

赛区评阅编号（由赛区组委会填写）：

2022 年高教社杯全国大学生数学建模竞赛

承 诺 书

我们仔细阅读了《全国大学生数学建模竞赛章程》和《全国大学生数学建模竞赛参赛规则》（以下简称“竞赛章程和参赛规则”，可从 <http://www.mcm.edu.cn> 下载）。

我们完全清楚，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、“贴吧”、QQ 群、微信群等，与队外的任何人（包括指导教师）交流、讨论与赛题有关的问题；无论主动参与讨论还是被动接收讨论信息都是严重违反竞赛纪律的行为。

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。

我们以中国大学生名誉和诚信郑重承诺，严格遵守竞赛章程和参赛规则，以保证竞赛的公正、公平性。如有违反竞赛章程和参赛规则的行为，我们将受到严肃处理。

我们授权全国大学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）。

我们参赛选择的题号（从 A/B/C/D/E 中选择一项填写）：_____ A _____

我们的报名参赛队号（12 位数字全国统一编号）：_____ T10807878686 _____

参赛学校（完整的学校全称，不含院系名）：_____ 南京大学 _____

参赛队员（打印并签名）：1. _____

2. _____

3. _____

指导教师或指导教师组负责人（打印并签名）：_____

（指导教师签名意味着对参赛队的行为和论文的真实性负责）

日期：_____ 2022 _____ 年 _____ 7 _____ 月 _____ 30 _____ 日

（请勿改动此页内容和格式。此承诺书打印签名后作为纸质论文的封面，注意电子版论文中不得出现此页。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。）

数学建模竞赛论文评审问题

摘要

本文主要研究论文评审过程中公平性，合理性与可行性的问题。保证尽可能均匀的评委分配，降低评委打分偏好的影响都是保证评审公平的重要方面，解决上述相关问题是非常具有实际意义的。由于实际中问题规模往往不在可以人工处理的范围内，所以本文主要采用 matlab，python 编程，使用 excel，python 进行数据处理

问题一中，采用了如下基本方法：

- (1) 直接利用组合数定义得到， $m \leq C_n^3$
- (2) 该问题是实际上是施坦纳三元系问题 (steiner triple system)，直接利用相关结论即可
- (3) 将问题转化为 01 优化问题，使用 lingo 求解出最优解

问题二中，通过对评委打分情况进行分析，构造一种比较合理的分数调整方案，调整分数后可以得到更为公平的结果

问题三中，依托于问题二的基础之上，对各个步骤进行评估，对各个步骤进行修改以适应动态变化的权系数，再根据权系数修正论文的最终评分

问题四中，根据问题一二三中研究结果，引入圆桌模型以及裁判复核来使评审过程更为合理高效

关键词: 三元系 01 优化 加权分数修正 动态权系数 圆桌模型 裁判复核

一、问题重述

随着信息时代的到来，科学技术日新月异。在信息不断更新、不断变革的大潮中，考试和竞赛非但没有被取代，反而成为衡量教学水平和考察学生素质的一个重要标准。但这同时也带来了许多问题，尤其突出的是考试或竞赛中评阅是否公平、经济、合理的问题。因此，必须建立出一个较为客观的判断及改进评阅公正性的数学模型。从实际情况出发，根据不同的评审制度，解决如下问题：

(1) 若参加竞赛共有 m 组， n 个评委，每组论文 3 位评委评审，给出论文与评委的分配方法，找出任意不同的两个组的评委尽量不同时 m 、 n 值的满足条件。

(2) 在评委打分偏紧偏松的情况下，若 10 个评委评阅所有论文，请你给出数学模型与算法，估计每位评委打分的偏差，并对评委打分进行调整，计算获奖结果。

(3) 在评委只评阅了部分论文的情况下，给出数学模型与算法，就评审的每个步骤将分数进行调整，并计算出获奖结果。

(4) 对现行评审步骤的公正性，评委的工作量安排的合理性给出数学模型对现行的评审步骤进行评价并改进

二、问题分析

1) 以第一题的第 1, 2 小问得出的结果作为第 3 小问求解最优分配方案时的首要考虑因素，尽量减少任意两组有 2 个评委相同的情况，从而提高评委阅卷的广泛性，实现评审的公正性以及评委工作安排的合理性。

(2) 在评卷过程中，不可避免地会出现评委评分偏松或偏紧的情况。于是，给出一种评分一致性或公正性的检验方法，是非常必要的。运用统计学的原理，引入几个统计量，根据各评委的评分特点在这些统计量上表现出来的不同特征，对不同特点的评委进行分类，从而实现评分一致性或公正性的检验。

(3) 第三题中评分已经分成了 4 个阶段，这样，就不能原样的沿用问题二的模型，但是在假设评委态度不变的情况下，我们可以考虑，通过步骤 1 来确定各个评委之间的差异，以及调整方案。这样，就可以在第二题的模型基础上稍加改动，运用于第三题中。

三、模型假设

1. 每个评委都必须参与其中。
2. 每位评委在评审的整个过程中，评阅论文的风格始终保持一致。
3. 评委阅读论文速度相同。
4. 评委之间不会相互交流给分情况
5. 论文编号是随机分配的。

四、符号说明

符号	说明
n	评委数量
m	分组数量
$[x]$	表示 x 的整数部分
A	评分关系矩阵
a_{ij}	评委 i 是否给第 j 组评分
s_{ij}	评委 i 给第 j 篇论文评分的分值
\bar{s}_j	第 j 篇论文得分的均值
w_i	评委 i 的权系数
S_j	第 j 篇论文经过调整后的分数
R	重复度
R'	一个评委参与评分的分组数量的最大值
S_a^i	离差绝对值之和
m_i	均值差绝对值的均值
n_i	均值差绝对值的标准差
r_i	均值变异系数
C	完成度
Q	匹配度

五、模型的建立与求解

(注意：这个部分里面的标题可根据你的论文内容进行调整，我这里给的是一个通用的模版)

5.1 问题一

5.1.1 第一问建模与求解

任意两个组的评委不能完全相同，求解组数 m 和评委数 n 的关系。从评委角度出发。

(1) 根据组合数的定义，得出 n 个评委，每 3 个一组，总共能构成 C_n^3 个不完全相同的组，根据抽屉原理可知

$$m \leq C_n^3$$

(2) 每个评委都要参与评分，那么极端情况为所有组的评委都完全不一样，可知

$$n \leq 3m$$

综上所述， m ， n 的关系为

$$\frac{n}{3} \leq m \leq C_n^3$$

5.1.2 第二问的建模与求解

要保证任意两个组出现没有两位评委相同的情况，求 m, n 的关系 (1) 建模

给 n 个评委编号为 $1, 2, \dots, n$ ，并且用集合表示 $J = \{1, 2, \dots, n\}$ ，问题转化为求 J 的所有三元素子集并且这些子集中任意两个只有一个重复元素

(2) 模型求解

定义符号 $[x]$ 表示取 x 的整数部分，那么每一个元素都至多在 $[\frac{n-1}{2}]$ 个不同的子集中，那么如果计入重复，那么有 $n[\frac{n-1}{2}]$ 个，而再上述计算中，每个元素会重复 3 次，那么可以得到一个比较紧的 m 的上界，即

$$[\frac{n[\frac{n-1}{2}]}{3}]$$

下面说明这是一个比较紧的上界：

case1: 当 $n = 1, 3 \pmod 6$ 时，

原上界可以化简为 $\frac{n(n-1)}{6}$ ，T.Skolem 已经证明该上界是可以取到的 [1]

case2: 当 $n = 0, 2 \pmod 6$ 时，同 case1 可以得出该上界是可以取到的

case3: 当 $n = 4, 5 \pmod 6$ 时，当 $n=6k+5$ 时，记上界为 S ，即 $S = 6k^2 + 4k + 3$

根据定理 1.1.3[2]，含有 $6n+5$ 个元素的 PBD 可以为一个长为 5 的块和 L 个长为 3 的块，即有最大值 M 满足

$$M \geq [\frac{(6k+5)(3k+2) - S}{3}] = 6k^2 + 9k + \frac{10 - S}{3}$$

注意到 L 为整数，所以 $10-S$ 是 3 的倍数，即 $S=1$ 或 4

从而得出 $M \geq 6k^2 + 9k + 3$ 或者 $M \geq 6k^2 + 9k + 3$ ，即 $M \geq S - 1$

当 $n=6k+4$ 时，对比 $n=6k+5$ 的情况

1. $M=S$ 则去掉一个点，减少 $3k+2$ 种可能， $M = 6k^2 + 6k + 1$

2. $M=S-1$ ，则必有一个点的配对数为 $\frac{n}{2} - 1$ 个，那么将这个点删去将会减少 $3k+1$ 个可能，此时， $M=6k^2 + 6k + 1$

又，对于 $n=6k+4$ ，有上界 $S' = 6k^2 + 6k + 1$

综上所述， m 和 n 的关系为

$$m \leq [\frac{n[\frac{n-1}{2}]}{3}]$$

只有当 $n = 5 \pmod 6$ 时，该上界不是紧的，最大值最少只会比上界小 1

5.1.3 第三问的建模与求解

(1) 建模有 m 个组， n 个评委，将评委给一个组评分这个关系表示成大小为 $m \times n$ 的矩阵，其中矩阵第 i 行第 j 列的元素为 1 表示 j 号评委给第 i 个组评分，反之则为 0

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

由于每个组都有且仅有 3 个评委评审，所以有

$$\sum_{j=1}^n a_{ij} = 3$$

引入一个重复度 R 表示任意两个组的评委重复个数之和，对于任意 m, n ，可知 $R=3m-n$ ，给定 m, n 确定值时，重复度 R 是确定的

此外引入 R' 表示一个评委参与评分的分组数量的最大值

结合上述两个指标，原问题转化为最优化问题，如下：

$$\begin{aligned} \min \quad R' &= \max_j \sum_{i=1}^m a_{ij}, j \in \{1, 2, \dots, n\} \\ \text{s.t.} \quad \sum_{j=1}^n a_{ij} &= 3, \forall i \in \{1, 2, \dots, n\} \end{aligned}$$

- (2) 模型求解带入具体数据 $m=20, n=10$ ，使用 lingo 求解上述模型优化问题
一个最优的分配方式如下：

评委 分组	1	2	3	4	5	6	7	8	9	10
1	1	1	1	0	0	0	0	0	0	0
2	1	0	0	1	1	0	0	0	0	0
3	1	0	0	0	0	1	1	0	0	0
4	1	0	0	0	0	0	0	1	1	0
5	0	1	0	1	0	1	0	0	0	0
6	0	1	0	0	1	0	1	0	0	0
7	0	1	0	0	0	0	0	1	0	1
8	0	0	1	1	0	0	1	0	0	0
9	0	0	1	0	1	1	0	0	0	0
10	0	0	1	0	0	0	0	0	1	1
11	0	0	0	0	0	0	0	1	1	1
12	0	0	0	0	0	0	1	0	1	1
13	0	0	0	0	1	1	0	1	0	0
14	0	0	0	1	0	0	0	0	1	1
15	0	0	0	0	0	1	1	1	0	0
16	0	0	1	1	1	0	0	0	0	0
17	1	1	0	0	0	0	0	0	0	1
18	0	0	0	0	0	0	1	1	1	0
19	0	0	0	1	1	1	0	0	0	0
20	0	1	1	0	0	0	0	0	0	1

表 1: 分配方案

5.2 问题二模型的建立与求解

5.2.1 数据预处理

观察评委打分情况 根据上述图标，可以按评分风格大致分为如下几类：1) 客观公平型：这种评委

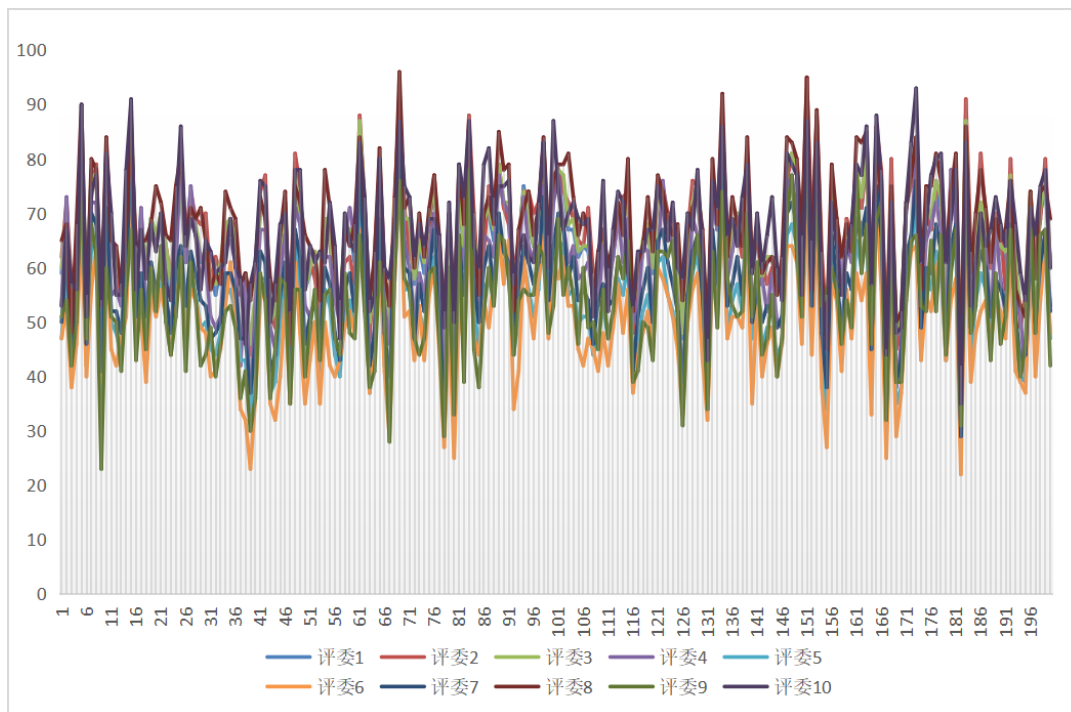


图 1: 评委打分情况

的评分线围绕平均分线做小幅波动。他们试图按照客观标准掌握评分标准，保持客观公正态度，谨慎小心，即使在平均分线上下波动，但起伏不大。从经验角度看，这类评委的评分水平比较高，主观倾向与客观实际相接近，是比较理想的评委。

2) 一致性偏高型：评委的评分线始终高于全体评委打分确定的平均分线，与平均分线呈近似平行的关系。这种评委是一种带有主观色彩的“公正”评委，有一个稳定的主观倾向在理解和掌握评分标准，坚持按偏松的倾向打分。

3) 一致性偏低型：评委的评分线始终低于全体评委打分确定的平均分线，与平均分线呈近似平行的关系。与一致性偏高型评委相似，坚持从严的倾向打分，

4) 大幅度波动型：这种类型评委的评分结果的波动性很严重。但均值与平均分线相差不大。

5) 作弊型：这种评委的评分线与平均分线的关系有明显的不规则形态。这里最常见的情况是，该评委给出的大多数对象的分数与平均分线呈一有规律的吻合，但是在少数个别分数上出现明显的跳跃，远离平均分线。似乎在有意压低其他大多数测评对象的分数，而故意抬高自己看好或是有特殊关系的个别测评对象的分数。

5.2.2 模型建立 [3]

上一部分中，根据数据变化的趋势大致分为了四类，下面就需要更为精确的堆评委进行分类
先给出如下新变量的定义：

1) 离差绝对值之和 S_a^i

以每一份答卷为对象，计算所获得的 10 位裁判打分的平均分。考虑某一位裁判打分与上述平均值之差的绝对值，在 200 份答卷中求和可得离差绝对值之和。

$$S_a^i = \sum_j |s_{ij} - \bar{s}_j| \quad i = 1, 2, \dots, 10; j = 1, 2, \dots, 200$$

上式表示第 i 个评委对第 j 份答卷的评分， \bar{s}_j 表示第 j 份答卷的平均分。

2) 均值差绝对值的均值 m_i

均值差绝对值的均值描述的是该裁判打分与作品所得平均分之间的平均差距。越大就意味着这位裁判的打分在不同作品之间分差较大。

$$m_i = \frac{S_a^i}{200}$$

3) 均值差绝对值的标准差 n_i

均值差绝对值的标准差描述的是该裁判打分与作品所得平均分之差的离散程度，包含着裁判打分围绕平均值振荡程度大小的信息。

4) 变异系数

变异系数是某一位裁判对于所有答卷评分结果标准差与裁判分均值之比。用来描述数据的离散程度。当变异系数越大时，表明数据的离散程度越显著，也即裁判的判分分差较大，在不同作品中打分有明显差异。在裁判的分类中起到一个公平指数的作用。我们认为，公正的裁判在面对作品品质呈现正态分布的情况下所做出的判分应该是相对紧凑的，不会出现大量评分明显高于或低于作品平均分的情况。所以认为，该裁判的变异系数在某个范围内时，可判定该裁判是公平的，否则为不公平的。

5) 均值差变异系数 r_i

均值差变异系数是每个评委的均值差绝对值的标准差与均值差绝对值的均值的比值，则对第 i 个评委而言，

$$r_i = \frac{n_i}{m_i}, \quad i = 1, 2, \dots, 10$$

与被用作公平指数的变异系数不同，均值差变异系数反映出不同的裁判打分的横向对比，保留着裁判与平均打分差异的信息。均值差变异系数越大，则意味着裁判打分的波动越强。所以当均值差变异系数在某个范围内时，可以认为该裁判的打分波动虽大但在允许范围之内，但当均值差变异系数超过该范围时，可以理解为该裁判经常会做出明显区别于其他正常裁判的判分，故认定为作弊。

问题转换为按照上述标准堆各个评委的打分情况进行分析，然后进行分类

计算全部评委的均值差绝对值的均值，将超过 7.49 的视为不公平型，将低于 7.4 的视为公平型。其次，对于不公平型计算均值差变异系数，以 0.5 为界限将其分为大幅波动型和作弊型，对于公平型再根据均值差均值，以 4 和 -4 为标准来区分一致偏高型和一致偏低型

然后在根据分类结果堆数据进行调整

再引入权系数 w ，权系数和所属分类有关

1) 客观公平型， $w=1$

2) 一致偏高（低）型，及波动型：

$$\text{对于评委 } i \text{ 而言, } w_i = \frac{\sum_{j=1}^n \overline{s_j}}{\sum_{j=1}^n s_{ij}}$$

3) 作弊型，权系数 $w=0$ ，相当于直接将评分作废

那么最终得出的分数调整公式如下：

$$S_j = \frac{1}{m_{eff}} \sum_{i=1}^n w_i s_{ij}$$

其中 m_{eff} 表示权系数不为 0 的评委个数，也就是有效评分个数

5.2.3 模型求解

首先利用 excel 得到各个评委给分情况，然后根据设定的指标给评委分类，并计算权系数，然后调整各个文章的得分，最终得到调整后成绩和排名的情况

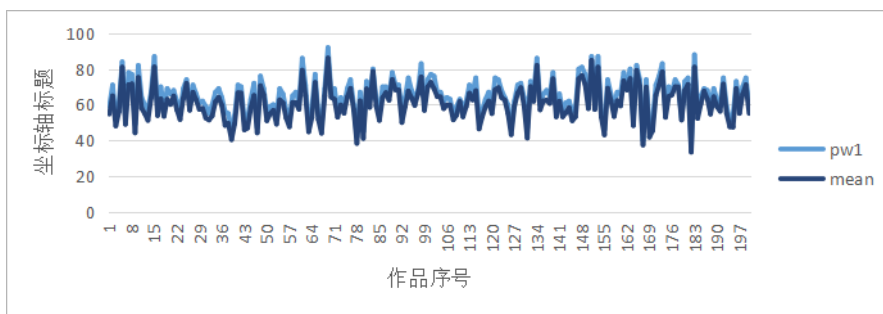


图 2: 评委 1 一致偏高

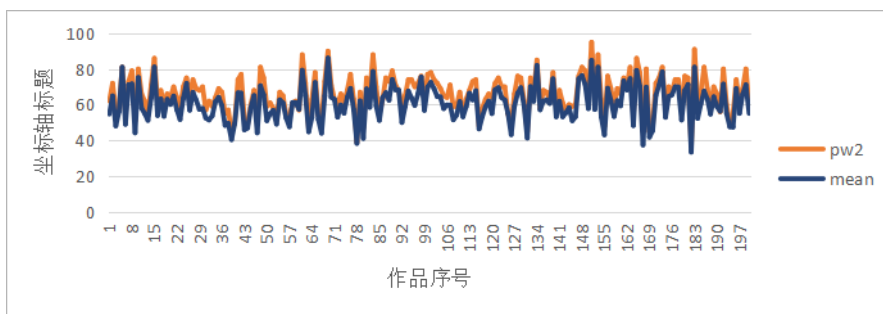


图 3: 评委 2 一致偏高

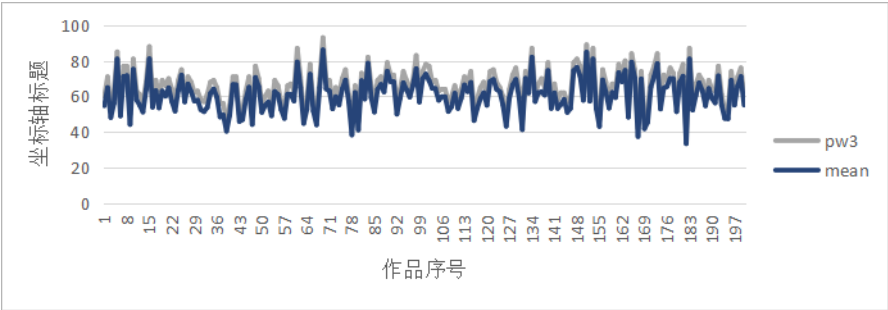


图 4: 评委 3 一致偏高

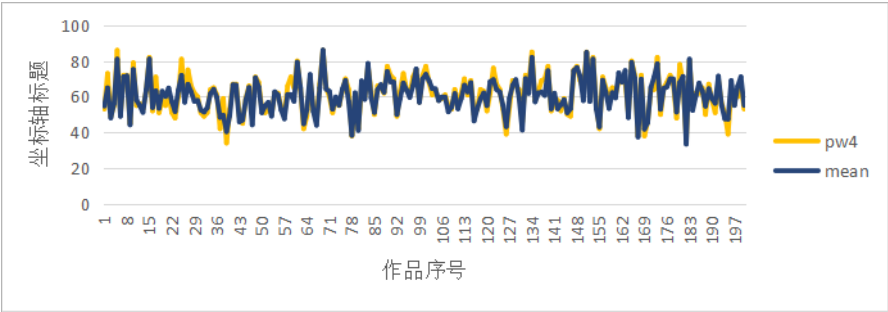


图 5: 评委 4 公正

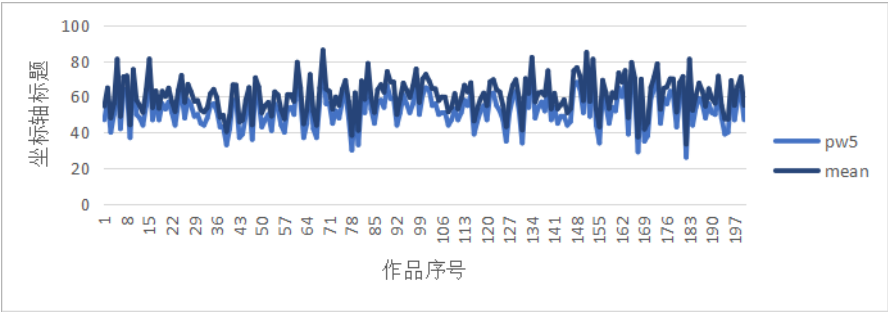


图 6: 评委 5 一致偏低

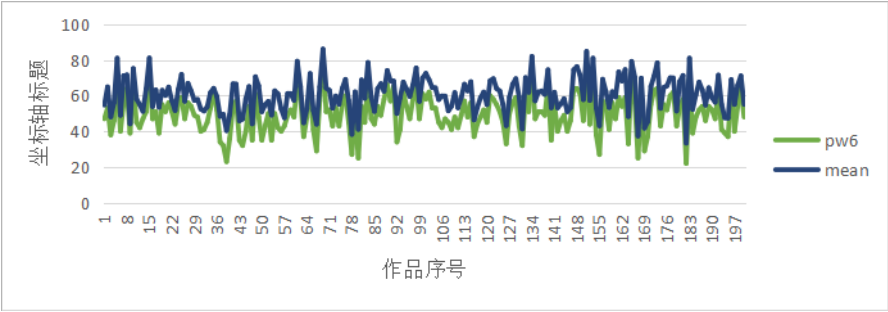


图 7: 评委 6 一致偏低且波动大

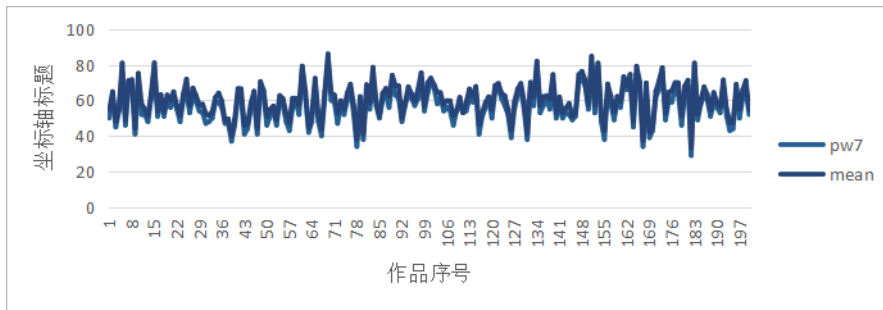


图 8: 评委 7 公正

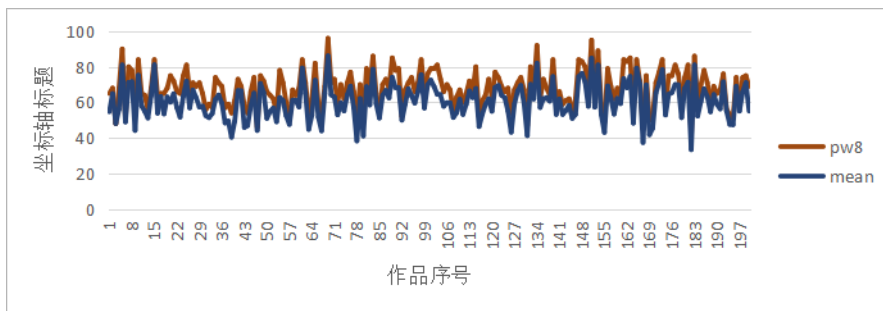


图 9: 评委 8 一致偏高

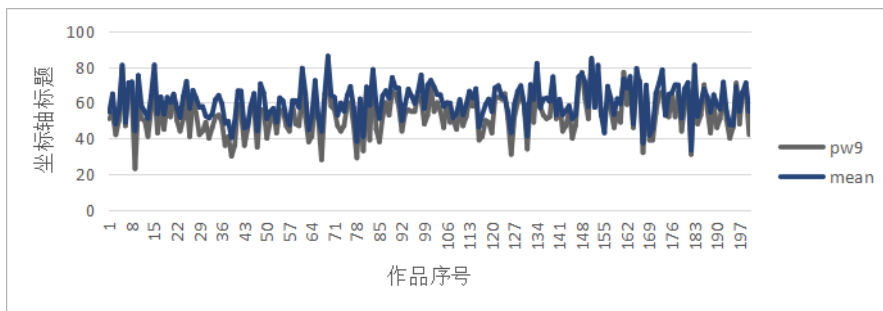


图 10: 评委 9 一致偏低

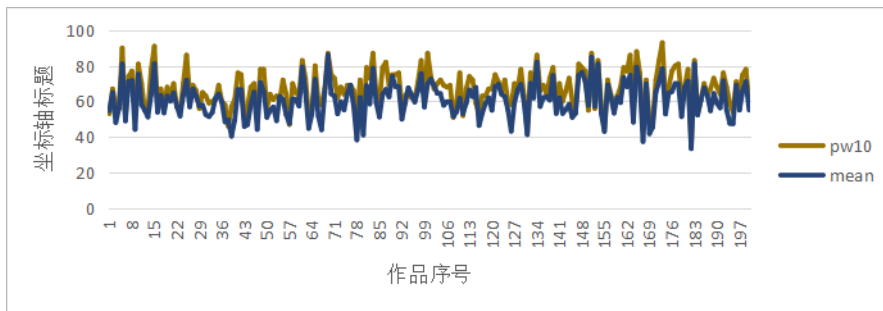


图 11: 评委 10 一致偏高且波动大

1) 公正型与非公正型

评委序号	均值差绝对值的均值	公平型 (<7.49)	不公平型 (>7.49)
1	4.4465	*	
2	5.6005	*	
3	5.2965	*	
4	2.8715	*	
5	7.7035	*	
6	10.8785		*
7	3.7685	*	
8	7.4625	*	
9	7.4975		*
10	6.5085		*

2) 公正型中客观公正型、一致偏高型、一致偏低型的细致分类

评委序号	均值差均值	客观公正型	一致偏高型 (>4)	一致偏低型 (<-4)
1	4.4465		*	
2	5.5815		*	
3	5.2965		*	
4	0.5815	*		
7	-3.7685	*		
8	7.4615		*	
10	6.2115		*	

3) 波动较大与作弊型分类

评委序号	均值差变异系数	波动较大型	作弊型
5	-0.1262	*	
6	-0.2877	*	
9	-0.6546		*

4) 评委权系数分配情况

评委序号	1	2	3	4	5
权系数	0.9316	0.9157	0.9196	1	1.1455
评委序号	6	7	8	9	10
权系数	1.218	1	1.0663	作弊型	0.907

5) 调整分数后成绩情况

排名	一等奖 论文序号	最终分数	排名	二等奖 论文序号	最终分数	排名	三等奖 论文序号	最终分数
1	69	86.54	21	8	72.2	51	27	66.89
2	151	84.03	22	25	72.09	52	186	66.78
3	134	82.16	23	101	71.81	53	41	66.66
4	15	81.91	24	149	71.61	54	87	66.5
5	153	81.23	25	48	71.44	55	42	66.44
6	183	81.13	26	177	71.11	56	128	66.4
7	5	80.25	27	172	70.96	57	62	65.8
8	61	79.94	28	192	70.92	58	113	65.79
9	165	79.63	29	100	70.86	59	171	65.65
10	173	78.83	30	181	70.58	60	49	65.61
11	83	77.85	31	199	70.51	61	97	65.52
12	98	76.28	32	7	70.49	62	2	65.38
13	10	76.04	33	132	70.14	63	175	65.26
14	148	75.64	34	102	69.78	64	46	65.23
15	163	74.43	35	122	69.6	65	198	65.09
16	89	74.31	36	166	69.6	66	68	64.95
17	139	73.86	37	76	69.42	67	176	64.73
18	147	73.83	38	156	69.4	68	35	64.71
19	65	72.85	39	168	69.12	69	86	64.43
20	161	72.55	40	129	68.87	70	189	64.23
			41	178	68.67	71	104	64.22
			42	196	68.52	72	21	64.2
			43	91	68.38	73	24	64.13
			44	94	68.35	74	70	64.05
			45	162	68.3	75	75	64
			46	121	68.28	76	95	63.97
			47	180	68.2	77	14	63.85
			48	81	67.96	78	103	63.69
			49	90	67.61	79	123	63.45
			50	115	67.19	80	137	63.32
						81	187	63.32
						82	17	63.28
						83	71	63.01
						84	88	62.94
						85	28	62.92
						86	54	62.79
						87	119	62.65
						88	133	62.5
						89	159	62.49
						90	19	62.46

表 2: 最终评奖情况

5.3 问题三模型的建立与求解

在评委不可能评阅所有的论文的实际情况下，除了评委阅卷水平参差的偏差，由于数据量的减少，评委选择而产生的随机性，也增添了打分的偏差。因此，我们需要在问题二的数学模型基础上进行适当修正，以获得更好的精度。

我们的修正方式大致如下：

(1) 步骤 1 的修正：

由于在步骤 1 中关于评委给分情况的信息比较少，不能立即给评委加权系数，暂且把权系数均设为默认值 1，所以这样设置是合理的。

(2) 步骤 2 的修正：在第二个步骤中，我们认为收集到的数据已经有了一定规模，所以在步骤二中利用同问题二中的方法给每个评委赋予不同的权重。对于每组入围论文让另两名评委评阅的分数，可根据最终分数调整公式：进行修改，由 5 名评审得分的均值得出 120 篇论文的排名，并且淘汰排名靠后的 20 篇论文。

(3) 步骤 3 的修正：依据对题意的理解，入围的排名 1-15 及 41-60 的论文都分别被 5 名评委评阅过了（虽然每篇论文对应的五名评委不尽相同），则参与步骤的另两位评审从其他 5 名中随机抽取。并且因每篇入围论文未评的 5 个评审各不相同，故随机抽取要进行 50 次。

针对每篇入围论文，随机产生参与步骤的另两位评审后再进行评分。对于这两个分数，仍依据步骤获得的第 i 位评委的权系数，仍然利用如下公式进行修正。

$$S_j = \frac{1}{m_{eff}} \sum_{i=1}^n w_i s_{ij}$$

原排名 41-60 间的论文由 7 名评审的均分排序，新排名 41-50 的获二等奖，51-60 的获三等奖；原排名 1-30 间的论文由 7 名评审的均分排序，新排名 26-30 的获二等奖，1-30 的获一等奖。

(4) 步骤 4 的修正：排名 16-25 的 10 篇论文（已由不同的 7 位评审评过），再由对应剩下的三名评审评阅。仍依据步骤获得的第 i 位评委的权系数，对这三个评委的打分，分使用数调整公式：修正。最后，这原排名 16-25 的 10 篇文章，根据 10 个评委的修正后分数的均值重新排序，在新排名下 16-20 名的获一等奖，21-25 的获二等奖。

分数调整、平均再排序后的最终结果如表三所示：

一等奖	分数	二等奖	分数	三等奖	分数
69	85.4466368	139	69.4201361	81	66.78554358
151	82.8131351	163	69.14757404	27	66.20654165
134	81.5773962	147	68.57020374	113	65.77523223
15	81.5513706	149	65.31890943	21	65.59587584
5	80.7550176	181	64.0934456	2	65.54371862
153	80.2931649	121	68.74635463	198	65.51017528
183	79.9435776	115	68.72775682	41	65.19544806
165	78.6218661	156	68.47174433	171	64.68269272
61	78.5893982	122	68.07818851	46	64.58209593
173	77.9503471	91	67.94132869	35	64.22870445
83	77.0728246	94	67.87537765	42	65.65473531
10	74.9950488	162	67.79208635	49	65.46613909
98	74.7809135	186	67.27291826	189	65.41135829
148	74.6285201	90	67.19640844	87	65.33005449
89	74.4285974	97	66.84761129	128	65.23233146
8	85.2183696	177	70.95630529	70	65.08478959
101	81.2151886	102	70.86596654	68	64.96176632
7	79.4385469	166	70.75404211	62	64.86841987
161	72.7072038	129	70.5111525	17	64.51599168
65	72.4	25	70.2646399	75	64.22351192
		132	70.18861826	176	63.98000052
		178	70.00014476	54	63.94584648
		168	69.92017559	124	63.79453527
		100	69.62975557	123	63.7853499
		76	69.56285605	104	63.69193139
		192	70.52347621	88	63.62867361
		199	70.2563103	119	63.51180735
		172	69.87620155	14	63.33939303
		48	69.46203187	24	63.31441657
		196	68.87518304	103	63.23736322
				19	62.75466855
				95	62.69582046
				187	62.69299363
				136	61.97109646
				28	61.93481223
				137	61.8715837
				159	61.83174328
				79	61.81937536
				157	61.72844725
				71	61.61855982
				58	61.03054367
				34	60.99532132
				110	60.88720828
				114	60.68466516
				141	60.48956146
				133	60.36357963
				185	60.05068898
				36	59.96778644
				107	59.96564198
				55	59.79636945

表 3: 最终评奖情况

5.4 问题四模型的建立与求解

先前的流程中，评审的公正性和评委的工作量均存在不合理性，所以本问题主要是对原评审流程的评价与改进

5.4.1 新的衡量标准

1) 完成度 C ，评分系统总完成度 C^*

在评卷过程中，评委总数为 n ，若一个对象由 $X \in \{0 \leq X \leq n, X \in Z\}$ 个评委评阅，则定义其完成度为

$$C = \frac{X}{n}$$

依据上述定义，如果一队的论文被全部评委评阅，其置信度为 1；如果只有部分评委评阅，则其置信度位于 $(0,1)$ 。依据模型二的思想，从统计学意义上，一份试卷评阅次数越多，意味着评阅完成程度越高，其评分的系统误差越小，成绩更加接近于客观水平。一般来说，完成度紧密依赖于评委的类型，其对成绩的影响量和自身改变量并不一定是线性关系，但在本题中我们只关心成绩影响关于完成度的函数的单调性，因此不妨假设变化关系是线性的。

类似的，我们可以得到一个有 N 个样门的评分 1 系统的总完成度 C^* :

$$C^* = \frac{\sum_{i=1}^n C_i}{n}$$

其中 C_i 表示第 i 个样本的完成度

2) 阶段匹配度 Q

在类似于本题的有 n 份样本, m 位评委的多过程评分系统中（每轮每组评委数相同），仅凭借总完成度无法评估某个特定过程的评分情况，因此引入阶段匹配度的概念。在第 k 个阶段结束后，设每份作品已最多有 m_k 个评委打分， n_k 份样本确定结果（淘汰或获 x 等奖），定义阶段匹配度如下：

$$Q_k = \frac{\frac{m_k}{m}}{\frac{n_k}{n}}$$

当匹配度在 1 左右时，表明评分过程中评委数量分配情况与样本数量变化情况较符合；匹配度 $Q_k < 1$ ，表明评委数量分配过少，样本评分的系统误差将会增大；匹配度 $Q_k > 1$ ，表明评委工作量增大。一般而言，匹配度序列应满足如下要求：

1. 评委分配应保持大致一致，即 Q_k 的极差/标准差应控制在一定范围。
2. 评分后期应更加严谨，即 Q_k 最好为单调不减序列。

5.4.2 原评审流程的分析和修改

(一) 本问中假设和先前模型假设略有不同，故重新罗列基本假设：

1. 每个评委的评卷速度相近；
2. 试卷编号随机分配；

3. 每位评委的阅卷质量恒定；(*)

4. 每个评委独立地评出每份答卷的分数，对于评阅同一份答卷的评委不会相互交流各自所评的分数。

5. 各个评委都是公正的，即权系数 w 均为为 1。(*)

(二) 对于现有评审标准的简单评述

那我们可以主要考虑四个方面：总完成度、阶段匹配度、评委的阅卷总量（劳动量）和单个评委的阅卷量。

本实例中的评审步骤已经大大减少了劳动量，但因此导致了总完成度和阶段匹配度的一些问题，以下分析：

$$\text{劳动量 } M = 200 \times 3 + 120 \times 2 + 10 \times 3 = 970$$

$$\text{总完成度 } C^* = \frac{80 \times 0.3 + 70 \times 0.5 + 40 \times 0.7 + 10 \times 1}{200} = 43.5$$

$$Q_1 = \frac{\frac{3}{10}}{\frac{80}{200}} = 0.75$$

$$Q_2 = \frac{\frac{5}{10}}{\frac{150}{200}} = 0.67$$

$$Q_3 = \frac{\frac{7}{10}}{\frac{190}{200}} = 0.74$$

$$Q_4 = \frac{\frac{10}{10}}{\frac{200}{200}} = 1$$

可看出，原评审流程总完成度低于 50%，并不能很好控制打分造成的系统误差。进一步分析表明，阶段匹配度序列振荡较大，无单调性；第一、二、三轮匹配度都明显低于 1，即大多数样本在评分不充分的情况下就被淘汰或决定奖项，从而使整个比赛评判的不确定性加强，降低了比赛的客观公平性，应当给予改进。

(三) 修正与改进

修改一：应适当提升评委工作量以提高总完成度，并控制阶段匹配度。具体方法为：

步骤一淘汰人数减少至 60 人；

步骤二每篇文章评委人数增至 4 人；

步骤三每篇文章评委人数增至 3 人；

步骤四重新评分，需要每一位评委评分。

另一方面，注意到组与组之间的论文总体水平存在着显著差异。例如，组 11 论文水平普遍偏低，而组 7 普遍偏高。这样，若采取原评审方法中步骤一、二的组内评分方式和步骤一的组内淘汰方式，有可能会将水准较高的论文直接在步骤一或二时就淘汰，从而降低了公正性。（推导过程见附录）

修改二：每个组的 10 篇论文根据 3 位评委的给分进行平均，总的 200 篇论文排名后，淘汰后 60 篇。未淘汰的 140 篇论文，再由没有评审过的 4 位评委进行评审（评审时采取圆桌模式），给出得分。

注：圆桌模式即让所有评委围坐在一张桌子周围，将评委按顺时针编号。找一名工作人员（非评委）从第一位评委开始分发论文，如果该论文已经被该评委评过，则以顺时针方向传递，直到找到未评审过

该论文的评委为止。一轮分配完成后，即每位评委均拿到论文评审时，同时开始评阅，由于假设 2 中评委评审速度相近，每轮各个评委的评审工作可视为同时开始同时结束，这就保证了最后工作量大体均衡。

对 140 篇论文根据 7 个评委的平均给分进行总体排序，淘汰排名靠后的 40 篇文章，剩下 100 篇论文为获奖论文。其中 71~100 名评为三等奖，31~40 评为二等奖。1~30 名和 41~60 名进入下一轮。

(*)：实际评分中，(*) 假设很难满足，由此带来两个问题：

1) 大规模评分时评委出现状态波动，导致个别评分偏离客观水平，由于样本量较大，问题二中的模型很难检测出异常并修正。

2) 对于大幅波动型的评委，模型二中的权系数对评分的修正效果有限，从而影响评分的公正性。

5.4.3 评审流程最终修改结果

准备：从 10 名评委中选出两位资深评委（公正型）作为裁判组成员，两名裁判可互相交流，最终评出分数；将 200 份文章分成 20 组，每组 10 篇。设置“异常卷”条件，如极差大于 10 或标准差大于 6。

步骤 (1)

打分：每组的 10 篇论从余下 8 位评委中选取 3 位评委，评审每组的 10 篇论文，分别用百分制给出评分。若评分满足异常卷，则调至裁判组评判，用裁判组分数代替原平均分。评委的分配方式使用问题一中第三小问最优分配规划得到的分配方案。

排序：每个组的 10 篇论文根据 3 位评委的给分进行平均，总的 200 篇论文排名后，淘汰后 60 篇。

步骤 (2)

打分：未淘汰的 140 篇论文，再由没有评审过的 4 位评委进行评审（评审时采取圆桌模式），给出得分，异常卷同步骤 (1)。

排序：对 140 篇论文根据 5 个评委的平均给分进行总体排序，淘汰排名靠后的 40 篇文章，剩下的 100 篇论文为获奖论文。

100 篇获奖论文中，排名 31 到 40 的论文，获二等奖；排名 71~100 的论文获三等奖；

步骤 (3)

打分：排名 1~30 与排名 41~70 的 60 篇论文，由 1 位未评审过的评委和裁判组两位评委分别打分（此步骤裁判不能交流）。

排序：原排名 41~70 的 20 篇论文，计算得到均分后排名，新排名 41~50 的获二等奖，新排名 51~70 的获三等奖；

原排名 1~30 的 30 篇论文，计算得到均分后排名，新排名 1~15 名获一等奖。

步骤 (4)

打分：步骤 (3) 中平均分排名在 16 到 30 的论文，直接交给十位评委重新打分得到新的排名。

排序：最终排名 16~20 的获一等奖，最终排名 21~30 的获二等奖。

六、模型的评价、改进与推广

6.1 模型的优点

A. 用于解决问题一前两问的模型均有可靠的数学依据，非常严谨，求解模型得到的最优解几乎无法再进行优化

B. 考虑的模型假设数量有限，建立的模型都具有一定的实际意义

C. 运用统计学的原理，根据不同的评分特点把评委分成不同类型，再用层次分析法来检验评委的公平性，使得问题的描述比较清晰，并且根据不同类型赋予不同权系数，更有说服力

D. 问题三的模型仅依赖于部分数据，最终得出的排名和依据全部数据得到的结果比较相似，说明该模型适用性较好，在带有随机性的同时保证了不错的稳定性

E. 问题四中引入了圆桌模型和裁判复合模型，可以保证工作的均衡性，同时也能使评审更为高效

6.2 模型的缺点

A. 由于编程能力有限，采用的算法都很朴素，导致时间复杂度较高，数据量庞大时效率低下

B. 通过模型无法真正解决评委评分公平性的问题，模型即使有一定的实际意义，但具有更多局限性，要想从根本上解决评审公平问题，需要提高评委群体乃至整个社会的诚信观念和道德素养。

C. 没有考虑评委阅卷过程中疲劳程度的变化，从实际角度出发随着评卷进行，评委会更加疲劳，往往更容易草率地打出一个习惯性给出的分数，模型并没有考虑这一点

6.3 模型的改进

可以尝试采用更为高效的算法

可以适当修改模型，减少冗余部分

可以引入一些函数来模拟各种特殊因素（如评委疲劳程度）对于给分的影响，使得模型更加全面

6.4 模型的推广

本文中的模型不知可以运用于竞赛评分，多轮决策也可以使用，本文中模型的要点在于适应性，可以根据先前的评分结果调整评委的权系数，使得评审更加公平。在多轮决策中，着就提心为一个人在每一轮决策中的话语权（权系数）会根据之前评分的结果不断变化，属实比较合理

参考文献

- [1] Skolem, T. (1958). Some Remarks on the Triple Systems of Steiner. MATHEMATICA SCANDINAVICA, 6, 273–280.
- [2] D. R. Stinson, Combinatorial Designs: Constructions and Analysis, Springer, 2003
- [3] 吕书龙, 梁飞豹, 刘文丽. 关于评委评分的评价模型 [J]. 福州大学学报 (自然科学版), 2010, 38(03): 358-362.

附录 1

参考文件列表:

<https://github.com/tystys404/2022njumathematicalmodeling>

附录 2

问题一求解最优分配方案 lingo 程序

```
Lingo Model - best_match
model:
data:
pw=10; !评委组数;
zushu=20; !论文组数;
enddata

sets:
critics/1..pw/;
compositions/1..zushu/;
links(critics,compositions):x;
endsets

min=@max(critics(i):@sum(compositions(j):x(i,j)));!任意不同的两个组的评委尽量不同;
@for(compositions(j):@sum(critics(i):x(i,j))=3);!条件约束;
@for(links:@bin(x));
end
```

问题三权重控制代码

```
1 function quan = quanhzong(data, reference)
2     quan = ones(10,1);
3     mean_data = zeros(200,1);
4     mean_caipan = zeros(10,1);
5     std_caipan = zeros(10,1);
6     for i = [1:200]
7         data_i = data(i,:);
8         temp = reference(i,:) == 1;
9         data_i = data_i(temp);
10        mean_data(i) = mean(data_i);
11    end
12    for i = [1:10]
13        caipan_i = data(:,i);
14        temp = reference(:,i) == 1;
15        caipan_i = caipan_i(temp);
16        mean_caipan(i) = mean(caipan_i);
17        std_caipan(i) = std(caipan_i);
18    end
```

```

19
20 %GBKhei 变异系数
21 bianyi = std_caipan ./ mean_caipan;
22
23 %GBKhei 均值差绝对值的均值
24 mean_abs_mean = zeros(10,1);
25 for i = [1:10]
26     mean_abs_mean(i) = mean(abs(mean_data-mean_caipan(i)));
27 end
28 %GBKhei 均值差的均值
29 mean_mean = zeros(10,1);
30 for i = [1:10]
31     mean_mean(i) = mean(-mean_data+mean_caipan(i));
32 end
33 %GBKhei 均值差的标准差
34 mean_std = zeros(10,1);
35 for i = [1:10]
36     mean_std(i) = std(-mean_data+mean_caipan(i));
37 end
38
39 %GBKhei 均值差变异系数
40 mean_bianyi = mean_std ./ mean_mean;
41
42
43
44 for i = [1:10]
45     if mean_abs_mean(i) <= 8
46         if abs(mean_mean(i)) <= 4
47             quan(i) = 1;
48         else
49             temp = mean_data;
50             xuhao = temp(reference(:, i) == 1);
51             quan(i) = mean(xuhao) / mean_caipan(i);
52         end
53     else
54         if abs(mean_bianyi(i)) < 0.6
55             temp = mean_data;
56             xuhao = temp(reference(:, i) == 1);
57             quan(i) = mean(xuhao) / mean_caipan(i);
58         else
59             temp = mean_data;
60             xuhao = temp(reference(:, i) == 1);
61             quan(i) = mean(xuhao) / mean_caipan(i);
62         end
63     end
64 end

```

问题三整体计算流程代码

```
1 clear all;
2
3 data = xlsread("C:\Users\GSY\Desktop\data.xlsx");
4 data = data(:,2:end);
5 caipan = xlsread("C:\Users\GSY\Desktop\caipan.xlsx");
6 caipan(isnan(caipan))=0;
7 caipan = caipan(:,2:end);
8 caipan = sui(ji(caipan));
9 reference = zeros(200,10);
10
11 %GBKhei 第一次的权重
12 for i = [1:20]
13     temp = [1:10];
14     step1 = temp(caipan(i,:) == 1);
15     reference(i*10-9:i*10, step1) = 1;
16 end
17 quan_1 = quanhzong(data, reference);
18
19 num_list_after_1 = zeros(20,6);
20 score_list_after_1 = zeros(200,1);
21 for i = [1:20]
22     zhishi = caipan(i,:) == 1;
23     index = 0;
24     score_list_1 = zeros(10,1);
25     for j = zhishi
26         index = index+1;
27         if j == 1
28             score_list_1 = score_list_1 + data(1+(i-1)*10:i*10, index);
29         end
30     end
31     [~,IX] = sort(score_list_1, 'descend');
32     num_list_after_1(i,:) = IX(1:6)' + (i-1)*10;
33     score_list_after_1(1+(i-1)*10:i*10) = score_list_1;
34 end
35 num_list_after_1 = reshape(num_list_after_1, 120, 1);
36
37
38 %GBKhei 第二次的权重
39 for i = [1:120]
40     temp = [1:10];
41     hang = num_list_after_1(i);
42     zubie = ceil(hang / 10);
43     step2 = temp(caipan(zubie,:) == 2);
44     reference(hang, step2) = 1;
45 end
46 quan_2 = quanhzong(data, reference);
47
48 %GBKhei 第二次打分
49 [score_list_after_2, num_list_after_2] = suafan(data, reference, quan_2, num_list_after_1);
```

```

50 [~,IX] = sort( score_list_after_2 , 'descend');
51
52 %GBKhei 第三次打分
53 num_100 = IX(1:100);
54 num_100 = num_list_after_2(num_100);
55 sec1 = num_100(31:40);
56 score_sec1 = score_list_after_2 (IX(31:40));
57 third1 = num_100(61:100);
58 score_third1 = score_list_after_2 (IX(61:100));
59
60 num_30 = num_100(1:30);
61 %GBKhei score_list_before_30 = score_list_after_1(num_30);
62 num_20 = num_100(41:60);
63 %GBKhei score_list_before_20 = score_list_after_1(num_20);
64
65 %GBKhei 第三次权重
66 num_3 = [num_30;num_20];
67 for i = [1:50]
68     temp = [1:10];
69     hang = num_3(i);
70     zubie = ceil(hang / 10);
71     step3 = temp(caipan(zubie,:) == 3);
72     reference(hang,step3) = 1;
73 end
74 quan_3 = quanzhong(data, reference);
75
76 [ score_list_after_3_30 , num_30] = suafan(data, reference , quan_3, num_30);
77 [ score_list_after_3_20 , num_20] = suafan(data, reference , quan_3, num_20);
78
79
80 [~,IX1] = sort( score_list_after_3_30 , 'descend');
81 fir1 = num_30(IX1(1:15));
82 score_fir1 = score_list_after_3_30 (IX1(1:15));
83 sec2 = num_30(IX1(26:30));
84 score_sec2 = score_list_after_3_30 (IX1(26:30));
85 [~,IX2] = sort( score_list_after_3_20 , 'descend');
86 sec3 = num_20(IX2(1:10));
87 score_sec3 = score_list_after_3_20 (IX2(1:10));
88 third2 = num_20(IX2(11:20));
89 score_third2 = score_list_after_3_20 (IX2(11:20));
90
91
92
93 %GBKhei 第四次打分
94 num_list_before_4 = num_30(IX1(16:25));
95 %GBKhei 第四次权重
96 for i = [1:10]
97     temp = [1:10];
98     step4 = temp(caipan(i,:) == 4);
99     hang = num_list_before_4(i);
100     zubie = ceil(hang / 10);
101     reference(hang,step4) = 1;
102 end

```

```

103 quan_4 = quanzhong(data, reference );
104
105 score_list_after_4 = sum(quan_4.*data(num_list_before_4,:),2) / 10;
106 [~,IX] = sort( score_list_after_4 , 'descend');
107 fir2 = num_list_before_4 (IX(1:5));
108 score_fir2 = score_list_after_4 (IX(1:5));
109 sec4 = num_list_before_4 (IX(6:10));
110 score_sec4 = score_list_after_4 (IX(6:10));
111
112 fir = [ fir1 ; fir2 ];
113 score_fir = [ score_fir1 ; score_fir2 ];
114 third = [ third2 ; third1 ];
115 score_third = [ score_third2 ; score_third1 ];
116 sec = [ sec4;sec3;sec1;sec2];
117 score_sec = [ score_sec4;score_sec3;score_sec1;score_sec2 ];

```

问题三结合权重与数据的计算代码

```

1 function [ score , index ] = suaфан ( data , reference ,quanzhong,index)
2     N = length(index);
3     score = zeros(N,1);
4     for i = 1:N
5         temp = reference (index(i) ,:) == 1;
6         num = sum(temp);
7         score(i) = sum(temp .* data(index(i) ,:) .* quanzhong') / num;
8     end
9 end

```

除了支撑材料的文件列表和源程序代码外，附录中还可以包括以下内容：

- 某一问题的详细证明或求解过程；
- 自己在网上找到的数据；
- 比较大的流程图；
- 较繁杂的图表或计算结果