.8,0.9]
# colsample_bytree = [0.5,0.6,0.7,0.8,0.9]
# max_depth = [3,5,6,7,9]

# parameters = {'learning_rate':learning_rate,
#               'subsample':subsample,
#               'colsample_bytree':colsample_bytree,
#               'max_depth':max_depth}
# xgb_model = XGBClassifier()

# # 进行网格搜索
# xgb_model_GS =
GridSearchCV(xgb_model,parameters,scoring='accuracy',verbose=1,n_jobs=-1)
# xgb_model_GS = xgb_model_GS.fit(x_train,y_train_xgb)
# print(xgb_model_GS.best_params_)
# print(xgb_model_GS.best_score_)
# print(xgb_model_GS.best_estimator_)

# xgb_model_new =
XGBClassifier(colsample_bytree=0.5,learning_rate=0.01,max_depth=3,subsample=0.
5)
# xgb_model_new.fit(x_train,y_train_xgb)

# # 使用数据预测类别
# y_train_pred = xgb_model_new.predict(x_train)
# y_test_pred = xgb_model_new.predict(x_test)

# print('Accuracy of the optimized xgb_model in training set is
 {:.0%}'.format(accuracy_score(y_train_xgb, y_train_pred)))
# print('Accuracy of the optimized xgb_model in testing set is
 {:.0%}'.format(accuracy_score(y_train_xgb, y_train_pred)))

# 随机森林
from sklearn.ensemble import RandomForestClassifier
rfc_model = RandomForestClassifier()
rfc_model.fit(x_train,y_train)
y_test_pred = rfc_model.predict(x_test)
print('Accuracy of the rfc_model is {:.0%}'.format(accuracy_score(y_test,
y_test_pred)))
print(classification_report(y_test,y_test_pred))

# ET
from sklearn.ensemble import ExtraTreesClassifier
et_model = ExtraTreesClassifier()

```

```
et_model.fit(x_train,y_train)
y_test_pred = et_model.predict(x_test)
print('Accuracy of the et_model is {:.0%}'.format(accuracy_score(y_test,
y_test_pred)))
print(classification_report(y_test,y_test_pred))

# 朴素贝叶斯
from sklearn.naive_bayes import GaussianNB
gnb_model = GaussianNB()
gnb_model.fit(x_train,y_train)
y_test_pred = gnb_model.predict(x_test)
print('Accuracy of the gnb_model is {:.0%}'.format(accuracy_score(y_test,
y_test_pred)))
print(classification_report(y_test,y_test_pred))

# K 近邻
from sklearn.neighbors import KNeighborsClassifier
knn_model = KNeighborsClassifier()
knn_model.fit(x_train,y_train)
y_test_pred = knn_model.predict(x_test)
print('Accuracy of the knn_model is {:.0%}'.format(accuracy_score(y_test,
y_test_pred)))
print(classification_report(y_test,y_test_pred))

# 逻辑回归
from sklearn.linear_model import LogisticRegression
lr_model = LogisticRegression(max_iter=1000)
lr_model.fit(x_train,y_train)
y_test_pred = lr_model.predict(x_test)
print('Accuracy of the lr_model is {:.0%}'.format(accuracy_score(y_test,
y_test_pred)))
print(classification_report(y_test,y_test_pred))

# 支持向量机
from sklearn.svm import SVC
svc_model = SVC()
svc_model.fit(x_train,y_train)
y_test_pred = svc_model.predict(x_test)
print('Accuracy of the svc_model is {:.0%}'.format(accuracy_score(y_test,
y_test_pred)))
print(classification_report(y_test,y_test_pred))

# 决策树节点信息输出成 txt 文本
output_list = plot_tree(dtc_model,feature_names=feature_names.columns)
```

```

file = open('C_nodes_2.txt','wt',-1)
file.write(str(output_list))
file.close()

# 决策树节点信息输出成 png 图片
dot_data = export_graphviz(dtc_model,
                            feature_names=feature_names.columns,
                            out_file=None,
                            node_ids=True,
                            rounded=True,
                            filled=True,
                            special_characters=True)
dot=graphviz.Source(dot_data,filename='C_DecisionTree_2.gv',format='png')
dot.view()

# 模型预测
feature_names = data_predict[[
    'mother_age',
    'pregnancy_duration',
    'marriage_status',
    'educational_level',
    'CBTS',
    'EPDS',
    'HADS']]

X_var = feature_names.values
Y_var = data_predict['infant_behavioral'].values
predict = dtc_model.predict(X_var)
print('The prediction result using the decision tree model is: ')
print(predict)

# 预测结果保存为 csv 文件
predict = predict.reshape((-1,1))
df_predict = pd.DataFrame(predict)
df_predict.to_csv('./C_prediction_2.csv',index=False,header=None)

```

问题三：治疗方案 1 的最优解

```

f = [870.67;659;2440];
intcon = [1,2,3];
A = [-0.38,-0.38,-0.24];
b = [-8.62];
lb = [0;0;0];
ub = [18;22;15];
[xval, fval] = intlinprog(f, intcon,A, b, [],[], lb, ub);

```

问题三：治疗方案 2 的最优解

```

f = [870.67;659;2440];
intcon = [1,2,3];
A = [-0.38,-0.38,-0.24];
b = [-12.42];
lb = [0;0;0];
ub = [18;22;15];
[xval, fval] = intlinprog(f, intcon,A, b, [],[], lb, ub);

```

第四题决策树模型:

```

import warnings
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import graphviz
from sklearn.tree import DecisionTreeClassifier # 树算法
from sklearn.model_selection import train_test_split # 拆分数据
from sklearn.metrics import accuracy_score,classification_report # 模型准确度
from sklearn.tree import plot_tree # 树图
from sklearn.tree import export_graphviz

warnings.filterwarnings('ignore')

plt.figure(figsize=(10,6),dpi=300)

df = pd.read_csv('./C_Python.csv')

def fun(value,lb,ub,M):
    if value < lb:
        return 1.0-(lb-value)/M
    elif value > ub:
        return 1.0-(value-ub)/M
    else:
        return 1.0

def fun2(value,sum):
    value = value / sum
    if(value==0):
        return 0
    else:
        return np.log(value) * value

# TOPSIS
class TOPSIS():

```

```

def __init__(self,data:pd.DataFrame):
    self.data = data
    # 母亲最佳生育年龄
    self.ma_lb = 25
    self.ma_ub = 29
    self.ma_M = max(self.ma_lb-
self.data['mother_age'].min(),self.data['mother_age'].max()-self.ma_ub)

    # 母亲最佳妊娠时间
    self.pd_lb = 38
    self.pd_ub = 40
    self.pd_M = max(self.pd_lb-
self.data['pregnancy_duration'].min(),self.data['pregnancy_duration'].max()-
self.pd_ub)

    # 婴儿最佳睡眠时间
    self.st_lb = 10
    self.st_ub = 12
    self.st_M = max(self.st_lb-
self.data['sleeping_time'].min(),self.data['sleeping_time'].max()-self.st_ub)

# 数据预处理
def process(self):
    # 删除 id
    self.data = self.data.drop(columns='id')

    # 对全部异常值的处理
    # 婚姻状态大于 2
    self.data =
self.data.drop(self.data[self.data['marriage_status']>2].index)
    # 填充后 20 行空缺值
    self.data = self.data.fillna(value=0)
    # 睡眠时间超过正常值
    self.data = self.data.drop(self.data[(self.data.sleeping_time<1) &
(self.data.sleeping_way>0)].index)

    # 区间型指标的正向化处理
    # 母亲生育年龄
    self.data['mother_age'] = self.data['mother_age'].apply(lambda
x:fun(x,self.ma_lb,self.ma_ub,self.ma_M))

    # 母亲妊娠时间

```



```

        self.data['pregnancy_duration'] =
self.data['pregnancy_duration'].apply(lambda
x:fun(x,self.pd_lb,self.pd_ub,self.pd_M))

        ## 独热编码
        ## 婚姻状态
        # dummies =
pd.get_dummies(self.data['marriage_status'],prefix='marriage_status')
        # self.data = self.data.drop(columns='marriage_status')
        # self.data = pd.concat([self.data,dummies],axis=1)

        ## 婴儿性别
        # dummies =
pd.get_dummies(self.data['infant_gender'],prefix='infant_gender')
        # self.data = self.data.drop(columns='infant_gender')
        # self.data = pd.concat([self.data,dummies],axis=1)

        ## 受教育程度
        # dummies =
pd.get_dummies(self.data['educational_level'],prefix='educational_level')
        # self.data = self.data.drop(columns='educational_level')
        # self.data = pd.concat([self.data,dummies],axis=1)

        ## 分娩方式
        # dummies =
pd.get_dummies(self.data['delivery_way'],prefix='delivery_way')
        # self.data = self.data.drop(columns='delivery_way')
        # self.data = pd.concat([self.data,dummies],axis=1)

        ## 婴儿年龄
        # dummies =
pd.get_dummies(self.data['infant_age'],prefix='infant_age')
        # self.data = self.data.drop(columns='infant_age')
        # self.data = pd.concat([self.data,dummies],axis=1)

        ## 睡眠方式
        # dummies =
pd.get_dummies(self.data['sleeping_way'],prefix='sleeping_way')
        # self.data = self.data.drop(columns='sleeping_way')
        # self.data = pd.concat([self.data,dummies],axis=1)

        # 心理指标归一化
self.data['CBTS'] = self.data['CBTS'].apply(lambda x:(30-x)/30)
self.data['EPDS'] = self.data['EPDS'].apply(lambda x:(30-x)/30)

```

```

        self.data['HADS'] = self.data['HADS'].apply(lambda x:(30-x)/30)

        self.data['sleeping_status'] = np.nan
        self.data.loc[(self.data.loc[:, 'sleeping_time'] <=
9.35), 'sleeping_status'] = 1
        self.data.loc[(self.data.loc[:, 'sleeping_time'] <= 9.64) &
(self.data.loc[:, 'sleeping_time'] > 9.35), 'sleeping_status'] = 2
        self.data.loc[(self.data.loc[:, 'sleeping_time'] <= 11.27) &
(self.data.loc[:, 'sleeping_time'] > 9.64), 'sleeping_status'] = 3
        self.data.loc[(self.data.loc[:, 'sleeping_time'] >
11.27), 'sleeping_status'] = 4
        return self.data

# 数据预处理
TOPSIS_data = TOPSIS(df)
data = TOPSIS_data.process()

# 数据分割
data_known = data[:-20]
data_predict = data[-20:]

feature_names = data_known[[
    'mother_age',
    'pregnancy_duration',
    'marriage_status',
    'educational_level',
    'CBTS',
    'EPDS',
    'HADS']]
X_var = feature_names.values
Y_var = data_known['sleeping_status'].values

x_train, x_test, y_train, y_test =
train_test_split(X_var, Y_var, test_size=0.2, random_state=0)

# 使用决策树
dtc_model = DecisionTreeClassifier(criterion='entropy', max_depth=6)
dtc_model.fit(x_train, y_train)
y_test_pred = dtc_model.predict(x_test)
print('Accuracy of the model is {:.0%}'.format(accuracy_score(y_test,
y_test_pred)))
print(classification_report(y_test, y_test_pred))

# 决策树输出成 txt 文件

```

```

output_list = plot_tree(dtc_model,feature_names=feature_names.columns)
print(type(output_list[0]))
file = open('C_nodes_4.txt','wt',-1)
file.write(str(output_list))
file.close()

# 决策树输出成 pdf 文件
dot_data = export_graphviz(dtc_model,
                            feature_names=feature_names.columns,
                            out_file=None,
                            node_ids=True,
                            rounded=True,
                            filled=True,
                            special_characters=True)
dot=graphviz.Source(dot_data,filename='C_DecisionTree_4.gv',format='png')
dot.view()

# 模型预测
feature_names = data_predict[[
    'mother_age',
    'pregnancy_duration',
    'marriage_status',
    'educational_level',
    'CBTS',
    'EPDS',
    'HADS']]

X_var = feature_names.values
Y_var = data_predict['sleeping_status'].values
predict = dtc_model.predict(X_var)

print('The prediction result using the decision tree model is: ')
print(predict)

# 预测结果保存为 csv 文件
predict = predict.reshape((-1,1))
df_predict = pd.DataFrame(predict)
df_predict.to_csv('./C_prediction_4.csv',index=False,header=None)

```

问题五：治疗方案 3 的最优解

```
f = [870.67;659;2440];
```

```
intcon = [1,2,3];
```