arXiv:2304.04700v1 [cs.DS] 10 Apr 2023

通过随机化实现次模最大化中的长期公平性

Shaojie Tang[1], Jing Yuan[2], 和 Twumasi Mensah-Boateng[2]

[1] 德克萨斯大学达拉斯分校纳文·金达尔管理学院
shaojie.tang@utdallas.edu[2] 北德克萨斯大学计算机科学与工程系
jing.yuan@unt.edu twumasimensah-boateng@my.unt.edu

摘要。次模函数优化在机器学习和数据分析中有着广泛的应用，包括数据摘要，其目标是从大型数据集中识别出简洁且多样的数据点集合。在处理可能包含种族或性别等敏感属性的数据项时，采用公平性感知算法十分重要，以防止可能导致不同群体代表性不平等的偏见。基于此，我们研究了在满足群体公平性约束的同时最大化单调次模函数的问题。不同于该领域以往的研究，我们允许采用随机化解，目标是计算一个关于可行集的分布，使得从每个群体中选取的项目数的期望值受到上限和下限阈值形式的约束，从而确保每个群体的代表性在长期内保持平衡。如果一个集合的大小不超过常数 b，则称其为可行集。我们的研究包括为该问题开发一系列近似算法。

1 引言

如果一个集合函数遵循收益递减原则，即向更大的集合中添加一个项目所带来的收益更小，则称其为子模函数。这一概念被应用于各种现实场景，例如特征选择 [10]，其目标是从大量潜在特征中选择最相关的特征，以用于机器学习模型；主动学习 [15]，其目标是为机器学习模型选择一组实例作为学习对象；基于代表点的聚类 [11]，其目标是选择一组代表点来表示一组数据点；社交网络中的影响力最大化 [22]，其目标是选择一组目标个体，以最大化信息或影响力在网络中的传播；以及推荐系统 [13] 和多样化数据摘要 [20]。子模优化的目标是在满足大小限制、拟阵约束或背包约束等约束条件的同时，选择一个能够优化子模函数的项目集合。

在实践中，物品或个体通常会根据性别、种族、年龄、宗教或其他因素进行分组。然而，如果监控不当，现有算法可能会产生偏见，导致最终选出的集合中过度或不足代表某些群体。为了解决这一问题，我们提出研究长期公平子模最大化问题。其目标是随机选择物品的一个子集，使子模函数达到最优，同时确保每个组中被选物品的期望数量落在预期范围内。这种方法确保最终选出的物品不仅经过优化，而且公平公正，能够长期公平地代表所有群体。

形式化地说，我们考虑一个由 V 个项目组成的集合，这些项目被划分为 m 个（不一定互不相交的）群体： V_1、V_2、$\cdots$、V_m，每个群体中的项目都具有相似属性（例如种族）。为确保公平，随机项目选择算法必须对所有满足 $[m] =\{1, 2, \cdots, m\}$ 的群体 $t \in [m]$ 满足以下标准：从群体 V_t 中选出的项目的期望数量必须处于 $[\alpha_t, \beta_t]$ 的范围内，其中 α_t 和 β_t 是任意参数，且可能因群体而异；此外，选出项目的数量必须始终不超过 b 的基数约束。简而言之，公平的随机化解决方案必须满足两个重要要求 [3]: (a) 受限支配，即每个群体所占的项目比例必须处于一定范围内；(b) 少数群体保护，即每个群体所占的项目比例不得低于一定限度。我们的公平性表示法已在学术界获得广泛认可，并被应用于多项研究，包括多获胜者投票系统 [8], 、公平推荐系统[14], 以及受拟阵约束的优化问题 [9]。事实上，这种表示法能够刻画其他公平性定义，例如统计均等 [12], 、80%规则 [4], 和比例代表制 [18]。

与大多数专注于寻找满足公平性限制的固定物品集的既有研究不同，我们的方法支持随机化解，因此在满足公平性限制方面更加灵活。以具有公平性意识的产品推荐为例，目标是向在线消费者推荐一组产品，同时确保每个卖家组（例如男性卖家和女性卖家）都有至少一件产品预计会被推荐。由于展示空间有限，假设我们只能向消费者展示一件产品。在这种情况下，任何确定性解都无法满足公平性要求，但很容易找到一个满足该要求的随机化解。例如，可以以相同的概率从每个组中推荐一件产品。

1.1 我们的贡献

– 我们首次研究了长期公平子模最大化问题。由于该问题包含指数数量的变量，因此构成了重大挑战。Consequently，使用传统的线性规划（LP）求解器很难对其进行求解。

– 我们开发了一种 $(1-1/e)^2$-近似算法，可近似满足公平性约束。具体而言，我们的算法确保从组 V_t 中选出的项目数量处于 $[\lfloor\alpha_t\rfloor, \lceil\beta_t\rceil]$范围内。值得注意的是，如果 α_t 和 β_t 均为整数，我们的解将严格满足原问题中的公平性约束。
– 需要注意的是，前述算法需要优化底层子模函数的连续近似，即多线性扩展 [6]。这是通过执行连续贪心算法实现的，而该算法在实践中的实现计算成本较高。我们的第二项贡献是提出一种快速贪心算法，其近似比有所降低，为 $(1-1/e)^2/2$。

– 我们提出一种 $(1-1/e)$-近似随机化算法。我们的方法采用椭球法，并为原问题的对偶线性规划引入近似分离预言机；该对偶线性规划具有多项式数量的变量和指数数量的约束。与确定性解不同，我们的随机化方法具有三项关键优势。首先，我们的解不依赖于群体互不重叠这一假设。其次，我们的方法严格满足所有公平约束。第三，我们实现了最优近似比 $1-1/e$。

1.2 其他相关工作

人们日益认识到公平、客观的决策系统的重要性，这推动了各领域对公平算法的研发兴趣不断增长，例如影响力最大化 [24] 和分类 [26]。公平算法也已应用于投票系统[8],，其目标是确保选举结果能够公平地代表选民的偏好。此外，老虎机学习领域 [17], 涉及基于不确定信息进行序贯决策，该领域对公平算法研发的兴趣也日益浓厚。最后，数据摘要领域 [7] 越来越重视公平算法的研发，旨在为数据提供平衡的代表性。现有研究所采用的公平性度量取决于具体情境和所针对的偏见类型，因此产生了各种优化问题，以及为满足不同应用的特定需求而量身定制的公平算法。我们对公平性的定义足够宽泛，能够涵盖许多现有概念，例如 80% 规则 [4],统计均等 [12], 和比例代表制 [18]。与大多数以求得确定性解集为目标的公平感知算法设计研究 [8,14,9,23,25], 不同，我们的目标是计算一个随机化解，使其能够在平均意义上满足群体公平性约束。

2 预备知识与问题陈述

考虑一个由 n 个元素组成的集合 V，以及一个非负子模效用函数 $f\colon 2^V \to \mathbb{R}_+$。元素 $e \in V$ 在集合 $S \subseteq V$ 上的边际效用

记为 $f(e \mid S)$，即 $f(e \mid S) = f(\{e\} \cup S) - f(S)$。若对于任意集合 X、$Y \subseteq V$（满足 $X \subseteq Y$）以及任意元素 $e \in V \setminus Y$，函数 f 满足以下不等式，则称其为子模函数：

$$f(e \mid Y) \leq f(e \mid X).$$

如果对于任意集合 $X \subseteq V$ 和任意项目 $e \in V \setminus X$，均满足以下条件，则称其为单调：

$$f(e \mid X) \geq 0.$$

假设 V 被划分为 m 个组：V_1、V_2、$\cdots$、V_m，则对可行解中必须包含的每个组的项目期望数，都有一个指定的下界和上界。这些界分别称为 $\alpha \in \mathbb{R}^m_{\geq 0}$ 和 $\beta \in \mathbb{R}^m_{\geq 0}$，代表群体公平性约束。此外，所选项目的数量还受到硬约束 b 的限制。令 $\mathcal{F} = \{S \subseteq V \mid |S| \leq b\}$ 表示可行选择的集合。公平子模最大化问题（记为 P.0）的目标是确定一个分布 $x \in [0,1]^{\mathcal{F}}$，使其定义在 $\mathcal{F}$ 中的集合上，并最大化期望效用，同时确保从每个组中选出的项目期望数均满足公平性约束。即：

P.0 $\max_{x \in [0,1]^{\mathcal{F}}} \sum_{S \in \mathcal{F}} x_S f(S)$
subject to:

$$\begin{cases} \alpha_t \leq \sum_{S \in \mathcal{F}} (x_S \cdot |S \cap V_t|) \leq \beta_t, \forall t \in [m]. \\ \sum_{S \in \mathcal{F}} x_S \leq 1. \end{cases}$$

这里，每个决策变量 x_S 表示 $S \in \mathcal{F}$的选择概率。该线性规划共有 $2m+1$ 个约束，不包括用于规定对所有 $S \in \mathcal{F}$均有 $x_S \geq 0$ 的显然约束。尽管如此，该线性规划中的变量数量仍等于 $\mathcal{F}$中的元素数量，而这个数量可能随 n呈指数增长。因此，传统的线性规划求解器无法高效求解该问题。下一个引理指出，P.0 是一个 NP-hard 问题。

引理 1. 问题 P.0 是 NP-hard 的。

证明：我们通过将其归约到经典的基数约束单调次模最大化问题来证明这一点，下面将对此问题进行描述。

定义 1. 基数约束单调次模最大化问题的输入包括一组物品 V、一个单调次模函数 f：$2^V \to \mathbb{R}_+$，以及一个基数约束 b。目标是在确保 $|S| \leq b$的同时，选择物品子集 $S \subseteq V$，使 $f(S)$ 最大化。

为展示该归约，我们取一个基数约束单调次模最大化问题的实例，并创建一个对应的P.0 实例。为此，我们只考虑一个没有公平性约束的组，即 $V = V_1$，其中有 $\alpha_1 = 0$ 和 $\beta_1 = |V|$。可以很容易验证，该实例的最优解是一个分布，分布在一组解上，其中每个解

其中一个是该基数约束单调子模最大化问题实例的最优解。此外，尽管 P.0 允许随机化解，但至少存在一个最优解是确定性集合。具体而言，基数约束单调子模最大化问题的每个最优解都必须是其对应 P.0 实例的最优解。因此，这两个实例是等价的。归约证明至此完成。 □

3 近似可行的确定性算法

本节将介绍一种用于 P.0 的确定性算法。这里假设m 各组互不重叠。首先，我们介绍单调次模函数 f的多线性扩展。给定一个向量 $y \in [0,1]^n$，令S_y 为一个随机集合，其中每个元素 $i \in V$ 都以概率 y_i独立地加入 S_y 。然后我们令

$$F(y) = \mathbb{E}[f(S_y)] = \sum_{S \subseteq V} f(S) \prod_{i \in S} y_i \prod_{i \notin S} (1 - y_i).$$

接下来，我们引入一个新的优化问题 P.1。P.1 的目标是计算一个向量 $y \in [0,1]^n$ $F\ y$ $\alpha_t \leq \sum_{i \in V_t} y_i \leq$，使 ( ) 最大化，同时满足 $\beta_t\ \forall t \in [m]$ $\sum_{t \in [m]} \sum_{i \in V_t} y_i \leq b$ ，以及 。

P.1 $\max_{y \in [0,1]^n} F(y)$
subject to:

$$\begin{cases} \alpha_t \leq \sum_{i \in V_t} y_i \leq \beta_t, \forall t \in [m]. \\ \sum_{t \in [m]} \sum_{i \in V_t} y_i \leq b. \end{cases}$$

下面的引理建立了问题 P.0 的最优解与问题 P.1 的最优解之间的联系。该引理是理解这两个问题之间关系的重要基础，并使我们能够通过求解 P.1，为 P.0 求得一个近似最优解。

引理2. 设 x^* 表示 P.0 的最优解， y^* 表示 P.1 的最优解，则有

$$(1 - 1/e) \sum_{S \in \mathcal{F}} x_S^* f(S) \leq F(y^*). \tag{1}$$

证明：令 B 为由所有满足 P.1 中条件的向量 $y \in [0,1]^n$ 构成的多面体，即，

$$B = \{y \in [0,1]^n \mid \alpha_t \leq \sum_{i \in V_t} y_i \leq \beta_t, \forall t \in [m]; \sum_{t \in [m]} \sum_{i \in V_t} y_i \leq b; 0 \leq y_i \leq 1, \forall i \in V\} \tag{2}$$

给定最优解 x^* P.0 的情况下，我们随后引入一个向量$\hat{y} \in [0,1]^n$ ，使得 $\hat{}$ $y_i = \sum_{S \in \mathcal{F}} x_S^* \cdot \mathbf{1}_{i \in S}$ ，其中 $\mathbf{1}_{i \in S} = 1$ 若 $i \in S$ ，否则为 $\mathbf{1}_{i \in S} = 0$ 。它

很容易验证，$\hat{y}_i$ 的值表示根据 x^* 定义的分布，选择项 i 的概率。接下来我们将证明，要证明这个引理，只需证明

$$\hat{y} \in B. \tag{3}$$

正如 [2], 中所确立的，如果 f 是单调且子模的，并且 $\hat{y}_i = \sum_{S \in \mathcal{F}} x_S^* \cdot \mathbf{1}_{i \in S}$，那么 $(1-1/e)\sum_{S \in \mathcal{F}} x_S^* f(S) \leq F(\hat{y})$。这里 $1-1/e$ 也称为单调子模函数的相关性间隙。假设 (3)成立，且 y^* 是 P.1 的最优解，则有 $F(\hat{y}) \leq F(y^*)$。因此，该引理直接源于以下观察：$(1-1/e)\sum_{S \in \mathcal{F}} x_S^* f(S) \leq F(\hat{y}) \leq F(y^*)$。

证明的其余部分致力于证明$\hat{y} \in B$。首先，由于 x^* 是 P.0的一个可行解，因此有 $\alpha_t \leq \sum_{S \in \mathcal{F}}(x_S^* \cdot |S \cap V_t|) \leq \beta_t, \forall t \in [m]$。由此可得 $\alpha_t \leq \sum_{i \in V_t} \hat{y}_i \leq \beta_t, \forall t \in [m]$，这是因为 $\sum_{S \in \mathcal{F}}(x_S^* \cdot |S \cap V_t|) = \sum_{i \in V_t} \hat{y}_i$ 表示根据 x^*所定义的分布从组 V_t 中选取项目的期望项目数。其次，由于 x^* 是P.0 的一个可行解，根据 x^* 所定义的分布选取的项目数至多为 b。因此，$\sum_{t \in [m]} \sum_{i \in V_t} \hat{y}_i \leq b$。第三，很容易证明 $0 \leq \hat{y}_i \leq 1, \forall i \in V$。这就完成了对 $\hat{y} \in B$的证明。 □

3.1 算法设计

接下来，我们介绍我们的算法。首先，我们使用连续贪心算法为 P.1 计算一个分数解，然后将其舍入以获得一个整数解。

算法 1 连续贪心算法

设置 $\delta = 9n^2, l = 0, y^0 = [0]^n$。
当 $l < \delta$ 时执行
    对于每个 $i \in V$，估计 $F(i \mid y^l)$
    求 P.A 的最优解 $z \in [0,1]^n$

> P.A $Maximize_y \sum_{i \in V} y_i F(i \mid y^l)$
> 约束条件：$y \in B$。

    $y^{l+1} = y^l + z$
    增量 $l = l + 1$
$y' \leftarrow y^{\delta}$
返回 y'

连续贪心算法。我们首先详细介绍连续贪心算法（见算法 1）。该算法框架最初由 [6] 提出，我们对其进行调整，以便在多面体 B（见式 (2)）中求得分数解。请注意，多面体 B 不是向下封闭的，这给我们的研究带来了独特的挑战。该算法维护

一个分数解 $y^l \in [0,1]^n$，初始值为 $y^0 = (0, 0, \cdots, 0)$。在每一轮 l 中，它会根据 F，计算在 y^l 的基础上加入每个元素 $i \in V$ 的边际效用，如下所示：

$$F(i \mid y^l) = F(\mathbf{e}_i \vee y^l) - F(y^l). \tag{4}$$

其中，$\mathbf{e}_i \in \{0,1\}^n$ 是在第 i个坐标中取值为1、其他位置取值为0 的向量；$\mathbf{e}_i \vee y^l$ 表示向量 $\mathbf{e}_i$ 和 y^l 的逐元素最大值。

然后我们求解以下线性规划问题 P.A，该问题为每个项目 i 分配权重 $F(i \mid y^l)$，并寻求 B 中的最大加权向量。

P.A $Maximize_y \sum_{i \in V} y_i F(i \mid y^l)$ 约束条件为：$y \in B$.

在第 l 轮求解P.A并获得一个最优解 $z \in [0,1]^n$后，我们将分数解更新如下：$y^{l+1} = y^l + z$。经过 δ 轮，其中$\delta = 9n^2$，y^δ 被作为最终解 y'返回。

舍入。接下来，我们采用 Pipage舍入 [1], 一种将线性松弛进行舍入的简单确定性过程，将 y' 舍入为整数解。该算法由三个阶段组成。

– 阶段 1：对于每个 $t \in [m]$，重复执行以下操作，直到 V_t 至多有一个非整数坐标：选择任意两个分数坐标 i、j，使得 $i, j \in V_t$。计算 $\theta_1 = \min\{1 - y'_i, y'_j\}$ 和 $\theta_2 = \min\{y'_i, 1 - y'_j\}$。创建两个向量，$y^a = y' + \theta_1(\mathbf{e}_i - \mathbf{e}_j)$ 和 $y^b = y' + \theta_2(\mathbf{e}_j - \mathbf{e}_i)$。如果 $F(y^a) \geq F(y^b)$，则设置 $y \leftarrow y_a$，否则设置 $y \leftarrow y^b$。– 阶段 2：假设 y_1、$\cdots$、y_k 是剩余的分数坐标。重复第一阶段中的相同过程，直到 y 至多有一个非整数坐标。– 阶段 3：令 i 表示最后一个非整数坐标（如果有）。设置 $y_i = 1$。输出 $A \subseteq V$，其在 y 中的坐标为 1。

请注意，类似的框架已被用于在确定性设定下解决公平的子模最大化问题 [8]。该问题旨在确定一个固定的项目集合，在满足群体公平性约束的同时优化一个子模函数。他们的方法在舍入阶段与我们的方法有相似之处，但不需要第三阶段。这是因为在他们的设定中，α_t 和 β_t 都是整数，因此他们能够确保在前两个舍入阶段之后不存在非整数坐标。

3.2 性能分析

回顾一下，x^* 表示 P.0 的最优解，令 $OPT = \sum_{S \in \mathcal{F}} x^*_S f(S)$ 表示最优解的效用。以下定理指出，A，即我们的算法返回的解集，是 P.0 的近似可行解，且其效用至少为 $(1 - 1/e)^2 OPT$。

定理1.设 A 为我们的算法返回的集合， OPT 为 P.0 的最优解的效用。由此可得：

$$f(A) \geq (1-1/e)^2 OPT. \tag{5}$$

此外， A 始终满足基数约束，并且近似满足 P.0 的公平性约束，即 $|A| \leq b$ 和 $\lfloor \alpha_t \rfloor \leq |A \cap V_t| \leq \lceil \beta_t \rceil, \forall t \in [m]$。

证明：我们首先证明 $|A| \leq b$ 始终成立。注意，连续贪心算法得到的分数解 y' 属于 B，因此，$\sum_{i \in V} y'_i \leq b$。此外，舍入阶段的阶段 1 和阶段 2 不会改变该值，而阶段 3 会将最后一个非整数坐标（如果存在）舍入为 1。因此，$|A| \leq \lceil \sum_{i \in V} y'_i \rceil \leq b$，其中第二个不等式依据以下观察： $\sum_{i \in V} y'_i \leq b$ 且 b 是整数。

接下来，我们证明 A 几乎满足 P.0 的公平性约束，即$\lfloor \alpha_t \rfloor \leq |A \cap V_t| \leq \lceil \beta_t \rceil$, $\forall t \in [m]$。由于 $y' \in B$，因此有 $\alpha_t \leq \sum_{i \in V_t} y'_i \leq \beta_t, \forall t \in [m]$。注意，阶段 1 不会改变该值；阶段 2 和阶段 3 至多将每个组中的一个分数坐标舍入为二元值。因此，$\lfloor \sum_{i \in V_t} y'_i \rfloor \leq |A \cap V_t| \leq \lceil \sum_{i \in V_t} y'_i \rceil, \forall t \in [m]$。结合 $\alpha_t \leq \sum_{i \in V_t} y'_i \leq \beta_t, \forall t \in [m]$, 可得 $\lfloor \alpha_t \rfloor \leq |A \cap V_t| \leq \lceil \beta_t \rceil, \forall t \in [m]$。

最后，我们证明 A 的近似比。回顾， y^* 表示 P 的最优解。**1**, [6] 已证明，如果 f 是单调且子模的，那么从连续贪心算法返回的分数解 y' 的效用至少为 $(1-1/e)F(y^*)$，即 $F(y') \geq (1-1/e)F(y^*)$。结合引理2，这意味着 $F(y') \geq (1-1/e)^2 \sum_{S \in \mathcal{F}} x^*_S f(S) = (1-1/e)^2 OPT$。为了证明 $f(A) \geq (1-1/e)^2 OPT$，只需证明 $f(A) \geq F(y')$。接下来我们证明这一不等式。注意，在舍入阶段的阶段 1 和阶段 2 中，我们执行 Pipage 舍入，将 y' 舍入为一个至多包含一个非整数坐标的向量 y 。根据 [6], , Pipage 舍入不会降低 y' 的期望效用，即 $F(y) \geq F(y')$。在阶段 3 中，如果 y 中存在最后一个非整数坐标，我们将其舍入为 1。根据 f 是单调的这一假设，该操作不会降低 y 的期望效用。因此， $F(y) \geq F(y')$ 仍然成立。回顾， y 是 A 的指示向量，因此， $f(A) = F(y)$。因此，$f(A) = F(y) \geq F(y')$。 □

注释 1：由前述定理可立即得出，如果对于所有 $t \in [m]$， α_t 和 β_t 都是整数，那么我们的解将严格满足问题 P.0 的所有公平性约束。

3.3 快速贪心算法

我们之前的算法需要求解一个多线性松弛问题，这一过程可能较慢且计算开销较大，尤其是在处理大规模问题时。本节将介绍一种简单的贪心算法，它能显著提高速度，但代价是近似比有所下降。

尽管问题 P.0 允许使用随机化解，定理 1 表明，确定性解足以为问题 P.0 获得常数因子近似。接下来，我们将介绍一种简单的贪心算法，它能够有效地找到

一个近似最优的确定性解，进而为问题 P.0 带来常数因子近似。为此，我们引入一个新的优化问题 P.2，即问题 P.0 的确定性版本（采用放宽后的公平性约束）。

P.2 最大化$_{S\in\mathcal{F}}$ $f(S)$
约束条件：
$\lfloor\alpha_t\rfloor \le |S\cap V_t| \le \lceil\beta_t\rceil,\ \ \forall t\in[m]$。

请注意，在 P.2 中，我们使用 $\lfloor\alpha_t\rfloor$ 和 $\lceil\beta_t\rceil$ 作为下界和上界，因此 P.2 的可行解是原问题 P.0 的近似可行解。以下引理表明，P.2 的最优解可达到问题 P.0 的 $(1-1/e)^2$近似。

引理 3. 令 A^{P2}表示 P.2 的最优解，则有 $f(A^{P2}) \ge (1-1/e)^2 OPT$，其中 OPT 是 P.0 的最优解。

证明：回顾定理 1，我们证明了 $f(A) \ge (1-1/e)^2 OPT$，其中 A满足 P.2 中的所有约束。由于 A^{P2} 是 P.2 的最优解，因此有 $f(A^{P2}) \ge f(A) \ge (1-1/e)^2 OPT$。 □

接下来，我们介绍一种简单的贪心算法，以获得 P.2 的 $1/2$ 近似解。首先，我们介绍 P.3，这是 P.2 的一个松弛问题。

P.3 max$_{S\subseteq V}$ $f(S)$
约束条件：
$\begin{cases} \sum_{t\in[m]} \qquad\qquad \max \\ \{\lfloor\alpha_t\rfloor, |S\cap V_t|\} \le b. \end{cases}$

容易验证，P.2 的任何可行解都必须是 P.3 的可行解。因此，$f(A^{P2}) \le f(A^{P3})$，其中 A^{P3}是 P.3 的最优解。已有研究表明，P.3 中列出的约束构成一个拟阵 [14]。因此，P.3 是一个受拟阵约束的经典次模最大化问题。简单的贪心算法可以保证 P.3 的 $1/2$ 近似解[19]。该算法通过迭代地将元素添加到解集中来运行，使得在每一步中目标函数值的边际增量最大化，同时满足拟阵约束；当当前解集无法继续扩展时，算法终止。令 A^g 表示贪心算法返回的解，则有

$$f(A^g) \ge (1/2) f(A^{P3}) \ge (1/2) f(A^{P2}). \tag{6}$$

此外，很容易验证，A^g 必须是 P.2 的可行解。

定理 2. 令 A^g 表示贪心算法返回的解，则有 $f(A^g) \ge (1-1/e)_2^{\ 2} \cdot OPT$。此外，$A^g$始终满足基数约~~束，并~~且近似满足 P.0 的公平性约束，即 $|A^g| \le b$ and $\lfloor\alpha_t\rfloor \le |A^g\cap V_t| \le \lceil\beta_t\rceil, \forall t\in[m]$。

证明：定理第一部分的证明源自不等式 (6) 和引理 3。定理第二部分成立是因为 A^g 是问题 P.2 的一个可行解。 □

4 一种可行的 $(1 - 1/e)$ 近似随机化算法

现在，我们介绍一种用于 P.0 的随机化算法。与上一节介绍的结果不同，我们的随机化解具有三项优势：（1）我们的解不依赖于组之间互不重叠这一假设；（2）我们的解严格满足所有公平性约束；（3）我们达到了最优近似比 $1 - 1/e$。

如前所述，P.0 的变量数量等于 $\mathcal{F}$ 中元素的数量；当 n 较大时，这一数量可能会变得极其庞大。这意味着标准 LP 求解器无法有效处理这个 LP 问题。为了解决这一问题，我们转而求解其对应的对偶问题（P.0 的对偶问题），并使用椭球算法 [16] 来求解。在对偶问题中，我们为每个组 V_t 分配两个“权重” $z_t \in \mathbb{R}_{\geq 0}$ 和 $u_t \in \mathbb{R}_{\geq 0}$，此外还有一个变量 $w \in \mathbb{R}_{\geq 0}$。

P.0 的对偶 $\min_{z \in \mathbb{R}^m_{\geq 0}} \max_{u \in \mathbb{R}^m_{\geq 0}, w \in \mathbb{R}_{\geq 0}} \sum_{t \in [m]} (\beta_t u_t - \alpha_t z_t) + w$ 满足： $w \geq f(S) + \sum_{t \in [m]} \lvert S \cap V_t \rvert \cdot (z_t - u_t) \ \forall S \in \mathcal{F}.$

从高层次来看，椭球法是一种强大的工具，用于判断给定的非退化凸集 C 是否为空。对于我们的问题， C 表示 P.0 的对偶问题的可行域。该方法首先定义一个保证包含 C 的椭球。在每次迭代中，算法都会检查当前椭球的中心是否位于 C 内。如果是，则意味着 C 非空，且当前解可行。在这种情况下，该方法会尝试一个更小的目标值。另一方面，如果当前椭球的中心不在 C 内，该方法会使用一个（近似）分离预言机来识别一条被违反的约束，并找到一个更小的椭球，其中心满足该约束。这个过程会持续进行，直到找到一个可行解，或者包围椭球的体积小到足以认为 C 为空，即找不到目标值更小的可行解。值得注意的是，该方法以几何方式运作，不需要显式描述线性规划。唯一的要求是存在一个多项式时间的（近似）分离预言机，它能够判断一个点是否位于 C 内；如果不在，则返回一个分离超平面。该方法在求解线性问题时只需进行多项式次数的迭代。

在我们的问题背景下，我们近似求解 SubMax 问题，以检查当前解的可行性，并充当分离预言机。

定义 2（SubMax）。给定效用函数 f、基数约束b以及两个向量 $z \in \mathbb{R}^m_{\geq 0}$ 和 $u \in \mathbb{R}^m_{\geq 0}$，SubMax$(z, u, b)$旨在

$$\max_{S \in \mathcal{F}} (f(S) + \sum_{t \in [m]} |S \cap V_t| \cdot (z_t - u_t)). \tag{7}$$

SubMax(z, u, b)要求一个集合 S，其大小至多为 b，使得 $f(S)+\sum_{t\in[m]} |S\cap V_t| \cdot (z_t - u_t)$最大化。注意到 f 是非负单调子模的；而$\sum_{t\in[m]} |S \cap V_t| \cdot (z_t - u_t)$关于 S是一个模函数，因此， SubMax(z, u, b)是在基数约束下最大化一个非负单调子模函数与一个模函数之和的经典问题。 [21] 提出了一种随机多项式时间算法，该算法生成一个集合 A ，使得对于每个 $S \in \mathcal{F}$，都有

$$f(A) + \sum_{t \in [m]} |A \cap V_t| \cdot (z_t - u_t) \geq (1 - 1/e) f(S) + \sum_{t \in [m]} |S \cap V_t| \cdot (z_t - u_t). \tag{8}$$

现在我们可以陈述本节的主要定理。

定理3.存在一个多项式时间（$1-1/e$）近似算法（带有加性误差 ϵ）用于 P.0。

本节其余部分致力于证明定理3，即我们为 P.0 提出一个多项式 $(1 - 1/e)$-近似算法。令 $C(L)$ 表示满足以下条件的 $(z \in \mathbb{R}^m_{\geq 0}, u \in \mathbb{R}^m_{\geq 0}, w \in \mathbb{R}_{\geq 0})$ 的集合，即

$$\sum_{t \in [m]} (\beta_t u_t - \alpha_t z_t) + w \leq L,$$

$$w \geq f(S) + \sum_{t \in [m]} |S \cap V_t| \cdot (z_t - u_t), \forall S \in \mathcal{F}.$$

我们使用二分搜索确定使 $C(L)$ 非空的最小 L 。对于给定的 L，我们首先评估不等式 $\sum_{t\in[m]}(\beta_t u_t - \alpha_t z_t) + w \leq L$。然后，我们运行来自 [21]（标记为 $\mathcal{A}$）的算法来求解 Sub-Max(z, u, b)。设 A 为 $\mathcal{A}$ 返回的解集。

– 如果 $f(A) + \sum_{t\in[m]} |A \cap V_t| \cdot (z_t - u_t) \leq w$，则将 $C(L)$ 标记为非空。在这种情况下，我们尝试一个更小的 L。– 如果 $f(A) + \sum_{t\in[m]} |A \cap V_t| \cdot (z_t - u_t) > w$，则 $(z, w) \notin C(L)$，且 A 是一个分离超平面。我们找到一个中心满足该约束的更小椭球。此过程会持续进行，直到在 $C(L)$ 中找到一个可行解（此时，我们尝试一个更小的 L），或者包围椭球的体积小到足以将 $C(L)$ 视为空集（此时，我们得出当前目标无法实现的结论，因此将尝试一个更大的L）。

若要了解使用分离预言机运行椭球算法所涉及的具体步骤，以及如何实现（乘法和加法）近似保证，建议参阅 [5] 的第 2 章。假设 L^* 是算法判定 $C(L)$ 非空时 L 的最小值。因此，存在一个 (z^*, u^*, w^*)，使得

$$\sum_{t\in[m]} (\beta_t u_t^* - \alpha_t z_t^*) + w^* \leq L^* \tag{9}$$

and

$$f(A) + \sum_{t\in[m]} |A \cap V_t| \cdot (z_t^* - u_t^*) \leq w^* \tag{10}$$

其中，A 是算法 $\mathcal{A}$ 求解 $\textsc{SubMax}(z^*, u^*, b)$后得到的输出。令 $\mu = 1 - 1/e$，可得 $\forall S \in \mathcal{F}$,

$$\begin{aligned} f(S) + \sum_{t\in[m]} |S \cap V_t| \cdot (u_t^* - z_t^*)/\mu &\leq (f(A) + \sum_{t\in[m]} |A \cap V_t| \cdot (u_t^* - z_t^*))/\mu \\ &\leq w^*/\mu \end{aligned} \tag{11}$$

其中，第一个不等式由式（8）得出，第二个不等式由不等式（10）得出。此外，不等式（9）意味着娱乐主管

$$\sum_{t\in[m]} (\beta_t u_t^* - \alpha_t z_t^*)/\mu + w^*/\mu \leq L^*/\mu. \tag{12}$$

不等式 (11) 意味着 $(z^*/\mu, u^*/\mu, w^*/\mu)$ 是 P.0 的对偶问题的一个可行解。结合不等式 (12)，这意味着 P.0 的对偶问题的最优解，从而 P.0 的最优解（根据强对偶性），有上界 $\frac{1}{\mu} \cdot L^*$。通过找到一个取值为 L^* 的 P.0 的解，我们便获得了原始问题 P.0 的一个 μ-近似解。这里，我们说明如何仅使用与分离预言机找到的分离超平面相对应的可行解集合来计算这样的解。假设 $L^* - \epsilon$ 是算法判定 $C(L)$ 为空的 L 的最大值，其中ϵ 由算法的精度决定。令 $\mathcal{F}'$ 表示在椭球算法于 $C(L^* - \epsilon)$ 上的实现过程中违反对偶约束的 $\mathcal{F}$ 子集。那么，$|\mathcal{F}'|$ 是多项式的。我们考虑以下多项式规模的 P.0 对偶问题（标记为多项式规模的 P.0 对偶问题），其中使用了来自 $\mathcal{F}'$的分离超平面。

P.0 的多项式规模对偶 $\min_{z\in\mathbb{R}^m_{\geq 0}, u\in\mathbb{R}^m_{\geq 0}, w\in\mathbb{R}_{\geq 0}} \sum_{t\in[m]} (\beta_t u_t - \alpha_t z_t) + w$ 满足约束：$w \geq f(S) + \sum_{t\in[m]} |S \cap V_t| \cdot (z_t - u_t)\ \forall S \in \mathcal{F}'$.

因为 $C(L^* - \epsilon)$为空，多项式规模的P.0的对偶问题的最优解的值至少为 $L^* - \epsilon$。因此，多项式规模的P.0的对偶问题的对偶问题（列于多项式规模的 P.0 中）的值至少为 $L^* - \epsilon$。注意，多项式规模的 P.0 具有多项式规模。

多项式规模的 P.0 最大值 $\max_{x\in[0,1]^{\mathcal{F}'}} \sum_{S\in\mathcal{F},} x_S f(S)$
满足：

$$\begin{cases} \alpha_t \le \sum_{S\in\mathcal{F}} x_S \cdot |S \cap V_t| \le \beta_t \ \forall t \in [m]. \text{ ( )} \\ \sum_{S\in\mathcal{F}'} x_S \le 1. \end{cases}$$

，

回顾，P.0的最优解的上界为$\frac{1}{\mu}\cdot L^*$，从多项式规模的P.0中获得最优解可为 P.0 提供 μ-近似（加性误差为 ϵ），其中 $\mu = 1 - 1/e$。

5 结论

我们引入一组公平性准则，包括受限支配和少数群体保护，以确保从每个组中选出的项目比例都在期望范围内。我们提出了一种近似比为 $(1 - 1/e)^2$ 的确定性算法，以及一种能够近似满足公平性约束的快速贪心确定性算法。此外，我们还提出了一种严格满足所有公平性约束并达到最优近似比 $1 - 1/e$ 的随机化算法。

参考文献

1. Ageev, A.A., Sviridenko, M.I.：Pipage 舍入：一种构造具有已证明性能保证的算法的新方法。《组合优化杂志》8，307–328（2004） 2. Agrawal, S., Ding, Y., Saberi, A., Ye, Y.：相关性鲁棒随机优化。载于：第二十一届 ACM-SIAM 离散算法年度研讨会论文集。第 1087–1096 页。SIAM（2010） 3. Bera, S., Chakrabarty, D., Flores, N., Negahbani, M.：聚类公平算法。神经信息处理系统进展 32（2019） 4. Biddle, D.：不利影响与测试验证：有效且经得起质疑的就业测试实务指南。Routledge（2017） 5. Bubeck, S. 等：凸优化：算法与复杂性。机器学习基础与趋势 8(3-4)，231–357（2015） 6. Calinescu, G., Chekuri, C., P´al, M., Vondr´ak, J.：在拟阵约束下最大化次模集合函数。载于：整数规划与组合优化国际会议论文集。第 182–196 页。施普林格（2007） 7. Celis, E., Keswani, V., Straszak, D., Deshpande, A., Kathuria, T., Vishnoi, N.：公平且多样化的基于 DPP 的数据摘要。载于：国际机器学习会议论文集。第 716–725 页。PMLR（2018） 8. Celis, L.E., Huang, L., Vishnoi, N.K.：具有公平性约束的多获胜者投票。载于：第 27 届人工智能国际联合会议论文集。第 144–151 页（2018） 9. Chierichetti, F., Kumar, R., Lattanzi, S., Vassilvtiskii, S.：拟阵、匹配与公平性。载于：第 22 届人工智能与统计国际会议论文集。第 2212–2220 页。PMLR（2019）

10. Das, A., Kempe, D.：线性回归中的子集选择算法。载于：第四十届 ACM 计算理论年度研讨会论文集。第 45–54 页（2008） 11. Dueck, D., Frey, B.J.：用于无监督图像分类的非度量亲和传播。载于：2007 年 IEEE 第 11 届国际计算机视觉会议。第 1–8 页。IEEE（2007） 12. Dwork, C., Hardt, M., Pitassi, T., Reingold, O., Zemel, R.：通过认知实现公平。载于：第三届理论计算机科学创新会议论文集（2012） 13. El-Arini, K., Guestrin, C.：超越关键词搜索：发现相关科学文献。载于：第 17 届 ACM SIGKDD 知识发现与数据挖掘国际会议论文集。第 439–447 页（2011） 14. El Halabi, M., Mitrovi´c, S., Norouzi-Fard, A., Tardos, J., Tarnawski, J.M.：流式次模最大化中的公平性：算法与困难性。神经信息处理系统进展 33，13609–13622（2020） 15. Golovin, D., Krause, A.：自适应次模性：在主动学习和随机优化中的理论与应用。《人工智能研究杂志》42，427–486（2011） 16. Gr¨otschel, M., Lov´asz, L., Schrijver, A.：椭球法及其在组合优化中的后果。Combinatorica 1(2)，169–197（1981） 17. Joseph, M., Kearns, M., Morgenstern, J.H., Roth, A.：学习中的公平性：经典老虎机与上下文老虎机。神经信息处理系统进展 29（2016） 18. Monroe, B.L.：完全比例代表制。《美国政治学评论》89(4)，925–940（1995） 19. Nemhauser, G.L., Wolsey, L.A., Fisher, M.L.：最大化次模集合函数的近似算法分析（一）。《数学规划》14(1)，265–294（1978） 20. Sipos, R., Swaminathan, A., Shivaswamy, P., Joachims, T.：利用次模词汇覆盖进行时间语料摘要。载于：第 21 届 ACM 信息与知识管理国际会议论文集。第 754–763 页（2012） 21. Sviridenko, M., Vondr´ak, J., Ward, J.：有界曲率下次模与超模优化的最优近似。《运筹学数学》42(4)，1197–1218（2017） 22. Tang, S., Yuan, J.：具有部分反馈的影响力最大化。《运筹学快报》48(1)，24–28（2020） 23. Tang, S., Yuan, J.：自适应次模最大化中的群体公平。arXiv 预印本 arXiv:2207.03364（2022） 24. Tsang, A., Wilder, B., Rice, E., Tambe, M., Zick, Y.：影响力最大化中的群体公平。arXiv 预印本 arXiv:1903.00967（2019） 25. Yuan, J., Tang, S.：非单调次模最大化中的群体公平。《组合优化杂志》45(3)，88（2023） 26. Zafar, M.B., Valera, I., Rogriguez, M.G., Gummadi, K.P.：公平性约束：实现公平分类的机制。载于：人工智能与统计。第 962–970 页。PMLR（2017）