

摘 要:

本文运用相关分析、综合评价和建立模型等方法解决了足球队员体能的评价问题。

对于问题一，首先通过热卡填补法和均值替换法处理了表中的缺失和异常数据，并依据所给评分标准和评分权重表，分别计算出每位运动员的最终体能综合得分，每个团队的不合格球员名单以及整个团队的综合得分。

对于问题二，首先通过计算对原有及格线进行合理性分析，在保证尽可能满足题目要求“从每个足球队合格的人数中选取 22 人”的前提下，通过引入新定义的变量“有效贡献率”来逐一对候选及格线进行筛选，以确认最终的及格线。

对于问题三，采用层次分析法，通过成对比较矩阵建立各体能指标与不同球场位置之间的关系，重新分配各项体能指标的权重，并验证其是否满足一致性检验，最终建立基于球场位置特征的评价模型。

对于问题四，以问题三新建立的评价模型为基础，运用正态分布的概念对原有测试方法进行合理性分析。然后，通过采用启发式算法中的爬山算法建立优化模型，对原有测试方法进行调整和改进，使其最终数据符合正态性检验。

本文综合利用相关系数矩阵、基于层次分析法的综合评价、基于启发式算法的优化模型等方法，结合 Python、MATLAB、Excel 等软件，对足球运动员体能的评价问题进行了多角度的分析，给出了更加合理的足球运动员的体能测试方法和评价模型，并在最后对其进行了评价与讨论。

关键字： 足球 成绩评价 相关系数 层次分析法 正态分布 启发式算法 爬山算法

目录

1. 问题重述	3
1.1 问题背景	3
1.2 需要解决的问题	3
2. 问题分析	4
2.1 问题一的分析	4
2.2 问题二的分析	4
2.3 问题三的分析	4
2.4 问题四的分析	4
3. 模型假设	4
4. 符号说明	5
5. 模型的建立与求解	5
5.1 问题一	5
5.1.1 数据的预处理	5
5.1.2 每位运动员的最终体能综合得分	8
5.1.3 每个球队中不合格队员的名单及成绩	9
5.1.4 每个球队的综合评分	9
5.2 问题二	9
5.2.1 对合格线的合理性分析	9
5.2.2 体能综合得分及格线的设定	10
5.3 问题三	12
5.3.1 建立结构层次图	12
5.3.2 确定准则层对目标层的权重向量	13
5.3.3 建立具体测试项目的权重表	14
5.3.4 建立各位置球员的项目权重表	14
5.4 问题四	14
5.4.1 对现有体能测试方法的评价	14
5.4.2 离散优化	15
5.4.3 启发式算法中的爬山算法	15
5.4.4 算法实现	16
5.4.5 结果展示	17
5.4.6 基于不同年龄段球员的应用	19
6. 模型的评价	20
6.1 模型的优点	20
6.2 模型的缺点	20
7. 参考文献	20
附录 A 表格	21
1.1 问题一附录	21
1.1.1 各球员的各项及综合体能得分	21
1.1.2 相关系数	31
1.1.3 所有球员的总成绩排序	32

1.1.4 各队不合格成员名单	40
1.2 问题二附录.....	42
1.2.1 候选及格线的生成	42
附录 B 代码	44
2.1 问题一代码.....	44
2.2 问题三代码.....	46
2.3 问题四代码.....	47

1. 问题重述

1.1 问题背景

对参加足球比赛的运动员进行身体素质测试。一方面收集特定年龄段球员的身体及运动能力数据,为未来建立运动员线性成长追踪模型提供参考依据,也为后续测试工作的开展积累经验,逐步建立各级年龄段专项测试数据标准。另一方面作为参赛球队排名的衡量依据之一,为各参赛球队的身体情况进行综合打分排名。

1.2 需要解决的问题

附表数据(文件名: tongjimcm2021d 体能数据)是 15 个足球赛队的体能测试数据,每个球队提供了最多 35 名候选运动员的体测数据,每个队需要从中挑选出合格的 22 人组成正式的比赛球队. 体能测试的标准见文件 (tongjimcm2021d 测试标准.docx). 请根据要求完成下面的工作

1. 按照下列评分权重表,

项目	箭头跑	30 米跑	引体向上	纵跳	立定跳远	YOYO 得分
权重	15%	15%	10%	10%	10%	40%

表 1 评分权重表

- a) 给出每位运动员的最终体能综合得分,处理其中缺失和可能不一致的数据;
 - b) 给出每个球队中不合格队员(体能综合得分小于 6 分)的名单及成绩;
 - c) 每个球队的综合评分(综合评分为参加测试球员的总分之和除以参加测试球员人数)。
2. 一般规定每个球队最多可以有 35 个队员参加体能测试,通过体能测试的队员最多可以有 22 个队员参加最终比赛名单。如果按所有参赛球员的 85% 作为合格线,请问最终体能综合得分的及格线为多少比较合理? 哪些球队可能无法挑选出 22 人?
 3. 对于不同场上位置的球员,问题 1 的权重表是否合理? 某些球队候选的几名守门员都不能顺利通过体测,从而造成该队没有守门员的窘境. 对于守门员这个体测标准是否过于严格? 更进一步,能否针对不同位置的球员,根据他们位置的特征设置合理的评分权重? 试着说明其合理性.
 4. 这些球员有相当一部分以后可能成为中国职业球员,你们可否从球员成长的角度出发,说明目前的测试方式是否可以作为球员体能的测试标准? 这个测试方法及其中的参数是否需要改进和提高,以及如何改进和提高? 请给出具体的模型和解释。

2. 问题分析

2.1 问题一的分析

问题一要求计算每位运动员的最终体能综合得分、每个球队的综合评分，并给出每个球队中不合格队员的名单及成绩。在依照所给评价标准和权重表进行计算前，需要先对所给数据中的缺失和异常部分进行预处理，以减小计算误差。

2.2 问题二的分析

问题二是需要根据现有球员的成绩，以所有参赛球员的 85% 为合格线，对最终体能综合得分的及格线进行划分。题目的最终目的是要求从每个足球队合格的人数中选取 22 人，而所有参赛队员的前 85% 分配到 15 个队后，并不能确保绝大多数的队伍都有 22 位及以上的合格队员。因此需要重新对体能综合得分的及格线进行划分，在保证区分度的情况下，尽可能多地让各队及格人数达到 22 人。

2.3 问题三的分析

问题三是对第一问中的权重表进行合理性分析，并针对球场上不同位置的特征设置更加合理的评分权重。该问题的最终目的是建立一个评价模型，且对不同位置的情况有各自的评价标准。我们选择采用层次分析法，依据五种身体素质对不同球场位置的重要性排序，重新计算出各项指标所占权重，从而建立全新的评价模型。

2.4 问题四的分析

问题三首先需要建立一个评价系统，以一个合适的标准来对现有的测试方法进行评价。其次，需要根据上面的评价结果，在第三问所优化的模型基础上对该测方法进行提高和改进，即重新对成绩标准中的分数进行划分，使得最终的测试方法达到评价系统的合格标准。

3. 模型假设

1. 个人得分：全部缺失按 0 处理，部分缺失热卡填补法
2. 项目成绩中，立定跳远？其余项目？
3. 经检查发现：第 10 队和第 6 队数据完全一致，故将两队认定为同一组数据

4. 符号说明

C	表示变量球员 30M 跑成绩
D	表示变量球员箭头跑成绩
E	表示变量球员立定跳远成绩
F	表示变量球员纵跳成绩
G	表示变量球员引体向上成绩
H	表示变量球员 YOYO 成绩
X	表示变量 X 剔除缺失数据后的的变量
$\rho(X, Y)$	表示变量 X 与变量 Y 的相关系数，数值大小反映了 X 与 Y 的相关性
CI	一致性检验
C_i	层次分析法中的第 i 个因素
A	正互反矩阵
λ_{max}	最大特征根
RI	随机性一致性指标
CR	一致性比率
w	权重向量
σ	标准差
μ	均值
$norm$	正态分布方程
$totalNumber$	参与测评的总人数
$loss_function$	损失函数
lr	学习率
$epochs$	学习轮数

表 2 符号表

5. 模型的建立与求解

5.1 问题一

5.1.1 数据的预处理

经过对数据的查找，我们发现原始数据存在异常，另有些类型的数据存在缺失，因此需要预先将其正常化处理。题中要求分别求运动员个人得分和球队综合评分，对于这两种情况我们决定分别采用两种不同的数据处理方式。

(1) 对于运动员个人得分的处理

考虑到个人得分体现的是运动员自身的身体素质，无法由队内其他成员的平均值来代替，故采取**热卡填补法**来处理异常或缺失数据。热卡填补法是指在数据库中找到一个与它最相似的对象，然后用这个相似对象的值来进行填充。我们采用相关系数矩阵来确定与缺失值所在变量（如变量 X ）最相关哪个变量（如变量 Y ）。然后将所有变量 Y 按取值的大小进行排序，找到缺失值所对应的变量 Y 的值，用与该值相等或接近的变量 Y 所对应的变量 X 的值代替缺失值。

以第十一组 7 号球员为例：

该球员在箭头跑的成绩（ D ）上有缺失，我们首先除去 7 号球员，对第十一组的剩余队员的数据进行处理，分别计算 30M 跑成绩与箭头跑成绩的相关系数

$\rho(C^*, D^*)$ 、立定跳远成绩与箭头跑成绩的相关系数 $\rho(E^*, D^*)$ 、纵跳成绩与箭头跑成绩的相关系数 $\rho(F^*, D^*)$ 、引体向上成绩与箭头跑成绩的相关系数 $\rho(G^*, D^*)$ 、YOYO 成绩与箭头跑成绩的相关系数 $\rho(H^*, D^*)$ ，相关系数 $\rho(X, Y)$ 的计算公式如下：

$$\rho(X, Y) = \frac{\Sigma XY - \frac{\Sigma X \Sigma Y}{N}}{\sqrt{\left(\Sigma X^2 - \frac{(\Sigma X)^2}{N}\right) \left(\Sigma Y^2 - \frac{(\Sigma Y)^2}{N}\right)}}$$

其中，N 为 X 和 Y 变量所对应的样本数量。

根据计算公式我们可得：

$\rho(C^*, D^*)$	0.3247088
$\rho(E^*, D^*)$	-0.006612
$\rho(F^*, D^*)$	0.145459
$\rho(G^*, D^*)$	-0.08486
$\rho(H^*, D^*)$	0.1332599

表 3 11 组其他变量与变量 D 的相关系数表

比较可得表中 $\rho(C^*, D^*)$ 的值最大，因为相关系数大小反映两个变量之间的相关性，所以 30M 跑成绩与箭头跑成绩的相关性最高，故而我们采用 30M 跑成绩来估计缺失的箭头跑成绩。

首先我们对于第 11 组的所有球员的成绩按照 30M 跑成绩 (C) 进行排序，找出与 7 号球员在 30M 跑成绩上与之最相近的成员，并找出他 (们) 的箭头跑成绩，结果如下：

球队编号	球员编号	30M 跑成绩	箭头跑成绩
11	7	9	缺失
11	26	9	10
11	29	9	10
11	30	9	10
11	31	9	10
11	12	9	6
11	22	9	6
11	15	9	5
11	17	9	5
11	2	9	4
11	1	9	3
11	4	9	3
11	11	9	3
11	21	9	3
11	9	9	0

表 4 11 组与 7 号球员 30M 跑成绩最接近的成绩表

然后我们取他（们）在箭头跑成绩的平均值作为对于缺失数据（7号球员的箭头跑成绩）补充数据。由计算可得结果为 5.57，根据箭头跑成绩得分均为整数，故将 6 作为 7 号球员的箭头跑成绩的补充数据。

(2) 对于整个球队综合评分的处理

考虑到某些球员的极端数据会极大影响整个球队的评分，且热卡填补法所填充的数据为估计值，使评分产生较大的误差，故采用均值替换法来处理球员的异常或缺失数据。均值替换法是将该球队剔除异常数据后取整剩余数据的平均值来替换异常或缺失数据。

以第 11 组为例：

第 11 组的 7 号球员在箭头跑成绩上有缺失，所以我们首先计算第 11 组剩余成员的箭头跑成绩的均值，11 组箭头跑成绩如下表：

球队编号	球员编号	箭头跑成绩
11	1	3
11	2	4
11	3	6
11	4	3
11	5	3
11	6	7
11	7	缺失
11	8	4
11	9	0
11	10	4
11	11	3
11	12	6
11	13	3
11	14	2
11	15	5
11	16	5
11	17	5
11	18	3
11	19	5
11	20	4
11	21	3
11	22	6
11	23	4
11	24	2
11	25	6
11	26	10
11	27	10
11	28	10
11	29	10
11	30	10
11	31	10

表 5 11 组球员箭头跑成绩

由表可计算除去 7 号的剩余球员的箭头跑成绩均值结果为 5.2，故用 5.2 作为 7 号球员箭头跑成绩去计算 11 组的总成绩，根据各项得分权值重新计算 7 号球员的成绩，最后再计算第 11 组所有成员的成绩的均分。

5.1.2 每位运动员的最终体能综合得分

首先根据附件（测试标准.docx）将每位运动员的各项体测数据转换为 10 分制的成绩，然后在由题中所给权重表计算运动员最终的综合成绩。整个计算过程使用 Python 对数据进行处理，处理结果见附录表“1a 各队成员综合得分”。

5.1.3 每个球队中不合格队员的名单及成绩

基于以上数据，以 6 分作为体能综合得分的及格线，将所有球员分为“合格”及“不合格”两类。其中，各队中不合格队员的名单及成绩见附件表“1b 各队不合格成员名单”。

5.1.4 每个球队的综合评分

球队的综合评分为参加测试球员的总得分之和除以参加测试球员人数，其各队的最终结果如下表：

球队编号	综合评分
1	6.61
2	7.24
3	7.00
4	6.76
5	6.33
6	4.76
7	6.61
8	8.04
9	6.95
11	7.78
12	6.99
13	8.39
14	6.75
15	7.66

表 6 每个球队的综合评分

5.2 问题二

5.2.1 对合格线的合理性分析

合格线是指设定一个分数线使全部球队的球员人数的 85% 的人能达到，首先，以个人得分对所有参赛球员进行排序，以 85% 为合格线将球员分为“合格”和“不合格”两类。其次，将前 85% 的球员分配到原有的各个球队中，并统计各球队的合格人数，结果如下表：

球队编号	成绩在 85% 以内人数
1	21
2	25
3	21
4	18
5	20
6	4
7	18
8	27
9	20
11	31
12	18
13	24
14	21
15	27

表 7 各球队达到合格线人数表

由上表可知：14 个球队中，满足“合格人数至少为 22 人”条件的队伍仅有 5 支，分别为第 2 队、第 8 队、第 11 队、第 13 队、第 15 队。由于满足条件的队伍在队伍总数的占比中不足 50%，与题目中“从每个队伍中挑选出合格的 22 人组成正式的比赛球队”这一目标相差过大，认为该合格线的标准过高，应当针对体能综合得分来设定更加合理的及格线分数。

5.2.2 体能综合得分及格线的设定

原有及格线分数为合格线 85% 所对应球员的个人得分，即 5.90 分。新及格线是将原有及格线进行下调，使部分不合格球员变为合格球员，因此研究对象应为合格线 85% 以下的所有球员。

以“球队合格人数是否满足 22 人”作为判断标准，对不合格球员按照体能综合成绩从高到低逐一进行判断：若对于新及格分数线，该球员恰好为球队的第 22 位合格球员，则将该球员所对应的分数作为候选及格线。

然后，分别对每个候选及格线计算其人员**有效贡献率**。人员有效贡献率的定义为：对于新及格线，所有新增的合格球员中，使球队满足“合格人数至少为 22 人”条件的球员数（有效贡献人数）占有新增合格球员的比例。

有效贡献率公式为：

$$\text{有效贡献率} = \frac{\sum_{i \in P} \text{合格人数达到 22 人的球队编号的集合 } i \text{ 队的有效贡献人数}}{\text{合格线到该候选及格线的人数}}$$

以第二条候选及格线（5.45 分）为例：

根据有效贡献人数的定义我们可以知道，合格线以上就满足 22 个人的球队（第 2、8、11、13、15）在计算有效贡献率的时候其队伍有效贡献人数均为 0。

在刚好满足 12 队能选出 22 个人的分数线以上，且在合格线以下的名单如下：

序号	球队编号	体能综合成绩	及格线选取可能值	选取该及格线的人员有效贡献率
1	5	5.85		
2	4	5.85		
3	6	5.80		
4	6	5.80		
5	15	5.80		
6	1	5.70	1 队满 22 人	0.17
7	6	5.70		
8	12	5.60		
9	4	5.60		
10	2	5.55		
11	12	5.55		
12	12	5.55		
13	12	5.45	12 队满 22 人	0.38

表 8 以 12 队满 22 人作为及格线贡献率计算表

可以看出，在选取 12 队刚好能选出 22 个人的分数线为及格线的情况下，合格人数达到 22 的球队为第 1 队和第 12 队，第 1 队的有效贡献人数为 1（表中第 6 行），第 12 队的有效贡献人数为 4（表中第 8、11、12、13 行），故代入公式可算得有效贡献率 = $(1 + 4)/13 = 0.38$ 。

经计算，所有候选及格线及其人员有效贡献率的数据如下表：

序号	体能综合成绩	选取该及格线的人员有效贡献率	候选标准
1	5.70	0.17	1 队满 22 人
2	5.45	0.38	12 队满 22 人
3	5.45	0.44	5 队满 22 人
4	5.30	0.40	14 队满 22 人
5	5.05	0.38	9 队满 22 人
6	5.00	0.52	4 队满 22 人
7	4.50	0.39	3 队满 22 人
8	3.40	0.69	6 队满 22 人

表 9 候选及格线的有效贡献率表

由上表可知，第八条线所对应的人员有效贡献率最高，为 0.69，其所对应的判断条件为“第 6 队合格球员满足 22 人”。然而，该条线以上的有效贡献人数为 33 人，其中第六队球员高达 18 位。若以此线作为及格线，一方面所有球员中仅有 5 人为“不合格”，选取第 8 条线对于球员的区分度不高；另一方面，第 6 队在 85% 对应的合格线下载标仅有 4 人，且考虑第 6 队的综合评分为 4.76，与其他组相差较大，选取第 8 条线对于球队的区分度不高，综合两方面考虑，我们在候选及格线中剔除第 8 条线。

除第八条线（3.40 分）以外，第六条线（5.00 分）所对应的人员有效贡献率最高，为 0.52。故选择第六条线所对应体能综合成绩 5.00 分作为新的及格线。新及格线划分后的各队合格人数如下表所示：

球队编号	合格成员人数
1	23
2	26
3	21
4	22
5	26
6	11
7	19
8	27
9	22
11	31
12	22
13	24
14	22
15	28

表 10 各个球队达到新及格线人数表

由上表可知，14 个球队中，满足条件的队伍数增加到 11 队，占队伍总数的 78.57%，说明该及格线能保证绝大多数队伍满足题目中的目标要求；无法满足目标要求的队伍，即无法挑选出 22 名参赛球员的队伍为：第 3 队、第 6 队、第 7 队（该队总人数不满 22 人）。

5.3 问题三

5.3.1 建立结构层次图

使用层次分析法对本题进行建模分析，以前锋为例：

1. 第一层为目标层：选拔优秀的足球前锋、中场、后卫、守门员。
2. 第二层为准则层，通过题目所给的六个项目对应的五种身体素质，作为选拔运动员的五个因素。
3. 第三层为方案层：参选的 n 名队员。

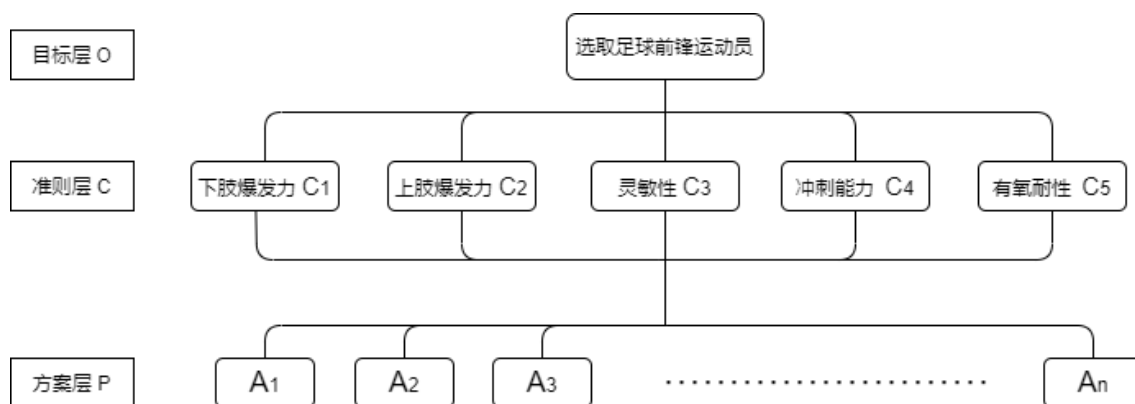


图 1 结构层次图

5.3.2 确定准则层对目标层的权重向量

首先，根据成对比较法和 1-9 比较尺度构造成对比较矩阵。由参考文献可知 1-9 尺度定义表如下表所示：

表 5.3 1~9 比较尺度定义表	
尺度 a_{ij}	含义
1	C_i 与 C_j 的影响相同
3	C_i 比 C_j 的影响稍强
5	C_i 比 C_j 的影响强
7	C_i 比 C_j 的影响明显的强
9	C_i 比 C_j 的影响绝对的强
2, 4, 6, 8	C_i 与 C_j 的影响之比在上述两个相邻等级之间
$1, 1/2, \dots, 1/9$	C_i 与 C_j 的影响之比为上面 a_{ij} 的互反数

图 2 比较尺度定义表

通过查阅文献以及咨询专业人士，我们得到 5 项身体素质（下肢爆发力 C1、上肢爆发力 C2、灵敏性 C3、冲刺能力 C4、有氧耐性 C5）对于 4 种球场位置（前锋、中场、后卫、守门员）的重要性优先顺序。以前锋为例，根据重要性两两比较，得到其成对比较矩阵：

$$A = \begin{pmatrix} 1 & 4 & 2 & \frac{1}{2} & 3 \\ \frac{1}{4} & 1 & \frac{1}{3} & \frac{1}{5} & \frac{1}{2} \\ \frac{1}{2} & 3 & 1 & \frac{1}{3} & 2 \\ 2 & 5 & 3 & 1 & 4 \\ \frac{1}{3} & 2 & \frac{1}{2} & \frac{1}{4} & 1 \end{pmatrix}$$

这是一个五阶正互反矩阵，用 Matlab 求出 A 的最大特征根为 λ_{max} 为 5.0681，对应的归一化特征向量为：

$$w = (0.2625, 0.0618, 0.1599, 0.4185, 0.0973)$$

一致性指标：

$$CI = \frac{\lambda_{max} - n}{n - 1} = \frac{5.0681 - 5}{5 - 1} = 0.0170$$

项目	箭头跑 (s)	30m 跑 (s)	引体向上 (次)	纵跳 (cm)	立定跳远 (cm)	YOYO IR2 (m)
权重	12.67%	33.15%	4.89%	20.79%	20.79%	7.71%

表 11 前锋位置权重表

项目	箭头跑 (s)	30m 跑 (s)	引体向上 (次)	纵跳 (cm)	立定跳远 (cm)	YOYO IR2 (m)
权重	23.92%	14.57%	5.63%	8.87%	8.87%	38.14%

表 12 中场位置权重表

查表得，随机性一致性指标

$$RI = 1.12$$

一致性比率：

$$CR = \frac{CI}{RI} = 0.0152$$

由于 $CR = 0.0152 < 1$ ，通过了一致性检验，所以 w 可以作为准则层对目标层的权重向量。

5.3.3 建立具体测试项目的权重表

根据测试项目与各项身体机能间的对应关系，可将权重进行转换，注意由于下肢爆发力同时对应立定跳远和纵跳，所以计算项目权重时，需要将下肢爆发力算作两次

例如，计算立定跳远的权重公式如下：

$$w = \frac{\text{下肢爆发力权重}}{1 + \text{下肢爆发力权重}} = \frac{0.2625}{1 + 0.2625} = 0.2079$$

最后可得前锋位置各项测试的权重表如下：

5.3.4 建立各位置球员的项目权重表

根据 3 中方法进行计算得出：

中场：

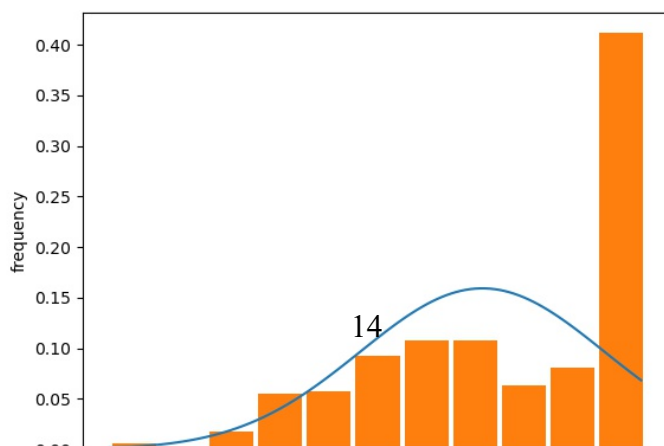
后卫：

守门员：

5.4 问题四

5.4.1 对现有体能测试方法的评价

以箭头跑项目得分分布为例子



项目	箭头跑 (s)	30m 跑 (s)	引体向上 (次)	纵跳 (cm)	立定跳远 (cm)	YOYO IR2 (m)
权重	11.27%	6.86%	18.51%	29.50%	29.50%	4.36%

表 14 守门员位置权重表

根据此得分分布图可以看出，原测试标准下，学生的得分分布大多分布在满分，不符合实际需求，因此需要对原有标准进行一定的改善。

统计规律表明，正态分布函数有着极其广泛的应用背景，生产与科学实验中很多随机变量的概率分布都可以近似地用正态分布来描述。故我们通过对参赛球员的成绩进行正态性检验来分析原有测试方法的科学性及合理性。

5.4.2 离散优化

我们希望对于每个项目，得分的分布符合正态分布

对于立定跳远，纵跳，引体向上和 YOYO 四个项目，得分情况如下：

$$score(x) = \begin{cases} 0 & x < standard[9] \\ i & standard[i] < x \leq standard[i+1] \\ 10 & x > standard[0] \end{cases}$$

而对于箭头跑和 30 米跑两个项目，得分情况如下：

$$score(x) = \begin{cases} 0 & x > standard[9] \\ i & standard[i+1] < x \leq standard[i] \\ 10 & x < standard[0] \end{cases}$$

$standard[i]$ 表示分数为 $10 - i$ 的标准线

而每个得分的分布情况，则是通过将单个项目的得分数据按照分数划分为 10 个值，然后统计这组数据在每个区间中出现的次数，记为 $real(x)$ 。

因此得分情况是离散分布的，该问题属于离散优化问题，即通过不断地改变测试标准，使得离散的得分分布直方图能够更加接近正态分布曲线

5.4.3 启发式算法中的爬山算法

解决离散优化模型问题的一个常用算法为启发式算法。

启发式算法可定义为：一个基于直观或经验构造的算法，在可接受的时间和空间复杂度下，给出待解决离散优化问题每一个实例的一个可行解。

我们使用了启发式算法中的爬山算法，将搜索过程比作爬山过程，沿着高度增加的方向爬。

我们选择该算法有以下两点理由：

- 启发式算法常用于解决现实世界中的一些问题，因为对于现实世界中可能出现的数据组合，不会使得启发式算法得到很坏的答案。
- 该问题在爬山问题中属于单峰问题，即使得损失函数 (每一个分值的频次与其正态分布情况下的频次差值) 最小，因此不会出现局部极值的问题。

损失函数形式如下：

$$loss_function(x) = norm(x) * totalNumber - real(x)$$

其中，正态分布函数：

$$norm(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

5.4.4 算法实现

对于不同的项目，设定不同的学习率 lr 和学习轮次 $epochs$ ：

- 学习率对应了每个项目测试标准数据的最小分度值。
- 为防止因浮点数运算的误差而导致测试标准不停上下浮动，永远无法到达爬山问题的封顶，因此为每一个项目的搜索设置了固定的轮次。

爬山算法伪代码如下 (以 30m 跑项目为例)：

Algorithm 1 爬山算法

Require: *originalStandard* 原始测试标准数组, *scoreData* 得分分布

Ensure: *currentStandard* 改进后的测试标准数组

```
1: function CLIMBALGORITHM(originalStandard, scoreData)
2:   currentStandard = originalStandard
3:   while epoch in epochs do originalStandard = currentStandard
4:     while i in [0 : 10] do
5:       if  $norm(i) * totalNumber - real(i) > 0$  then
6:         currentStandard[i] − = lr
7:       else if  $norm(i) * totalNumber - real(i) < 0$  then
8:         currentStandard[i] + = lr
9:       end if
10:    end while
11:    if originalStandard == currentStandard then
12:      Break;
13:    end if
14:  end while
15:  return result
16: end function
```

另外，考虑到解决实际问题时可能出现的一些 bug，对上述算法做了一些改进：

- 例如 30m 跑中，若 5 分的标准为 4.53 秒，而 4 分的标准为 4.54 秒，这样肯定是不合理的。
因此，对于每个项目，相邻两个分数之间需要有一定的区间，我们以原始测试标准中的最小的区间作为其改进后的最小区间。
- 例如，30m 跑中得到 5 分的人数频率小于其正态分布概率，则需要降低五分的标准，但如果降低五分的标准后，导致 4-5 分的区间过短，则需要将 4 分的标准同时降低，以此类推。

评分	30m (秒)	箭头跑 (秒)	立定跳远 (米)	纵跳 (厘米)	引体向上	YOYO IR2(米)
10	3.86	7.39	2.87	72.8	19	1160
9	3.99	7.59	2.76	65.1	18	1080
8	4.13	7.85	2.64	56.1	14	940
7	4.34	8.11	2.58	51.5	13	841
6	4.44	8.33	2.5	48.1	10	800
5	4.49	8.57	2.44	44.2	7	700
4	4.54	8.74	2.37	43.1	4	600
3	4.6	8.79	2.32	42.0	3	538
2	4.77	8.84	2.26	40.5	2	497
1	4.87	8.89	2.2	39.4	1	440

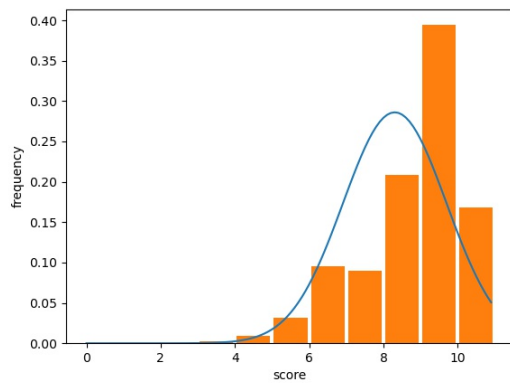
表 15 改进后的评分标准

5.4.5 结果展示

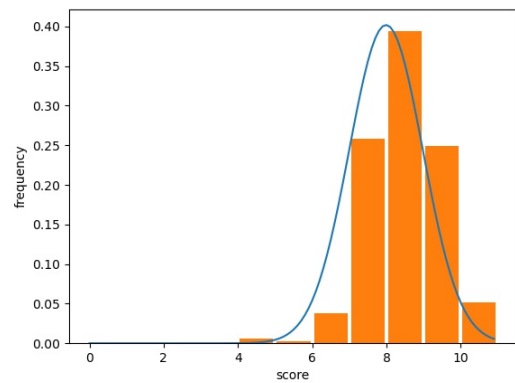
改进后的评分标准：

以下展示了各个项目，在原始测试数据，和经过爬山算法改进后的测试数据下，直方图与正态分布曲线的比较

- 30m 跑:



原始测试标准



改进后测试标准

图 4 30m 项目对比图

- 箭头跑:

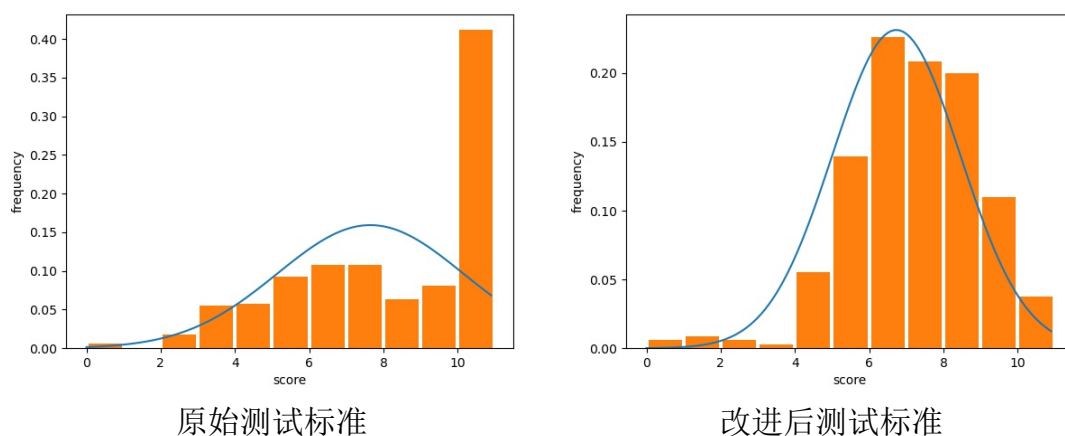


图 5 箭头跑项目对比图

• 立定跳远:

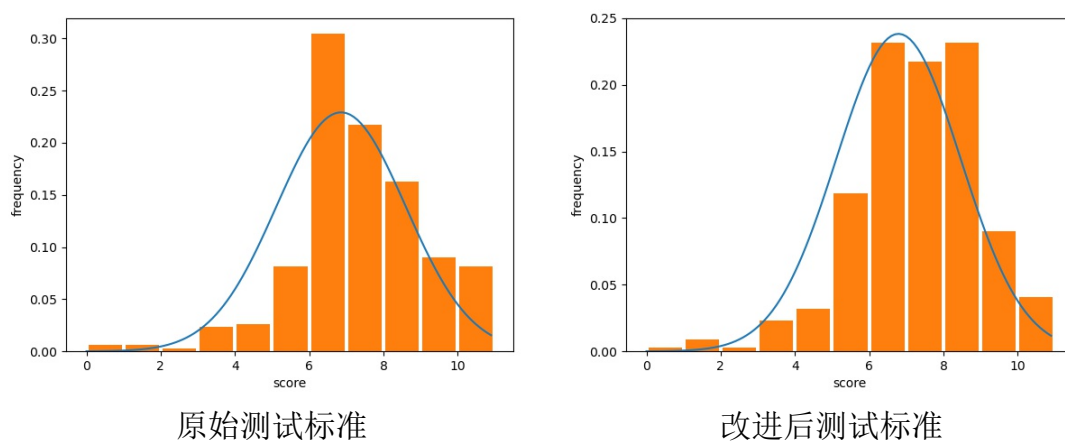


图 6 立定跳远项目对比图

• 纵跳:

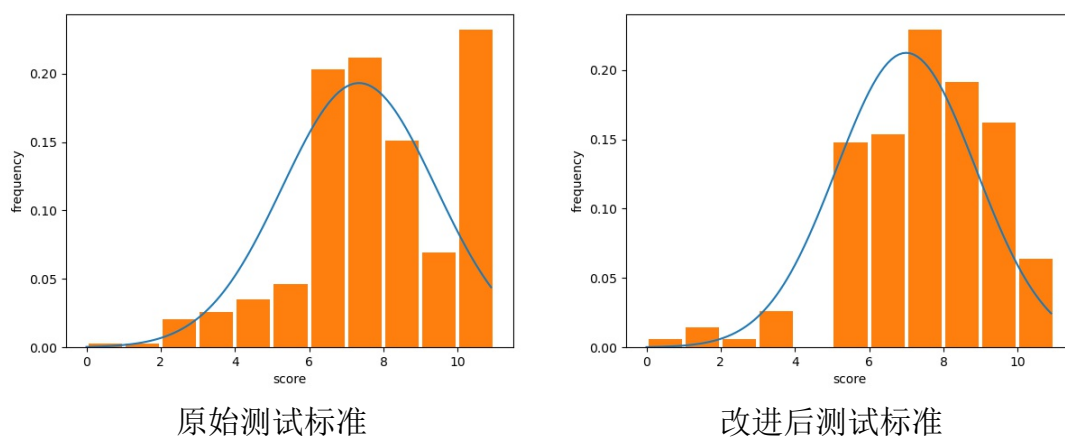


图 7 纵跳项目对比图

• 引体向上:

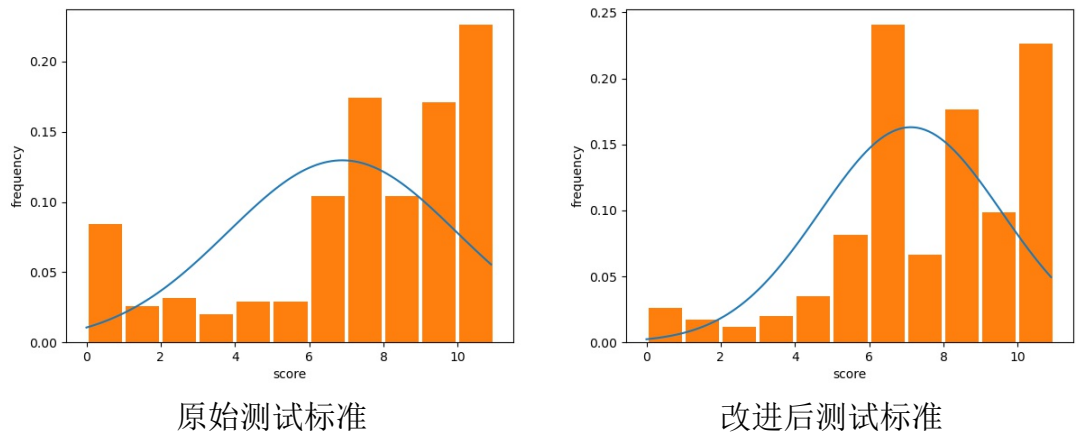


图 8 引体向上项目对比图

• YOYO:

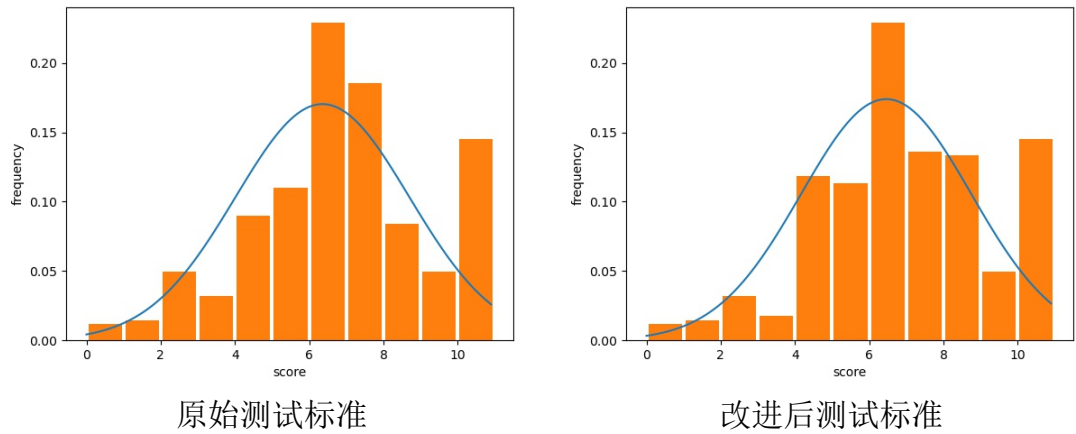


图 9 YOYO 项目对比图

上图可以看出，依据改进后的测试标准计算出来的数据，其频率直方图与正态分布曲线的相似度得到了极大的提高，绝大多数测试项目都能符合正态分布检验，说明优化后的新的测试方法更为科学合理。

5.4.6 基于不同年龄段球员的应用

通常，足球运动员随着年龄的增长身体素质也会发生不同程度上的改变。针对这一现象，最科学合理的解决办法就是分别建立各年龄段运动员专用的体能测试数据标准。

利用上述模型，首先分别收集各年龄段层次运动员的体能测试数据；其次代入模型进行数据分析与计算，分别得到各年龄段专用的测试评分标准；最终，整合对比各年龄段运动员的体能测试数据及评分标准，作为未来建立运动员线性成长追踪模型的参考依据。

6. 模型的评价

6.1 模型的优点

1. 本文的模型可以根据需求不同, 采用不同的方法对数据进行预处理, 使计算结果误差更小, 准确度更高。
2. 在使用层次分析法的过程中, 各矩阵的一致性均较好, 确保计算结果足够精确, 说明该评价模型的准确度高。
3. 优化模型采用启发模型可以在提高计算效率的同时, 得到更为理想的分析结果。
4. 该模型的系统性强。从数据的预处理到权重表的确定再到给出合理的测试方法, 整个过程紧密相连, 环环相扣。
5. 将不同年龄阶段球员的身体素质数据代入到模型中, 即快速可得出相关年龄段的测试数据标准, 具有良好的普遍适用性。

6.2 模型的缺点

1. 本文在确定及格线的模型建立中, 是以“满足合格人数在 22 人及以上”作为最终目标和判断依据, 在实际推广应用中具有一定的局限性。
2. 层次分析法的运用使得整个评价模型具有一定的主观性, 缺乏客观真实性。
3. 对于“引体向上”和“YOYO”这两个项目评分标准的确定上, 模型计算出的结果误差较大, 推测可能是由测试项目本身特点导致的, 以引体向上为例, 其测试标准分度值为 1, 因此对测试标准的调整难以趋近正态分布。

7. 参考文献

- [1] 夏雷震. 问卷调查中量表缺失值填补方法的模拟比较研究 [D]. 中国医科大学, 2018.
- [2] 段振华, 胡沛永, 代流通. 基于层次分析法的高职男子足球前锋运动员评价模型 [J]. 数学理论与应用, 2018, 38(Z2): 114-122.
- [3] 姜启源, 谢金星, 叶俊, 数学模型, 北京: 高等教育出版社, 2003.8, P224.
- [4] 李静, 张庆建, 张林虎. 足球运动员位置体能结构模型构建的理论探讨 [J]. 成都体育学院学报, 2011, 37(03): 42-45+57.
- [5] 朱军凯. 中国国家队男子足球运动员位置体能特征研究 [J]. 中国体育科技, 2012, 48(01): 68-75.
- [6] 百度百科, 正态分布, <https://baike.baidu.com/item/正态分布>, 2021 年 5 月 1 日;
- [7] 百度百科, 启发式算法, <https://baike.baidu.com/item/启发式算法/938987?fr=aladdin>, 2021 年 5 月 1 日;
- [8] 赵宁波. 青少年手球运动员选材标准修订与评分表的制定 [D]. 首都体育学院, 2015.

附录 A 表格

1.1 问题一附录

1.1.1 各球员的各项及综合体能得分

下列各表展示了经处理缺失及异常数据后的各个球员的各项及综合体能得分，其中异常数据用红颜色标出，使用热卡填充法填充的缺失数据用黄颜色标出

球队 1 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
1	1	8	7	9	6	7	5	6.45
1	2	9	8	8	6	7	6	7.05
1	3	7	8	8	5	10	7	7.35
1	4	8	10	6	4	0	6	6.10
1	5	9	10	8	7	2	5	6.55
1	6	9	10	9	2	6	4	6.15
1	7	9	10	7	7	9	7	7.95
1	8	9	10	10	6	7	4	6.75
1	9	6	10	4	3	2	6	5.70
1	10	8	10	8	6	7	4	6.40
1	11	9	10	8	7	7	5	7.05
1	12	9	10	5	3	7	5	6.35
1	13	6	9	8	4	10	0	4.45
1	14	7	10	5	4	7	6	6.55
1	15	10	10	7	2	7	6	7.00
1	16	10	10	9	6	10	7	8.30
1	17	9	10	9	7	8	6	7.65
1	18	9	10	7	2	10	6	7.15
1	19	7	9	8	6	6	5	6.40
1	20	9	10	8	4	6	5	6.65
1	21	9	10	7	7	3	4	6.15
1	22	9	10	8	3	7	2	5.45
1	23	8	9	6	3	4	6	6.25
1	24	9	10	7	7	9	4	6.75

球队 2 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
2	1	6	10	8	7	8	6	7.10
2	2	9	10	8	4	6	6	7.05
2	3	8	10	8	8	8	4	6.70
2	4	5	10	7	5	9	3	5.55
2	5	9	10	8	8	10	7	8.25
2	6	8	10	7	6	8	4	6.40
2	7	9	10	8	7	10	6	7.75
2	8	9	10	7	7	9	6	7.55
2	9	6	9	7	6	9	6	6.85
2	10	9	10	8	7	10	6	7.75
2	11	9	10	7	6	7	6	7.25
2	12	10	10	10	10	10	6	8.40
2	13	6	9	6	3	9	6	6.45
2	14	6	10	6	5	10	7	7.30
2	15	7	10	7	8	8	7	7.65
2	16	9	10	8	6	9	7	7.95
2	17	9	10	8	7	9	6	7.65
2	18	10	10	7	7	7	3	6.30
2	19	9	10	10	7	7	6	7.65
2	20	8	10	6	6	6	7	7.30
2	21	9	10	7	6	6	5	6.75
2	22	9	10	10	7	7	6	7.65
2	23	8	10	8	6	10	7	7.90
2	24	7	10	7	6	9	5	6.75
2	25	8	10	6	6	7	6	7.00
2	26	7	10	9	7	9	6	7.45

球队 3 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
3	1	10	10	7	6	10	5	7.30
3	2	9	10	10	7	10	5	7.55
3	3	9	10	10	9	9	2	6.45
3	4	9	10	8	7	8	7	7.95
3	5	8	10	6	4	9	6	7.00
3	6	9	10	8	4	10	5	7.05
3	7	9	10	9	6	8	4	6.75
3	8	10	10	7	6	9	4	6.80
3	9	9	10	7	7	8	5	7.05
3	10	10	10	8	8	10	7	8.40
3	11	9	10	10	8	10	4	7.25
3	12	10	9	10	9	9	3	6.85
3	13	9	10	7	6	4	7	7.35
3	14	8	10	8	6	8	4	6.50
3	15	9	10	10	9	9	4	7.25
3	16	7	10	7	6	6	7	7.25
3	17	9	10	9	7	10	7	8.25
3	18	9	10	6	1	6	6	6.55
3	19	9	10	6	6	10	3	6.25
3	20	9	10	6	7	8	3	6.15
3	21	5	9	6	3	7	2	4.50
3	22	9	10	8	6	10	6	7.65

球队 4 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
4	1	9	10	9	8	9	7	8.25
4	2	8	10	6	3	10	5	6.60
4	3	6	10	7	3	6	4	5.60
4	4	6	10	6	4	7	2	4.90
4	5	8	10	6	6	9	5	6.80
4	6	8	10	7	6	7	6	7.10
4	7	8	10	6	7	7	5	6.70
4	8	9	10	8	6	9	6	7.55
4	9	9	9	7	6	9	6	7.30
4	10	8	10	6	5	7	4	6.10
4	11	8	10	6	6	8	4	6.30
4	12	9	10	7	8	9	6	7.65
4	13	10	10	10	8	10	3	7.00
4	14	10	10	7	7	9	6	7.70
4	15	7	10	7	5	9	7	7.45
4	16	8	10	9	8	7	2	5.90
4	17	7	10	7	6	8	4	6.25
4	18	9	10	8	8	9	6	7.75
4	19	10	10	9	8	9	8	8.80
4	20	7	7	10	6	9	1	5.00
4	21	8	10	9	5	9	1	5.40
4	22	9	10	10	7	5	2	5.85
4	23	9	10	8	7	8	6	7.55

球队 5 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
5	1	8	9	10	8	9	0	5.25
5	2	9	10	5	6	7	2	5.45
5	3	8	10	7	7	8	6	7.30
5	4	6	10	7	6	9	4	6.20
5	5	10	10	9	7	10	4	7.20
5	6	7	10	6	2	7	6	6.45
5	7	6	10	4	2	9	2	4.70
5	8	7	10	3	2	10	5	6.05
5	9	9	10	8	6	9	4	6.75
5	10	10	10	10	8	8	5	7.60
5	11	9	10	7	6	9	2	5.85
5	12	7	10	6	6	8	5	6.55
5	13	8	10	8	7	9	2	5.90
5	14	9	9	8	7	9	0	5.10
5	15	9	10	6	4	7	5	6.55
5	16	8	10	9	8	8	5	7.20
5	17	8	10	6	6	10	4	6.50
5	18	9	10	8	6	9	6	7.55
5	19	8	10	6	5	6	4	6.00
5	20	9	10	10	8	7	3	6.55
5	21	9	10	10	8	9	3	6.75
5	22	9	10	7	8	9	6	7.65
5	23	8	10	7	7	5	2	5.40
5	24	9	10	8	7	8	5	7.15
5	25	7	10	7	4	10	6	7.05
5	26	7	10	7	5	10	5	6.75
5	27	8	4	10	7	7	1	4.60
5	28	9	10	7	4	4	2	5.15

球队 6 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
6	1	4	3	7	9	0	1	3.05
6	2	6	4	6	9	0	3	4.20
6	3	6	6	8	10	7	4	5.90
6	4	4	3	0	7	1	5	3.85
6	5	5	3	1	7	1	7	4.90
6	6	8	7	6	10	3	5	6.15
6	7	0	0	0	0	0	0	0
6	8	8	4	6	10	0	5	5.40
6	9	5	0	4	8	0	4	3.55
6	10	6	4	5	10	0	7	5.80
6	11	5	3	4	10	1	5	4.70
6	12	8	6	5	10	0	6	6.00
6	13	5	3	4	10	0	6	5.00
6	14	6	2	6	10	2	7	5.80
6	15	6	5	5	9	0	2	3.85
6	16	6	5	6	10	0	4	4.85
6	17	5	5	6	10	0	8	6.30
6	18	6	3	3	10	0	2	3.45
6	19	9	5	6	10	4	3	5.30
6	20	6	4	6	10	0	4	4.70
6	21	3	3	1	7	0	1	2.10
6	22	5	6	5	9	0	6	5.45
6	23	4	4	5	9	0	2	3.40
6	24	6	2	6	10	0	5	4.80
6	25	8	6	6	10	0	5	5.70

球队 7 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
7	1	8	7	7	10	5	7	7.25
7	2	8	7	6	10	3	10	8.15
7	3	0	0	0	0	0	0	0
7	4	5	6	6	10	5	6	6.15
7	5	7	4	6	10	8	7	6.85
7	6	8	9	6	10	9	7	7.85
7	7	8	6	7	10	3	8	7.30
7	8	9	6	4	9	7	7	7.05
7	9	7	7	7	10	7	9	8.10
7	10	9	9	6	10	4	7	7.50
7	11	7	7	6	10	8	4	6.10
7	12	6	4	7	10	7	5	5.90
7	13	7	5	6	10	7	8	7.30
7	14	8	5	7	10	6	10	8.25
7	15	8	7	5	10	8	10	8.55
7	16	6	7	7	10	5	9	7.75
7	17	9	7	6	10	3	6	6.70
7	18	6	6	5	10	4	0	3.70
7	19	6	4	5	10	7	6	6.10
7	20	9	8	6	10	0	8	7.35
7	21	8	6	4	10	7	2	5.00

球队 8 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
8	1	6	5	6	10	9	5	6.15
8	2	8	6	6	10	10	10	8.70
8	3	9	10	7	10	9	8	8.65
8	4	9	7	6	10	10	7	7.80
8	5	9	5	6	10	9	7	7.40
8	6	6	9	6	10	10	9	8.45
8	7	8	7	6	10	10	7	7.65
8	8	9	8	7	10	6	7	7.65
8	9	9	10	6	10	7	7	7.95
8	10	9	10	6	10	9	7	8.15
8	11	9	9	6	10	9	10	9.20
8	12	9	8	6	10	10	10	9.15
8	13	7	9	6	10	10	9	8.60
8	14	9	9	6	10	6	7	7.70
8	15	9	7	6	10	9	10	8.90
8	16	8	9	6	10	8	7	7.75
8	17	9	7	6	10	10	10	9.00
8	18	8	9	7	10	10	8	8.45
8	19	8	8	8	10	9	7	7.90
8	20	9	6	6	10	10	9	8.45
8	21	9	10	6	10	10	7	8.25
8	22	9	8	7	10	10	4	6.85
8	23	9	8	9	10	10	4	7.05
8	24	9	7	4	10	8	6	7.00
8	25	7	6	7	10	4	6	6.45
8	26	8	10	6	10	10	7	8.10
8	27	9	10	10	10	10	10	9.85

球队 9 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
9	1	6	5	7	9	2	4	5.05
9	2	8	6	7	10	0	6	6.20
9	3	8	5	6	10	5	7	6.85
9	4	6	6	6	10	5	10	7.90
9	5	7	6	5	9	2	7	6.35
9	6	10	10	9	10	5	5	7.40
9	7	9	7	3	9	4	9	7.60
9	8	9	6	6	10	3	6	6.55
9	9	5	5	5	9	7	8	6.80
9	10	6	4	7	10	8	5	6.00
9	11	9	7	8	10	9	8	8.30
9	12	8	6	7	10	0	9	7.40
9	13	10	6	9	10	2	7	7.30
9	14	6	6	5	9	0	7	6.00
9	15	7	6	6	10	6	6	6.55
9	16	9	6	6	10	10	10	8.85
9	17	8	7	6	9	1	10	7.85
9	18	9	6	6	10	7	9	8.15
9	19	7	6	3	8	0	5	5.05
9	20	8	7	6	10	5	7	7.15
9	21	7	5	3	9	0	9	6.60
9	22	7	6	6	10	6	7	6.95

球队 11 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
11	1	9	3	7	7	7	6	6.30
11	2	9	4	7	7	10	7	7.15
11	3	10	6	7	7	8	9	8.20
11	4	9	3	7	8	6	7	6.70
11	5	8	3	6	6	7	8	6.75
11	6	8	7	7	6	8	7	7.15
11	7	9	6	5	6	10	9	7.95
11	8	10	4	6	7	8	10	8.20
11	9	9	0	6	7	10	7	6.45
11	10	8	4	6	7	7	10	7.80
11	11	9	3	9	8	9	8	7.60
11	12	9	6	6	8	6	10	8.25
11	13	8	3	9	7	9	8	7.35
11	14	6	2	7	6	9	10	7.40
11	15	9	5	6	6	10	10	8.30
11	16	8	5	7	7	10	10	8.35
11	17	9	5	8	8	8	8	7.70
11	18	10	3	6	6	10	10	8.15
11	19	10	5	7	6	6	10	8.15
11	20	6	4	8	7	7	6	6.10
11	21	9	3	8	9	7	9	7.80
11	22	9	6	6	6	10	10	8.45
11	23	6	4	6	8	4	10	7.30
11	24	8	2	5	8	10	10	7.80
11	25	8	6	5	6	9	10	8.10
11	26	9	10	6	7	9	10	9.05
11	27	10	10	7	8	4	10	8.90
11	28	10	10	7	9	10	7	8.40
11	29	9	10	5	6	9	10	8.85
11	30	9	10	7	8	10	8	8.55
11	31	9	10	9	8	6	8	8.35

球队 12 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
12	1	9	5	5	5	6	9	7.30
12	2	9	3	9	7	10	3	5.60
12	3	9	5	6	4	10	8	7.30
12	4	10	10	7	7	10	5	7.40
12	5	8	4	5	7	7	2	4.50
12	6	10	6	9	7	10	5	7.00
12	7	8	6	7	8	10	8	7.80
12	8	9	4	3	8	10	8	7.25
12	9	9	7	5	6	10	8	7.70
12	10	8	7	9	8	10	10	8.95
12	11	7	2	8	6	7	5	5.45
12	12	9	3	6	7	10	7	6.90
12	13	10	9	5	6	10	6	7.35
12	14	6	4	4	6	10	7	6.30
12	15	8	5	7	6	9	6	6.55
12	16	9	10	6	6	7	6	7.15
12	17	7	8	2	5	10	4	5.55
12	18	7	5	5	6	9	8	7.00
12	19	9	8	9	7	0	8	7.35
12	20	8	9	5	7	7	10	8.45
12	21	9	8	9	8	10	7	8.05
12	22	9	5	8	8	10	10	8.70
12	23	9	2	6	2	7	6	5.55

球队 13 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
13	1	10	10	7	8	8	10	9.30
13	2	10	10	6	7	10	10	9.30
13	3	9	9	8	10	10	10	9.50
13	4	10	7	7	8	10	7	7.85
13	5	10	10	9	8	6	6	7.70
13	6	10	9	10	9	10	9	9.35
13	7	8	9	3	7	7	10	8.25
13	8	9	9	9	9	7	10	9.20
13	9	10	5	6	9	9	10	8.65
13	10	10	7	9	9	7	6	7.45
13	11	10	5	8	7	10	9	8.35
13	12	9	8	8	6	9	8	8.05
13	13	9	5	10	9	9	10	8.90
13	14	8	5	5	7	9	8	7.25
13	15	9	10	7	8	10	9	8.95
13	16	10	6	6	6	6	7	7.00
13	17	9	9	5	5	8	6	6.90
13	18	9	7	6	7	8	6	6.90
13	19	10	8	8	9	0	8	7.60
13	20	10	9	7	8	10	10	9.35
13	21	9	6	6	8	8	10	8.45
13	22	10	10	6	6	7	10	8.90
13	23	10	9	8	7	7	10	9.05
13	24	10	10	8	8	6	10	9.20

球队 14 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
14	1	8	7	8	7	0	6	6.15
14	2	9	7	6	8	1	6	6.30
14	3	9	8	3	7	6	7	6.95
14	4	10	5	5	5	5	7	6.55
14	5	10	5	10	10	6	6	7.25
14	6	10	7	8	7	1	6	6.55
14	7	10	3	6	6	7	6	6.25
14	8	10	8	10	10	6	6	7.70
14	9	9	6	8	7	0	6	6.15
14	10	8	4	6	8	2	7	6.20
14	11	7	3	6	7	1	6	5.30
14	12	8	5	6	6	6	7	6.55
14	13	9	7	6	6	6	6	6.60
14	14	10	6	8	10	7	6	7.30
14	15	10	5	6	8	7	6	6.75
14	16	9	7	10	8	6	6	7.20
14	17	8	5	7	8	6	6	6.45
14	18	10	7	6	7	2	9	7.65
14	19	10	8	9	6	6	6	7.20
14	20	9	10	8	7	8	7	7.95
14	21	8	7	7	7	3	6	6.35
14	22	9	10	10	7	2	6	7.15

球队 15 各球员的各项及综合体能得分

球队编号	球员编号	30M 跑成绩	箭头跑成绩	立定跳远成绩	纵跳成绩	引体向上成绩	YOYO 成绩	体能综合成绩
15	1	10	10	9	8	9	7	8.40
15	2	9	8	5	5	7	7	7.05
15	3	9	7	7	8	10	10	8.90
15	4	9	8	8	7	9	6	7.35
15	5	10	10	10	8	8	6	8.00
15	6	7	3	6	3	10	7	6.20
15	7	8	10	6	5	10	10	8.80
15	8	9	6	7	6	10	8	7.75
15	9	10	6	7	6	10	10	8.70
15	10	9	8	8	7	7	10	8.75
15	11	10	8	9	8	6	10	9.00
15	12	9	7	9	7	2	4	5.80
15	13	10	10	10	7	7	8	8.60
15	14	9	4	8	6	6	10	7.95
15	15	8	7	6	7	7	10	8.25
15	16	9	10	6	5	8	7	7.55
15	17	8	8	6	5	7	6	6.60
15	18	9	7	7	6	2	7	6.70
15	19	10	10	10	8	10	7	8.60
15	20	10	10	10	8	6	6	7.80
15	21	10	10	8	7	6	6	7.50
15	22	0	0	0	0	0	0	0
15	23	8	8	7	6	7	8	7.60
15	24	10	10	6	6	1	8	7.50
15	25	10	7	6	8	10	8	8.15
15	26	10	10	8	6	0	7	7.20
15	27	9	5	6	6	1	8	6.60
15	28	9	10	6	6	9	7	7.75
15	29	9	9	8	7	7	10	8.90
15	30	5	6	0	0	6	5	4.25

1.1.1.2 相关系数

11 队变量 C、E、F、G、H 与变量 D 的相关系数

$\rho(C^*, D^*)$	0.3247
$\rho(E^*, D^*)$	-0.0066
$\rho(F^*, D^*)$	0.1455
$\rho(G^*, D^*)$	-0.0849
$\rho(H^*, D^*)$	0.1333

12 队变量 C、D 与变量 E 的相关系数

$\rho(C^*, E^*)$	0.3080
$\rho(D^*, E^*)$	-0.0214

12 队变量 C、D 与变量 F 的相关系数

$\rho(C^*, F^*)$	0.0541
$\rho(D^*, F^*)$	0.2453

12 队变量 C、D 与变量 G 的相关系数

$\rho(C^*, G^*)$	0.0143
$\rho(D^*, G^*)$	-0.0696

12 队变量 C、D 与变量 H 的相关系数

$\rho(C^*, H^*)$	0.0334
$\rho(D^*, H^*)$	0.1792

1.1.3 所有球员的总成绩排序

下表给出了所有球员的总成绩排序表，用于在 5.2.2 中合格线的确定

球队编号	球员编号	体能综合成绩	
8	27	9.85	
13	3	9.50	
13	6	9.35	
13	20	9.35	
13	1	9.30	
13	2	9.30	
8	11	9.20	
13	8	9.20	
13	24	9.20	
8	12	9.15	
11	26	9.05	
13	23	9.05	
8	17	9.00	
15	11	9.00	
12	10	8.95	
13	15	8.95	
8	15	8.90	
11	27	8.90	
13	22	8.90	
15	3	8.90	
15	29	8.90	
13	13	8.90	
9	16	8.85	
11	29	8.85	
4	19	8.80	

15	7	8.80	
15	10	8.75	
8	2	8.70	
12	22	8.70	
15	9	8.70	
8	3	8.65	
13	9	8.65	
15	13	8.60	
8	13	8.60	
15	19	8.60	
7	15	8.55	
11	30	8.55	
8	6	8.45	
8	18	8.45	
8	20	8.45	
11	22	8.45	
12	20	8.45	
13	21	8.45	
2	12	8.40	
3	10	8.40	
11	28	8.40	
15	1	8.40	
11	31	8.35	
11	16	8.35	
13	11	8.35	
1	16	8.30	
9	11	8.30	
11	15	8.30	
2	5	8.25	
3	17	8.25	
4	1	8.25	
7	14	8.25	
8	21	8.25	
11	12	8.25	
13	7	8.25	
15	15	8.25	
11	3	8.20	
11	8	8.20	
7	2	8.15	
8	10	8.15	
9	18	8.15	
11	18	8.15	
11	19	8.15	
15	25	8.15	
8	26	8.10	

7	9	8.10	
11	25	8.10	
12	21	8.05	
13	12	8.05	
15	5	8.00	
1	7	7.95	
2	16	7.95	
15	14	7.95	
3	4	7.95	
8	9	7.95	
11	7	7.95	
14	20	7.95	
2	23	7.90	
8	19	7.90	
9	4	7.90	
7	6	7.85	
9	17	7.85	
13	4	7.85	
8	4	7.80	
11	10	7.80	
15	20	7.80	
11	24	7.80	
12	7	7.80	
11	21	7.80	
15	8	7.75	
2	7	7.75	
2	10	7.75	
4	18	7.75	
7	16	7.75	
8	16	7.75	
15	28	7.75	
4	14	7.70	
8	14	7.70	
13	5	7.70	
14	8	7.70	
12	9	7.70	
11	17	7.70	
2	15	7.65	
2	17	7.65	
2	19	7.65	
2	22	7.65	
3	22	7.65	
4	12	7.65	
5	22	7.65	
8	7	7.65	

8	8	7.65	
14	18	7.65	
1	17	7.65	
11	11	7.60	
13	19	7.60	
15	23	7.60	
5	10	7.60	
9	7	7.60	
2	8	7.55	
4	8	7.55	
5	18	7.55	
15	16	7.55	
3	2	7.55	
4	23	7.55	
7	10	7.50	
15	21	7.50	
15	24	7.50	
4	15	7.45	
2	26	7.45	
13	10	7.45	
8	5	7.40	
9	6	7.40	
9	12	7.40	
11	14	7.40	
12	4	7.40	
3	13	7.35	
7	20	7.35	
11	13	7.35	
15	4	7.35	
1	3	7.35	
12	13	7.35	
12	19	7.35	
2	14	7.30	
2	20	7.30	
3	1	7.30	
4	9	7.30	
5	3	7.30	
7	13	7.30	
9	13	7.30	
11	23	7.30	
14	14	7.30	
7	7	7.30	
12	1	7.30	
12	3	7.30	
2	11	7.25	

13	14	7.25	
3	11	7.25	
3	15	7.25	
3	16	7.25	
7	1	7.25	
12	8	7.25	
14	5	7.25	
14	16	7.20	
14	19	7.20	
15	26	7.20	
5	16	7.20	
5	5	7.20	
1	18	7.15	
9	20	7.15	
11	2	7.15	
11	6	7.15	
12	16	7.15	
14	22	7.15	
5	24	7.15	
2	1	7.10	
4	6	7.10	
2	2	7.05	
5	25	7.05	
7	8	7.05	
15	2	7.05	
1	2	7.05	
1	11	7.05	
3	6	7.05	
3	9	7.05	
8	23	7.05	
1	15	7.00	
2	25	7.00	
3	5	7.00	
4	13	7.00	
8	24	7.00	
12	6	7.00	
12	18	7.00	
13	16	7.00	
9	22	6.95	
14	3	6.95	
12	12	6.90	
13	17	6.90	
13	18	6.90	
2	9	6.85	
3	12	6.85	

7	5	6.85	
8	22	6.85	
9	3	6.85	
3	8	6.80	
4	5	6.80	
9	9	6.80	
14	15	6.75	
1	8	6.75	
1	24	6.75	
2	21	6.75	
2	24	6.75	
3	7	6.75	
5	9	6.75	
5	21	6.75	
5	26	6.75	
11	5	6.75	
15	18	6.70	
4	7	6.70	
7	17	6.70	
11	4	6.70	
2	3	6.70	
1	20	6.65	
4	2	6.60	
14	13	6.60	
15	17	6.60	
9	21	6.60	
15	27	6.60	
3	18	6.55	
9	8	6.55	
9	15	6.55	
12	15	6.55	
14	4	6.55	
14	12	6.55	
1	5	6.55	
1	14	6.55	
5	12	6.55	
5	15	6.55	
5	20	6.55	
14	6	6.55	
3	14	6.50	
5	17	6.50	
2	13	6.45	
14	17	6.45	
1	1	6.45	
3	3	6.45	

5	6	6.45	
8	25	6.45	
11	9	6.45	
1	10	6.40	
1	19	6.40	
2	6	6.40	
9	5	6.35	
14	21	6.35	
1	12	6.35	
2	18	6.30	
4	11	6.30	
6	17	6.30	
11	1	6.30	
12	14	6.30	
14	2	6.30	
1	23	6.25	
3	19	6.25	
4	17	6.25	
14	7	6.25	
5	4	6.20	
14	10	6.20	
15	6	6.20	
9	2	6.20	
1	6	6.15	
1	21	6.15	
6	6	6.15	
7	4	6.15	
8	1	6.15	
14	1	6.15	
14	9	6.15	
3	20	6.15	
1	4	6.10	
7	19	6.10	
11	20	6.10	
4	10	6.10	
7	11	6.10	
5	8	6.05	
5	19	6.00	
6	12	6.00	
9	10	6.00	
9	14	6.00	
4	16	5.90	
5	13	5.90	
6	3	5.90	
7	12	5.90	85% 分数线

5	11	5.85	
4	22	5.85	
6	10	5.80	
6	14	5.80	
15	12	5.80	
1	9	5.70	
6	25	5.70	
12	2	5.60	
4	3	5.60	
2	4	5.55	
12	17	5.55	
12	23	5.55	
12	11	5.45	
6	22	5.45	
1	22	5.45	
5	2	5.45	
4	21	5.40	
5	23	5.40	
6	8	5.40	
14	11	5.30	
6	19	5.30	
5	1	5.25	
5	28	5.15	
5	14	5.10	
9	1	5.05	
9	19	5.05	
4	20	5.00	
6	13	5.00	
7	21	5.00	
6	5	4.90	
4	4	4.90	
6	16	4.85	
6	24	4.80	
5	7	4.70	
6	11	4.70	
6	20	4.70	
5	27	4.60	
3	21	4.50	
12	5	4.50	
1	13	4.45	
15	30	4.25	
6	2	4.20	
6	4	3.85	
6	15	3.85	
7	18	3.70	

6	9	3.55	
6	18	3.45	
6	23	3.40	
6	1	3.05	
6	21	2.10	
6	7	0.00	
7	3	0.00	
15	22	0.00	

1.1.4 各队不合格成员名单

下表给出了 5.1.3 中各队不合格成员名单及成绩，其中第 8、11、13 队无不合格成员。

球队编号	球员编号	体能综合得分
1	9	5.70
1	13	4.45
1	22	5.45
2	4	5.55
3	21	4.50
4	3	5.60
4	4	4.90
4	16	5.90
4	20	5.00
4	21	5.40
4	22	5.85
5	1	5.25
5	2	5.45
5	7	4.70
5	11	5.85
5	13	5.90
5	14	5.10
5	23	5.40
5	27	4.60
5	28	5.15
6	1	3.05
6	2	4.20
6	3	5.90
6	4	3.85
6	5	4.90
6	7	0.00
6	8	5.40
6	9	3.55
6	10	5.80
6	11	4.70
6	13	5.00
6	14	5.80
6	15	3.85
6	16	4.85
6	18	3.45
6	19	5.30
6	20	4.70
6	21	2.10
6	22	5.45
6	23	3.40
6	24	4.80
6	25	5.70
7	3	0.00

续上表

球队编号	球员编号	体能综合得分
7	12	5.90
7	18	3.70
7	21	5.00
9	1	5.05
9	19	5.05
12	2	5.60
12	5	4.50
12	11	5.45
12	17	5.55
12	23	5.55
14	11	5.30
15	12	5.80
15	22	0.00
15	30	4.25

1.2 问题二附录

1.2.1 候选及格线的生成

下表给出了在合格线以下的球员成绩以及候选及格线的分析，用于 5.2.2 中候选及格线的选取以及各候选及格线的有效贡献率。

球队编号	球员编号	体能综合成绩	及格线选取可能值	选取该及格线的人员有效贡献率
5	11	5.85		
4	22	5.85		
6	10	5.80		
6	14	5.80		
15	12	5.80		
1	9	5.70	1 队满 22 人	0.17
6	25	5.70		
12	2	5.60		
4	3	5.60		
2	4	5.55		
12	17	5.55		
12	23	5.55		
12	11	5.45	12 队满 22 人	0.38
6	22	5.45		
1	22	5.45		
5	2	5.45	5 队满 22 人	0.44
4	21	5.40		
5	23	5.40		
6	8	5.40		
14	11	5.30	14 队满 22 人	0.40
6	19	5.30		
5	1	5.25		
5	28	5.15		
5	14	5.10		
9	1	5.05		
9	19	5.05	9 队满 22 人	0.38
4	20	5.00	4 队满 22 人	0.52
6	13	5.00		
7	21	5.00		
6	5	4.90		
4	4	4.90		
6	16	4.85		
6	24	4.80		

球队编号	球员编号	体能综合成绩	及格线选取可能值	选取该及格线的人员有效贡献率
5	7	4.70		
6	11	4.70		
6	20	4.70		
5	27	4.60		
3	21	4.50	3 队满 22 人	0.39
12	5	4.50		
1	13	4.45		
15	30	4.25		
6	2	4.20		
6	4	3.85		
6	15	3.85		
7	18	3.70		
6	9	3.55		
6	18	3.45		
6	23	3.40	6 队满 22 人	0.69
6	1	3.05		
6	21	2.10		
6	7	0.00		
7	3	0.00		
15	22	0.00		

附录 B 代码

2.1 问题一代码

```

import pandas as pd
import numpy as np

excel_path =
    'F:\\study\\数学建模\\2021\\tongjimcm2021D体能数据.xlsx'
standardArrow = [7.91, 8.01, 8.11, 8.21, 8.31, 8.41, 8.55, 8.73,
    8.89, 9.05]
standardThirty = [3.92, 4.07, 4.17, 4.22, 4.34, 4.42, 4.52,
    4.67, 4.77, 4.87]
standardJumping = [2.81, 2.74, 2.66, 2.59, 2.49, 2.43, 2.38,
    2.32, 2.26, 2.21]
standardUpJumping = [65, 60.1, 55.1, 51.1, 47.1, 45.6, 43.8, 42,
    40.2, 38.4]
standardUp = [19, 16, 14, 11, 10, 9, 8, 7, 6, 5]

```

```
standardYOYO = [1160, 1080, 1000, 880, 800, 720, 640, 600, 520, 440]
```

```
def computeScore(val, standard):  
    if val == 0:  
        return 0  
    if val <= standard[0]:  
        s = 10  
    elif standard[1] >= val > standard[0]:  
        s = 9  
    elif standard[2] >= val > standard[1]:  
        s = 8  
    elif standard[3] >= val > standard[2]:  
        s = 7  
    elif standard[4] >= val > standard[3]:  
        s = 6  
    elif standard[5] >= val > standard[4]:  
        s = 5  
    elif standard[6] >= val > standard[5]:  
        s = 4  
    elif standard[7] >= val > standard[6]:  
        s = 3  
    elif standard[8] >= val > standard[7]:  
        s = 2  
    elif standard[9] >= val > standard[8]:  
        s = 1  
    else:  
        s = 0  
    return s
```

对于如引体向上等项目，数据值越高，分数越高的计算方法

```
def computeScore2(val, standard):  
    if val == 0:  
        return 0  
    if val >= standard[0]:  
        s = 10  
    elif standard[1] <= val < standard[0]:  
        s = 9  
    elif standard[2] <= val < standard[1]:  
        s = 8  
    elif standard[3] <= val < standard[2]:  
        s = 7  
    elif standard[4] <= val < standard[3]:  
        s = 6  
    elif standard[5] <= val < standard[4]:  
        s = 5  
    elif standard[6] <= val < standard[5]:  
        s = 4  
    elif standard[7] <= val < standard[6]:  
        s = 3  
    elif standard[8] <= val < standard[7]:  
        s = 2
```

```

elif standard[9] <= val < standard[8]:
    s = 1
else:
    s = 0
return s

def writeTeamScore():
    teams = []
    for i in range(15):
        table = pd.read_excel(excel_path, sheet_name=str(i + 1))
        table = table.fillna(0)
        rows, cols = table.shape
        team = []
        for row in range(rows):
            team.append(
                (computeScore(table.iloc[row, 1], standardThirty),
                 computeScore(table.iloc[row, 2], standardArrow),
                 computeScore2(table.iloc[row, 3], standardJumping),
                 computeScore2(table.iloc[row, 4],
                                standardUpJumping), computeScore2(table.iloc[row, 5],
                                standardUp),
                 computeScore2(table.iloc[row, 6]), standardYOYO))
        teams.append(team)
    writer = pd.ExcelWriter('team.xlsx')
    for i in range(len(teams)):
        teamScore = np.array(teams[i])
        data = pd.DataFrame(teamScore)
        data.to_excel(writer, str(i + 1), float_format='%.5f',
                      header=False, index=False)
    writer.save()

if __name__ == '__main__':
    writeTeamScore()

```

2.2 问题三代码

```

disp('请输入判断矩阵A(n阶)');
A=input('A=');
[n,n]=size(A);
x=ones(n,100);
y=ones(n,100);
m=zeros(1,100);
m(1)=max(x(:,1));
y(:,1)=x(:,1);
x(:,2)=A*y(:,1);
m(2)=max(x(:,2));
y(:,2)=x(:,2)/m(2);
p=0.0001;i=2;k=abs(m(2)-m(1));
while k>p
    i=i+1;
    x(:,i)=A*y(:,i-1);

```

```

m(i)=max(x(:,i));
y(:,i)=x(:,i)/m(i);
k=abs(m(i)-m(i-1));
end
a=sum(y(:,i));
w=y(:,i)/a;
t=m(i);
disp(w);
    %以下是一致性检验
CI=(t-n)/(n-1);RI=[0 0 0.52 0.89 1.12 1.26 1.36 1.41 1.46 1.49
    1.52 1.54 1.56 1.58 1.59];
CR=CI/RI(n);
if CR<0.10
    disp('此矩阵的一致性可以接受!');
    disp('t=');disp(t);
    disp('CI=');disp(CI);
    disp('CR=');disp(CR);
end

```

2.3 问题四代码

```

import pandas as pd
import numpy as np # numpy是Python中科学计算的核心库
import matplotlib.pyplot as plt # matplotlib数据可视化神器
from scipy import stats

# 对于如跑步等项目，数据值越低，分数越高的计算方法
def computeScore(val, standard):
    if val == 0:
        return 0
    if val <= standard[0]:
        s = 10
    elif standard[1] >= val > standard[0]:
        s = 9
    elif standard[2] >= val > standard[1]:
        s = 8
    elif standard[3] >= val > standard[2]:
        s = 7
    elif standard[4] >= val > standard[3]:
        s = 6
    elif standard[5] >= val > standard[4]:
        s = 5
    elif standard[6] >= val > standard[5]:
        s = 4
    elif standard[7] >= val > standard[6]:
        s = 3
    elif standard[8] >= val > standard[7]:
        s = 2
    elif standard[9] >= val > standard[8]:
        s = 1
    else:
        s = 0

```

```

    return s

# 对于如引体向上等项目，数据值越高，分数越高的计算方法
def computeScore2(val, standard):
    if val == 0:
        return 0
    if val >= standard[0]:
        s = 10
    elif standard[1] <= val < standard[0]:
        s = 9
    elif standard[2] <= val < standard[1]:
        s = 8
    elif standard[3] <= val < standard[2]:
        s = 7
    elif standard[4] <= val < standard[3]:
        s = 6
    elif standard[5] <= val < standard[4]:
        s = 5
    elif standard[6] <= val < standard[5]:
        s = 4
    elif standard[7] <= val < standard[6]:
        s = 3
    elif standard[8] <= val < standard[7]:
        s = 2
    elif standard[9] <= val < standard[8]:
        s = 1
    else:
        s = 0
    return s

# 计算正态分布概率
def normFunction(x, mu, sigma):
    pdf = np.exp(-((x - mu) ** 2) / (2 * sigma ** 2)) / (sigma *
        np.sqrt(2 * np.pi))
    return pdf

# 读取excel表格，生成各项目对应的统计数据
def addProject(k):
    excel_path = 'F:/study/数学建模/2021/tongjimcm2021D体能数据 -
        副本.xlsx'
    data = []
    for i in range(14):
        table = pd.read_excel(excel_path, sheet_name=str(i + 1))
        rows, cols = table.shape
        for row in range(rows):
            data.append(table.iloc[row, k])
    return data

# 计算列表中相同值元素的个数
def countList(score, target):

```



```

countNum = 0
for i in range(len(score)):
    if score[i] == target:
        countNum += 1
return countNum

# 绘出并保存条状图和正态分布图
def draw(data, standard, string, flag):
    score = []
    if flag == "down":
        for i in range(len(data)):
            score.append(computeScore(data[i], standard))
    elif flag == "up":
        for i in range(len(data)):
            score.append(computeScore2(data[i], standard))
    mean = np.mean(score) # 获得数据集的平均值
    std = np.std(score) # 获得数据集的标准差
    x = np.arange(0, 11, 0.1)
    # 设定Y轴, 载入刚才定义的正态分布函数
    y = normFunction(x, mean, std)
    # 绘制数据集的正态分布曲线
    plt.plot(x, y)
    # 绘制数据集的直方图
    plt.hist(score, align="mid", bins=range(0, 12, 1),
              rwidth=0.9, density=True)
    plt.xlabel('score')
    plt.ylabel('frequency')
    plt.savefig(string + '.jpg')
    plt.show()
    return score

# 优化过程
def trainNetwork(data, standard, lr, epochs, string, flag, gap):
    originalString = string + "原始"
    draw(data, standard, originalString, flag)
    if flag == 'down':
        for epoch in range(epochs):
            score = []
            oldStandard = standard
            for i in range(len(data)):
                score.append(computeScore(data[i], standard))
            mean = np.mean(score) # 获得数据集的平均值
            std = np.std(score) # 获得数据集的标准差
            for i in range(len(standard)):
                if countList(score, 10 - i) - normFunction(10 - i,
                                                             mean, std) * 348 < 0.5 and countList(score, 10 -
                                                             i) > 1:
                    for k in range(i, len(standard)):
                        if k < len(standard) - 1 and standard[k] + lr
                           < standard[k + 1] - gap:
                            for j in range(i, k + 1):
                                standard[j] += lr

```

```

        break
    elif k == len(standard) - 1:
        standard[k] += lr
    elif countList(score, 10 - i) - normFunction(10 - i,
        mean, std) * 348 > 0.5 and countList(score, 10 -
        i) >= 1:
        for k in range(i, -1, -1):
            if k > 0 and standard[k - 1] + lr <
                standard[k] - gap:
                for j in range(i, k - 1, -1):
                    standard[j] -= lr
                break
            elif k == 0:
                standard[k] -= lr
elif flag == 'up':
    for epoch in range(epochs):
        score = []
        for i in range(len(data)):
            score.append(computeScore2(data[i], standard))
        mean = np.mean(score) # 获得数据集的平均值
        std = np.std(score) # 获得数据集的标准差
        for i in range(len(standard)):
            if countList(score, 10 - i) - normFunction(10 - i,
                mean, std) * 348 > 0.5 and countList(score, 10 -
                i) > 1:
                for k in range(i, -1, -1):
                    if k > 0 and standard[k] + lr < standard[k -
                        1] - gap:
                        for j in range(k, i + 1):
                            standard[j] += lr
                        break
                    elif k == 0:
                        standard[k] += lr
            elif countList(score, 10 - i) - normFunction(10 - i,
                mean, std) * 348 < 0.5 and countList(score, 10 -
                i) > 0.5:
                for k in range(i, len(standard)):
                    if standard[k] >= 1:
                        if k < len(standard) - 1 and standard[k +
                            1] + lr < standard[k] - gap:
                            for j in range(k, i - 1, -1):
                                standard[j] -= lr
                            break
                        elif k == len(standard) - 1:
                            standard[k] -= lr
        currentString = string + "优化后"
        draw(data, standard, currentString, flag)
        for i in range(len(standard)):
            standard[i] = round(standard[i], 2)
        print(string + "优化后的标准")
        print(standard)

if __name__ == '__main__':

```

```

standardArrow = [7.91, 8.01, 8.11, 8.21, 8.31, 8.41, 8.55,
8.73, 8.89, 9.05]
standardThirty = [3.92, 4.07, 4.17, 4.22, 4.34, 4.42, 4.52,
4.67, 4.77, 4.87]
standardJumping = [2.81, 2.74, 2.66, 2.59, 2.49, 2.43, 2.38,
2.32, 2.26, 2.21]
standardUpJumping = [65, 60.1, 55.1, 51.1, 47.1, 45.6, 43.8,
42, 40.2, 38.4]
standardUp = [19, 16, 14, 11, 10, 9, 8, 7, 6, 5]
standardYOYO = [1160, 1080, 1000, 880, 800, 720, 640, 600,
520, 440]
thirtyData = addProject(1)
arrowData = addProject(2)
jumpingData = addProject(3)
upJumpingData = addProject(4)
upData = addProject(5)
YOYOData = addProject(6)
trainNetwork(thirtyData, standardThirty, 0.01, 100, "30m跑",
"down", 0.05)
trainNetwork(arrowData, standardArrow, 0.01, 100, "箭头跑",
"down", 0.05)
trainNetwork(jumpingData, standardJumping, 0.01, 100,
"立定跳远", "up", 0.05)
trainNetwork(upJumpingData, standardUpJumping, 0.1, 100,
"纵跳", "up", 1)
trainNetwork(upData, standardUp, 1, 10, "引体向上", "up", 0.1)
trainNetwork(YOYOData, standardYOYO, 1, 100, "YOYO", "up", 40)

```