arXiv:2601.15178v1 [cs.CC] 21 Jan 2026

具有隐私特征的数据分类问题的复杂度分析与实际求解*

DavidPantoja[1],Ismael Rodr´ıguez[1,2],
Fernando Rubio[1,2],Clara Segura[1], †‡§

摘要

本文研究这样一个问题：给定一个总体的概率分布，如何对一个未知个体进行提问，使其能够代表该分布，从而显著降低我们对某些特征的不确定性，同时不降低我们对其他被视为隐私或敏感信息的特征的不确定性。因此，该问题的目标是在保护个体隐私的同时，提取与合法目的相关的信息。这对于实现多个领域中的非侵扰性选拔流程至关重要。例如，这对于设计公司和机构中不带歧视的人员选拔、晋升和裁员流程至关重要；对于获取与公司所提供服务相关的客户信息（仅限于此）也至关重要；对于确保认证过程不泄露与认证本身无关的敏感工业信息同样至关重要。为此，我们构建了交互式提问流程，这要求对 *decision trees*的概念加以推广，以衡量计划中每个分支所获取的期望信息和非期望信息的数量。我们对这一问题的研究兼具理论与实践意义：一方面，我们通过从集合覆盖问题归约，证明了该问题是NP完全的；另一方面，鉴于其难解性，我们提出了启发式解，以便在可接受的时间内找到合理的解。具体而言，本文提出了一种贪心算法和两种遗传算法。实验表明，采用贪心策略强化的遗传算法取得了最佳结果。

关键词：计算复杂性、NP完全性、数据分类、决策树、选拔流程、隐私保护。

*本文发表于《Complex & Intelligent Systems》。当前版本为作者接受稿。本研究部分由西班牙项目 PID2019-108528RB-C22 和 PID2023-149943OB-I00 资助。

†1 计算机系统与计算系。信息学院。马德里康普顿斯大学。西班牙马德里 28040。

‡2知识技术研究所。§电子邮件：dpantoja@ucm.es, isrodrig@sip.ucm.es, fernando@sip.ucm.es,clsegura@ucm.es

1 引言

在与他人互动时，我们会主动或无意中透露个人信息。人们通常认为，特定年龄、性别、族裔、文化、教育背景、财富水平等个体特征具有某些共性，因此，个体对于看似无关紧要的、涉及自身情况或习惯的问题所作的回答，综合起来可能以很高的概率表明他们属于某个特定群体，而这可能成为歧视的根源 [24, 19, 17]。尽管透露某些信息是提供服务或完成流程所必需的（例如医学测试或求职面试），但某些已透露的信息可能会使人发现与当前流程无关的隐私特征，或者至少提供很少的相关信息；而通过另一种方式，也能获得足够的相关信息，同时几乎不透露任何敏感问题相关的信息。因此，设计提问流程至关重要：无论何种情况，答案都应能充分说明当前关注的问题（例如某人是否胜任某项工作），同时避免过度透露有关其他隐私特征的信息（例如宗教或性取向）。

从技术角度而言，我们希望大幅降低[1] 对前一类问题的不确定性（例如，让答案在所有情况下都能以高于 85% 或低于 15% 的概率揭示候选人是否胜任该工作），同时不显著降低对后一类问题的不确定性（例如，如果总体人群中有 30% 具有某种敏感私人特征，那么对于我们的问题方案可能产生的任何答案，我们都希望在给定答案的条件下，具有该特征的概率既不低于 20%，也不高于 40%）。请注意，我们无法通过改变共同知识基础来实现这一点，因为这些信息要么是任何受过足够教育的人都熟知的，要么已通过公众调查发布（例如，众所周知，某些宗教的信徒不吃猪肉）。因此，我们考虑设计自适应 问卷（即下一个要提出的问题可能取决于之前的答案），在尽可能多 地推断目标特征的同时，尽可能少地推断那些被视为隐私的特征。

我们的问题的一个应用，是设计公司或机构中不存在歧视的人员选拔、晋升或裁员流程；这是当今备受关注的问题 [4, 16]。设想这样一种情境：人力资源经理需要设计一个选拔流程，该流程

[1]请注意，不确定性有多种类型。它们可以是内部的，也可以是外部的：内部不确定性仅仅源于知识的缺乏（无论是完全缺乏还是部分缺乏），而外部不确定性则是不可约的，例如由随机效应导致的不确定性。不确定性也可以分为参数不确定性和非参数不确定性。非参数不确定性源于模型不正确，而参数不确定性是指在假设模型正确的情况下，对模型参数的估计不够精确。在我们的情形中，我们将处理内部不确定性和参数不确定性：我们假设申请人的答案不存在随机性，并假设申请人的特征遵循给定的总体统计数据集，尽管我们目前尚不了解这些特征。

由一系列问题组成，旨在尽可能多地获取与工作相关的信息。这些信息用于预测每位受访者未来的表现，从而判定其是否适合该职位。与此同时，经理希望确保无法从受访者对这些问题的答案中推断出任何被视为私有的信息，例如国籍、宗教或性取向。我们假设经理可以获取一个规模较大的数据集，其中包含个体的工作绩效信息，而数据集中标注的个体特征，恰好就是经理可能会问或不会问的问题。该数据集可能来源于公司员工的历史数据，也可能来自有关人员绩效的一般常识。我们假设，受访者具有某些特征的似然等于数据集中具有这些特征的个体所占的比例。因此，有时会将数据集中的个体称为候选人。

例如，假设数据集总体中同性恋者的比例为 10%。那么，对于任何给定的受访者，其对设计好的访谈中问题的答案必须能够让我们以这样的方式确定其适应度：以相同方式作答的人所构成的子集中同性恋者的比例不会明显高于或低于 10%。通过这种方式，我们可以确保这些额外信息不能被用于歧视性目的，因为具有或不具有某种潜在歧视性特征的概率，与数据集中的相应概率大致相同（假设该数据集反过来能够代表潜在候选人或总体人群）。也就是说，在回答任何问题之前后，对于该潜在歧视性特征的不确定性基本保持不变。

我们可以从乍看之下似乎与隐私信息无关的问题中推断出隐私信息。许多问题在与某些其他问题结合提问时，可以推断出隐私信息，尽管单独提问时可能无法推断出任何隐私信息。相反，有些问题单独提问时可能会推断出隐私信息，但在获得某些信息后再提问，可能就无法推断出此类隐私信息。例如，考虑这样一个社会：素食主义与信仰某种宗教并不相关，并且假设我们已经知道受访者不吃鸡肉。那么，随后询问他们是否吃猪肉，很可能不会揭示其宗教信仰，因为该受访者可能是素食主义者，或者只是不喜欢鸡肉。正因如此，设计访谈时应使先前问题的回答能够影响后续问题的选择，这一点具有重要意义。从这个意义上说，这些访谈将是交互式的。

所讨论的问题也可用于其他不同的场景。例如，在为某项服务（如网页）设计用户分类方法时，如果希望确保这些方法不会提取与所提供服务无关的用户敏感信息，那么本质上也是同一个问题。这对于自动创建用户档案非常有价值，例如用于个性化广告的档案。这些档案是根据用户提供的信息创建的，其中包括通过 Cookie 获取的信息。

创建这些档案必须遵守隐私和个人数据保护法，因此，如今设计网页用户分类方法，使其不会直接（例如通过表单）或间接（例如通过浏览历史和点击率）提取敏感信息，具有重大意义 [29, 6]。这个问题也可应用于产品或服务的认证流程设计，以确保不会泄露工业机密等私人数据。

我们将证明该问题是 NP 完全的。虽然我们的访谈可以视为决策树，但在我们的问题中构造这类树的 NP困难性，并非源于构造最小决策树的 NP困难性 [18], ，因为即使不限制树的规模或深度，我们的问题仍然是 NP-困难的——而这项限制对于证明构造仅降低某些特征不确定性的决策树具有 NP困难性是必要的。[2] 相反，在我们的情形中，困难之处在于，额外的问题会降低目标特征和敏感特征的不确定性，因此通常这两个目标相互矛盾。

鉴于该问题具有 NP困难性，除非 $P = NP$，否则无法高效地计算精确解。不过，我们还将证明，利用启发式算法，可以在可接受的时间内找到合理的解。具体而言，我们将介绍一种用于计算次优解的贪心算法，并介绍遗传算法来解决同一问题。

据我们所知，本文提出的问题是首个研究决策树构建的问题，其目标是在尽可能保持其他特征不确定性原样的同时，降低某些特征的不确定性。如上所述，我们的问题不同于决策树构建问题，难点源于两个相互矛盾的目标，而非任何规模限制。此外，正如我们将在第 1.1 节中看到的，我们的问题不同于机器学习数据分类 技术，因为我们必须避免准确分类个体的某个标签，同时尝试分类另一个标签；它也不同于隐私学习技术，因为原始信息被假设为已广为人知，因此不能对其进行剪枝或混淆；而且它与随机回答和负面调查 方法无关，因为我们的目标是获取有关某个个体的信息，而非总体人群的信息。本文的主要贡献如下：（1）正式定义具有隐私特征的数据分类问题；（2）正式证明该问题为 NP完全；（3）提供算法以解决实际情境中的问题；（4）开展实验研究，并对结果进行统计分析。

本文其余部分的结构安排如下。接下来，我们将讨论相关工作，说明我们的问题与其他某些方面

[2]需要注意的是，如果在这种情况下不考虑大小或长度限制，那么我们只需分析一棵在所有分支中都提出所有问题的树，这样问题就变得微不足道了。

与我们的问题相似的问题之间的区别。随后，第2节将对所研究的问题进行形式化介绍和定义。然后，我们将在第3节概述该问题的 NP 完全性证明。第4节介绍用于解决该问题的不同算法，并通过具体基准测试对这些算法进行评估。最后，第5节给出结论并探讨未来的研究方向。感兴趣的读者还可以在附录 A 中找到我们问题 NP 完全性的详细形式化证明。

需要指出的是，第3节（以及附录 A 中给出的相应形式化证明）与后文讨论的算法和实验无关。证明该问题的时间复杂度对于论证使用启发式方法解决该问题的合理性具有重要意义（例如参见 [10, 11]），但实践型读者可以放心跳过有关计算复杂性的该节及附录，专注于该问题的实际解决方案。

1.1 相关工作

据我们所知，本文提出并分析的问题此前尚未在相关文献中得到研究。下面我们介绍若干与之存在某种关联的问题，并说明它们与我们的研究有何不同。这些问题可以分为三组。

决策树。 给定一个有限的可测试对象集合 X、一个有限的测试集合T以及 $w \in \mathbb{N}$，其中对于每个测试 $t \in T$ 和对象 $x \in X$，$t(x)$ 的值非真即假，决策树问题旨在确定是否存在一个识别程序（以二叉决策树的形式），能够完全识别集合中的任意给定对象，并且其外部路径长度（从树的叶节点到根节点的路径长度之和）小于 w。已知决策树问题是 NP完全问题 [18]。类似地，当可能的正确行为和错误行为的集合以外延方式定义，即逐个情况定义时，人们也研究了若干变体问题：是否可以通过自适应测试来区分被测实现 （IUT）的正确性 [25]。这些变体并不限制外部路径长度，而是要求限制树的深度。此外，测试可能具有非二元答案，并且某些变体允许 IUT 表现出非确定性行为。

我们的问题与这些研究中探讨的问题不同，因为我们并不关心限制可以提出的问题（或测试）的数量，而是关注限制这些树在任何情况下都允许推断出的私有信息量。一般而言，提出更多问题会获得更多期望的和非期望信息，因此必须谨慎设计这些问题，使其在提问树的所有分支中都能大幅降低前者的不确定性，同时不显著降低后者的不确定性。因此，树的大小（以深度或节点数衡量）并不受问题定义本身的限制，而是受避免分支泄露过多非期望信息这一需求所限制。实际上，正是这两种（通常相互矛盾的）力量之间微妙的相互作用

使树的大小通常趋于平衡。[3] 由于没有明确的大小限制，我们的问题并非先前任何问题的泛化，因此需要专门的 NP-困难性证明；而且，由于它也不是某个问题的特例，因此同样需要专门证明其属于 NP。

机器学习数据分类技术。该领域的研究将机器学习用于数据分类 [27]。分类器使用包含个体信息的数据集进行训练，这些信息以数据元组的形式呈现。这些元组带有标签，因此此处采用监督机器学习。一些分类器利用收集到的信息构建决策树。为避免过拟合（即过度学习训练数据，导致在其他数据集上的表现不佳），通常会对树进行剪枝。例如， [23], 介绍了不同的剪枝技术，并分析了它们的复杂度。需要注意的是，在我们的研究中，我们不考虑用于数据分类的机器学习方法，而是分析该问题的非随机、非近似 [22, 20] 的时间复杂度，随后为其开发启发式方法。更重要的是，我们不仅试图预测类别标签（在我们的研究中，类别标签对应于某个目标的适应度），还要处理被视为敏感的属性（隐私特征），并在预测类别标签的同时，限制我们能够推断出的有关这些属性的信息量。

隐私学习。 该领域的研究探讨了隐私保护数据分类方法 [8, 9, 32]。这些方法会对用于训练分类器的数据集中的个体信息进行匿名化。许多数据集以数据元组的形式包含信息，例如（*David,* 西班牙*, 9, August, 2001*）。然而，仅从元组中删除标识符（在我们的例子中，只有姓名）这种简单做法，可能不足以保证数据集中个体的匿名性。一种常见的匿名化技术是k-匿名化，它确保数据集中的每个个体都至少无法与 k 个其他个体区分开来 [2]。与这些方法不同，我们的问题并非试图将数据集中的个体信息匿名化，因为：（a）我们假定数据集指的是人群（或具有典型特征的个体类型），而不是特定个体；（b）我们假定数据集表示的是通常为某一领域的专家所知，或可从公众调查中提取的普遍人口统计学知识——例如，它可能展示年龄、性别和财富方面的普遍差异如何影响消费习惯。由于该数据集是众所周知的，因此不能以任何方式对其进行修改、删减或混淆。相反，我们试图获取与被询问个体相关的信息（假定该个体具有任意特征组合的可能性反映在数据集中），而不推断该个体被视为隐私的信息。

[3]请注意，仍然可能出现较长的分支——例如，如果它们提出的问题与不应学习的特征大致正交。

随机应答技术和负向调查方法。 随机应答技术和负向调查方法都是构建调查的方法，旨在确保无法从受访者身上推断出任何隐私信息。随机应答技术要求问题的答案为二元答案，而负向调查方法要求问题的答案多于两个。

采用随机应答技术 [31] 的调查会询问受访者属于两个互斥且穷尽的群体中的哪一个（也就是说，他们目前是否居住在法国）。调查会通过随机化器从两个可能的选项中选出一个，且只有受访者会得知选中的问题。因此，受访者不会透露自己属于哪个群体，但根据随机化器的特性以及给出的是和否答案的比例，便足以估算总体中各群体所占的比例。或者，也可以采用限制较少的配对选项 [12] 变体，将敏感问题与无关问题配对。

另一方面，负向调查方法 [5] 采用这样的调查：询问受访者不属于哪个类别。例如，不询问受访者在即将举行的选举中打算投票支持哪个政党，而是询问他们不会投票支持哪个政党。与随机应答技术类似，这些方法用于获取有关总体人群的信息，同时不泄露有关受访者的信息。

需要着重指出，随机应答技术和负向调查方法旨在获取有关总体的信息，而非有关个体受访者的信息。因此，这些方法适用于开展调查，而非访谈，与我们的方法有本质区别。在我们的问题中，必须收集每位个体受访者的具体信息。

2 问题定义

本节将介绍我们所研究的问题。首先，为了说明该问题的主要方面，我们给出一个简单的非正式示例。随后，我们将给出问题的正式定义。

2.1 问题实例的非正式示例

回到某公司人力资源经理的例子，假设该经理拥有一个数据集，其中包含该公司数百名员工的五项特征信息。这些特征包括国籍（本地/外国）、编程语言技能（高/低）、工作经验（是/否）、教育水平（硕士学位/学士学位/无）和适应度（健康/不健康）。例如，适应度可以用于衡量员工在公司的工作表现。

表1：带有隐私特征的数据分类问题示例

	国籍	编程	经验	教育	健康	数量
1	本地	High	Yes	硕士学位	Fit	100
2	本地	Low	Yes	硕士学位	不合格	100
3	本地	Low	No	硕士学位	Fit	100
4	本地	Low	No	学士学位	不合格	100
5	本地	Low	No	None	不合格	50
6	外国	High	Yes	学士学位	Fit	100
7	外国	Low	Yes	学士学位	不合格	100
8	外国	High	No	硕士学位	Fit	100
9	外国	High	No	学士学位	不合格	100
10	外国	High	No	None	不合格	50

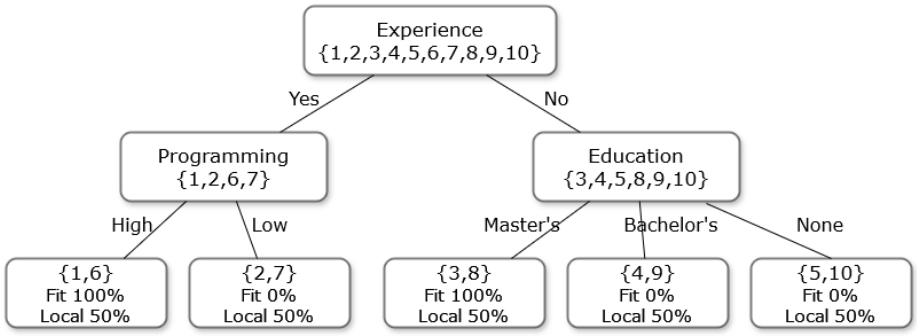


图1：交互式访谈示例

经理必须确定候选人是否会有良好的表现，而不推断其国籍，因为国籍会被视为隐私信息。经理将利用数据集中的信息设计一场交互式访谈，在不推断隐私信息的前提下，尽可能准确地预测受访者未来的表现。可用数据仅为示例而编制，见表1。

“数量 ”列表示按照每行所述方式回答的员工人数。请注意，并非所有可能的特征组合都已明确列出；不对应任何员工的组合不会存储在数据集中。

访谈，即我们要寻找的问题解，是一棵树。内部节点代表问题，从这些节点向下延伸的弧分别代表对应问题的每一种可能答案。叶节点代表访谈的最终状态。

图1中的树是人力资源经理可以设计的一种交互式访谈示例。访谈者会先询问受访者的经验。如果受访者回答有经验，访谈者就会询问其编程水平。如果受访者没有经验，经理就会询问其教育水平。我们可以看到，无论受访者对第一个和第二个问题作何回答，公司的所有员工只要给出相同的回答，其适应度都相同

，而且其中恰好有一半是本地员工（这与整个数据集的情况相同）。假设潜在受访者总体与员工总体相似，我们便可以推断（在这种情况下，误差范围为零）任何给定受访者的适应度，而无需推断此处被视为私有的信息。

请注意，并非所有访谈都能产生同样有效的结果。在询问受访者的工作经验之前，先询问其编程水平，会推断出有关其国籍的隐私信息，因为在我们的例子中，本地人的编程水平低于外国人。同样，受访者表示自己有工作经验后，若询问其教育水平而非编程水平，就无法区分合格个体和不合格个体。这说明选择合适的访谈至关重要，也暗示了这一过程在计算上的难度。

我们将把隐私保护访谈的构建作为一个决策问题来研究。因此，该问题会存在一些实例，无法为其设计出满足既定约束的访谈。如果目标特征与隐私特征之间存在极强的相关性，那么在未充分了解后者的情况下，就不可能充分了解前者。因此，对于某些要求较高的阈值——即对前者要求达到的最低了解程度，以及对后者允许的最高了解程度——答案有时会是否。

这并不意味着高度相关的问题实例永远无法求解，尤其是在所需阈值要求不高时，因为在这些情况下，可以巧妙地利用两种特征之间的细微差异来满足阈值。例如，如果目标特征与隐私特征在 $7/8$的情况下相同（或者在 $7/8$ 的情况下恰好相反，这同样难以处理），那么，要想获得更多关于前者而非后者的信息，唯一的办法就在于利用答案不同的 $1/8$ 情况。假设向量 01110001 中的每一位表示：对于以特定方式回答问题x、y、z 的个体，目标特征t 是否存在：第一位对应否、否、否；第二位对应否、否、是；依此类推；第八位对应是、是、是。同样，假设 11110001 是隐私特征f对应的比特向量。可以看出，特征t 与f 在 $7/8$ 的情况下相同，因此这是一个高度相关的情形。我们假设，对问题x、y、z的八种可能回答方式分别作出回答的个体比例均匀分布（也就是说，$1/8$ 的候选人分别以这 8 种可能方式回答问题）。

为了避免推断出超出实例关于f的约束所允许的信息，提出问题x 是不正确的，因为如果受访者回答否，我们就会知道他具有隐私特征f。在这种情况下，提出问题y会更明智：回答是 y 意味着具有f的概率为75%，而回答否 则意味着概率为50%。请注意，这些概率与

具有f的先验概率相差不大，该先验概率为62.5%（$5/8$）。不过，问题y能提供有关特征t的有价值的额外信息：回答是 意味着具有t的概率为75%，而回答否则意味着概率为25%。事实上，这些概率与具有t 的先验概率（50%）之间的偏差，比提出同一个问题y后基于对f 的先验认知所产生的偏差更大。因此，提出问题y 能让我们获得更多关于目标特征t的信息，而不是关于隐私特征f的信息——尽管该问题实例中的两者高度相关。

2.2 带有隐私特征的数据分类问题的形式化定义

在问题定义中，我们将有限集合答案 称为受访者可能对问题给出的所有答案的集合。例如，如果候选人对问题只能回答是 或否 ，那么答案 $=\{$是，否$\}$。如果问题的答案各不相同，那么集合答案 就是所有可能问题答案的并集。

类似地，问题 是访谈中可能提出的所有问题构成的有限集合。需要注意的是，可提出的问题与候选人的特征是类似的概念；每个问题都明确询问一个特定特征，而且每个特征都有一个与之对应、明确询问该特征的问题。例如，如果某个特征是年龄，那么就恰好有一个问题直接询问年龄。下文中，我们将使用特征名称（年龄），而不是疑问句（你多大了？）。

最后，我们需要说明如何表示候选人。我们不必逐一界定与其他候选人等价的候选人。候选人类型 是一组对所有问题都以相同方式作答的候选人。候选人类型是全函数问题 $\rightarrow$答案。每个函数都可以看作一种回答问题的方式。

例如，如果类型为 c 的候选人是 30 岁的女性，那么$c(age) = 30$和 $f(gender) = female$。我们不单独定义每位候选人的适应度，而是定义每种候选人类型的适应度以及每种类型的候选人数。如果某位候选人按照候选人类型所规定的方式回答问题，我们就说该候选人由该候选人类型表示。

我们假设，受访者能够代表数据集中候选人的概率分布，也就是说，遇到某个特定候选人类型的受访者的似然，与数据集中该类型个体所占的比例相同。请注意，我们的目标并非不断提问，直至完全确定候选人类型，而是以这样的方式提问：无论受访者属于哪种候选人类型，对某个特征的不确定性都能大幅降低，而对隐私特征的不确定性则不会。因此，问题通常会在受访者的候选人类型尚未完全确定时停止（即仍有几种候选人类型是可能的）。例如，当我们说受访者有 50% 的可能性是

健康，或具有任何其他特征时，我们的意思是：在那些截至该时点会以与受访者相同的方式回答已提出问题的候选人中，有 50% 具有该特征。受访者与数据集中的情况一致这一必要假设意味着，数据集足够完整，因此受访者总能与其中某种候选人类型相匹配（即存在一种能够代表受访者的候选人类型）。[4]

定义 1 ***(CDPC)*** 具有隐私特征的数据分类决策问题（***CDPC***）的一个实例是一个元组 (f, g, P, a, b, x, y)，其中，假设 $n \in \mathbb{N}$ 是问题（以及特征）的数量：

- f: (问题 $\to$ 答案) $\rightharpoonup \{\top, \bot\}$ 是一个用于评估候选人类型的部分适配函数。由于该问题是非穷尽的，不能代表任何候选人的函数c :问题 $\to$ 答案不属于 f的定义域。 f 的像是 $\top, \bot$，其中 $\top$ 表示候选人类型适配， $\bot$表示不适配。

- g : (问题 $\to$ 答案) $\rightharpoonup \mathbb{N}$ 是一个部分函数，给定候选人类型后，它返回能够以这种方式作答的候选人数。不能代表任何候选人的候选人类型不属于 g的定义域。

- $P \subseteq$问题是隐私特征的集合。

- $a, b \in \mathbb{Q}$，其中 $0 \leq a \leq b \leq 1$，限制能够推断出的隐私信息量。我们必须确保，对于每个隐私特征，在每次回答后，针对与该问题相关的某个特定答案作答的总体比例始终介于 a 和 b之间（为简化记号起见，我们将这一特定答案称为1 ，但它可以是任何特定答案，例如“是”）。例如，如果 $a = 0.2$ 且 $b = 0.7$ ，那么在访谈过程中，对于私人问题“您的年龄是否未满 *30* 岁？”，尚未被排除的候选人中回答“是”的比例，在所有可能情况下都必须保持在20 到70 之间。否则，我们将认为受访者的年龄隐私信息已被推断出来。我们将回答与某个隐私特征相关的问题时回答 *1* 的候选人称为具有该隐私特征的候选人。[5]

[4]由于缺乏有关极罕见情况的足够信息，在实际情形中，可能会遇到一种理论上不可能发生的情况：受访者不匹配任何候选人类型。这意味着需要处理一个根据底层知识（即数据集）被假定为概率为 0 的受访者，因此无法依据该知识提供任何信息。因此，在这种异常情况下，所构建的问卷树预计不会发挥任何作用。鉴于这种情况发生的概率微乎其微，它对树整体效用的影响也微乎其微。用技术术语来说，数据集中信息不足所导致的不确定性称为 $interpolation\ uncertainty$ [15]。[5]请注意，如果可能的答案有三个或更多，那么只为其中一个答案（答案 1）设定限制，就会间接限制其余答案——特别是，只有它们的概率 sum 受到答案 1 限制条件的约束。尽管这种受限格式简单且表达能力有限，但足以使该问题达到 NP-困难（实际上，仅二元答案

- $x, y \in \mathbb{Q}$，其中 $0 \leq x \leq y \leq 1$，表示我们对受访者适应度预测的确信程度。树的所有可能路径都必须将匹配候选人集合（即在树路径上的问题中会与受访者作出相同回答的候选人集合）缩小到一个子集，使其中适任候选人的比例低于 x 或高于 y。例如，如果 $x = 0.2$ 且 $y = 0.8$，那么在所有情况下，我们都需要将总体缩小到一个适应度足够低（适任候选人占 *0%* 至 *20%*）或足够高（适任候选人占 *80%* 至 *100%*）的子集。

显然，问题实例并不包含受访者是哪位候选人，因为树的目标是能够处理所规定的所有 可能候选人。

每当我们向受访者提出一个问题，本质上就是将与受访者相符的候选人集合 分成两部分：一部分候选人会给出与受访者相同的答案，另一部分则会给出其他任何可能的答案（除非所有剩余候选人对该问题的回答都相同）。尽管从访谈者的角度来看，受访者的每种答案都会将总体一分为二，但问题与答案构成的树并不是二叉树。每个问题可能有两个以上的答案，而解决该问题的交互式访谈必须能够应对所有选项。

我们不允许在访谈的同一路径中重复提问，因为在假设答案一致的情况下，重复提问不会进一步区分候选人，因此是多余的。因此，在最坏情况下，树的深度将等于问题实例中的问题数量。

如果存在一种交互式访谈（树），能够确保对于任何可能的受访者，我们都可以足够确定地预测其适应度（依据 x 和 y），且不会推断出超出允许范围的隐私信息（依据 a和 b），那么该**CDPC** 实例就是肯定的，即该实例的 **CDPC** 答案为 $\top$；否则，该实例就是否定的，即该实例的**CDPC** 答案为 $\bot$。

可以认为，**CDPC**定义的灵活性不足，无法表示实际情境。例如，我们的定义要求候选类型的适应度只能是 $\top$ 或 $\bot$（健康或不合格），但允许候选类型达到 70% 的健康程度或许会很有意义——这表示该类型中 70% 的个体是健康的，30% 的个体是不合格的。此外，限制可推断隐私信息量的上下界a 和 b 对所有隐私特征都是共用的。许多实际情境可能需要更加灵活的定义。之所以保留这样的定义，是因为针对问题最受限制的定义证明其困难性，会使分类更有意义；这是因为困难性（例如 NP困难性）会通过泛化而平凡地传播。因此，在证明某个问题的 NP困难性时，

将用于证明这一点）。定义 2 中给出了这一问题的推广形式，它将使我们能够更自然地定制私有极限的定义，也能够涵盖既非 *fully* 合格也非不合格的候选类型。

最好分析该问题最严格变体的复杂度，因为更一般的变体会立即继承更具体变体的复杂度。因此，分析**CDPC**的复杂度，比研究该问题其他更一般、更灵活的变体的复杂度更有意义。尽管如此，我们还定义了一个更适用于实际情境的问题变体。实际上，这个更现实的变体将用于后续的实验。

定义 2 ***(GCOPC)*** 通用隐私特征数据分类优化问题（***GCOPC***）的一个实例是一个元组 $(f_g, g, P_g k)$ ，它与 ***CDPC*** 的实例在适应度函数的值域、问题限制的添加方式以及隐私特征的定义方式上有所不同：

- f_g：（问题 $\rightarrow$ 答案） $\rightharpoonup \{q \mid q \in \mathbb{Q}, 0 \leq q \leq 1\}$ 是一个用于评估候选类型的部分适配函数。正如我们之前所述， f_g 允许候选类型具有部分适应度（例如，某一类型所代表的个体中有 *70%* 是健康的）。

- $k \in \mathbb{N}$ 是问题限制 k。它只是将访谈中的问题数量（即树的深度）限制为小于或等于 k。

- $P_g \subseteq$ 问题 $\times$ 答案 $\times \{a \mid a \in \mathbb{Q}, 0 \leq a \leq 1\} \times \{b \mid b \in \mathbb{Q},\ 0 \leq b \leq 1\}$是定义隐私特征的元组集合。如前所述，在实际情境中，所有隐私特征都使用相同的上下界 a 和 b 可能并不现实。*GCOPC* 中用元组集合 P_g 替换了 *CDPC* 实例参数 P、 a 和 b ；其中每个元组（r、p、a、b）表示一个问题 p （询问某项隐私特征）的隐私答案 r（即我们认为敏感的答案），以及一个区间 $[a, b]$ 。在整个访谈过程中，与受访者相匹配的候选人（即会以与受访者相同的方式回答问题的候选人）中，对问题 p 回答 r 的候选人比例应保持在该区间内。例如，如果 $P_g = \{$（age、< 30，*0.1*，*0.8*）、（age、> 60，*0.4*，*0.6*）、（$Spanish$、$C1$，*0.3*，*0.7*）$\}$，那么，年龄小于 *30* 岁、年龄大于 *60* 岁以及西班牙语水平为 *C1* 的候选人所占比例应分别保持在区间 $[0.1, 0.8]$、$[0.4, 0.6]$ 和 $[0.3, 0.7]$ 内。

此外，***GCOPC*** 与 ***CDPC*** 的不同之处在于，它不是像 ***CDPC*** 那样确定在所有情况下都足够高或足够低的适应度水平（由实例参数 x 和 y给出），并在适应度值超过这些阈值后忽略其实际值，而是一个优化问题。因此，它旨在尽可能降低适应度的不确定性（即在所有情况下都使适应度值尽可能接近0或1 ），同时保持所需的隐私水平。

访谈中的每条路径都代表一系列问题和答案。访谈的叶节点与初始种群的子集相关联，

这些子集是将总体逐步缩小后得到的，其中只包含那些会按照其所在树路径所指示的方式作答的候选人。每个叶节点都有一个适应度比，具体取决于其剩余人口中候选人的适应度。这个比值离 0.5 越远，我们就越能将该节点剩余人口中的适任候选人与不适任候选人区分开来。**GCOPC** 旨在找出一种交互式访谈，使其在最坏情况下（即适任候选人与不适任候选人比例最接近的访谈叶节点上）能够最大程度地区分适任候选人与不适任候选人。因此，在 **GCOPC** 中，返回的是交互式访谈本身，而不是像决策问题**CDPC**中那样返回 $\top$ 或 $\bot$ 。

3 复杂度分析

本节将非正式地介绍 NP 完全性证明的主要思想。对于希望了解详细形式化证明的读者，请参阅附录 A，其中正式证明了带有隐私特征的数据分类问题（**CDPC**）的 NP 完全性。

证明中最相关的部分，是从另一个 NP 完全问题（在我们的案例中为集合覆盖）定义一个适当的多项式归约。因此，我们将首先解释如何定义这种归约，然后通过一个示例对其进行说明。

3.1 从集合覆盖到 CDPC 的变换

集合覆盖判定问题 （**SC**）是一个经典的组合优化问题，在计算复杂性领域具有重要意义。它是 Karp 的 21 个 NP 完全问题之一 [13]。给定一个元组（U，S，k），其中全集 $U = \{u_1, ..., u_n\}$ 是一个元素集合，$S = \{S_1, ..., S_m\}$ 是 U 的子集构成的集合，其并集为全集，$k \in \mathbb{N}$且 $k \leq |S|$，集合覆盖判定问题要求确定是否存在一个基数为 k 或更小的 S 子集，使其覆盖整个全集。

附录 A 中通过从**集合覆盖**进行多项式归约，证明了**CDPC** 是 NP-困难的。我们需要定义一种多项式算法，将**集合覆盖**的实例变换为**CDPC**的实例。为了确保多项式归约正确，我们需要保证：对于**集合覆盖**的任意实例，该实例的**集合覆盖** 输出，与将原**集合覆盖**实例变换后得到的**CDPC**实例的输出相同。

如我们所见，**CDPC** 的一个实例是一个元组 (U，S，k)，其中 U 是一个由元素组成的集合、$S \subseteq \mathcal{P}(U)$ 和 $k \in \mathbb{N}$；而**CDPC** 的一个实例是一个元组 (f，g，P，a，b，x，y)。

我们将根据 k 在**集合覆盖**实例中的取值，将该变换分为两种情况。如果 $k = |S|$ （包括 $U = \emptyset$ 和$k = 0$的情况），那么该实例的结果显然是 $\top$ ，因为我们可以选取 S 中的所有集合以及 $\bigcup_{S_i \in S} S_i = U$。将这些**集合覆盖**实例变换为同样为正例的**CDPC** 实例并不困难。例如，我们可以使用以下实例

CDPC 的实例，定义见第2.1节。在本研究的其余部分，我们假设$k < |S|$。此外，我们还假设 S 不包含集合 $\emptyset$。如果包含，则只需从 S中移除 $\emptyset$，即可将其轻易变换为等价的**集合覆盖**实例。

为将集合覆盖判定问题的实例（其中 $k < |S|$）转换为其他形式，我们首先需要定义集合 问题和答案。集合 问题包含以下元素：

- S中的每个集合（因此 $S \subset$ 问题）。例如，如果 $\{1,2,3\} \in S$，那么 $\{1,2,3\}$ 就是一个可能的问题。
- 另一个问题，我们称之为私有。这个问题是必要的，用来表示**CDPC**实例中唯一需要的隐私特征。请注意，在任何解中，这个问题都不会成为访谈的一部分，因为提出这个问题会使得无法满足不得推断隐私信息的限制。尽管如此，这个问题仍然是必要的，因为根据我们的问题定义，每个特征都必须与一个问题相关联。

另一方面，只允许二元响应就足够了。我们定义答案 $= \{1,0\}$。

为了定义适应度函数和数量函数，即 f：（问题 $\to$ 答案）$\to \{\top, \bot\}$ 和 g：（问题 $\to$ 答案）$\to \mathbb{N}$，分别针对一个**集合覆盖**实例，我们需要定义以下候选人类型（用函数问题 $\to$ 答案表示），它们将作为函数 f 和g的输入：

- 每个元素 $e \in U$都有一种候选人类型。这些候选人会对与其所属集合相对应的问题回答 1，对其他问题回答 0。例如，与元素 3 对应的候选人 c_3 会对问题 $\{1,2,3\}$ ($c_3(\{1,2,3\}) = 1$ 因为 $3 \in \{1,2,3\}$)回答 1，对问题 $\{2\}$ ($c_3(\{2\}) = 0$ 因为 $3 \notin \{2\}$)回答 0，并对问题 私有($c_3(private) = 0$)回答 0。我们将属于这些类型的候选人称为元素。

- 每个集合 $c \in S$都有一种候选人类型。这些候选人会对与其自身集合相对应的问题回答 1，对与任何其他集合相对应的问题回答 0。例如，候选人 $\{1,2,3\}$ 会对问题 $\{1,2,3\}$ 回答 1，对问题 $\{2,4\}$回答 0。此外，他们会对问题私有回答 1。我们将属于这些类型的候选人称为集合，并且他们是唯一具有隐私特征的候选人。
- 另一种候选人类型，我们称之为空。这类候选人对所有问题的答案都是 0。

函数 f 将为空 候选人返回 $\top$，因为它们将是唯一健康的候选人。它将为元素 和集合 候选人返回 $\bot$，而对其他候选人则返回未定义。

元素 候选人的数量将远多于集合 候选人，而空 候选人的数量还会更多。函数 g 将为空 候选人返回 Ω^2，为元素 候选人返回 Ω，并为 set 候选人返回 1，其中 $\Omega \in \mathbb{N}$ 是一个足够高的值，可确保变换的正确性。具体而言，$\Omega = 2 * |U| * |S|$。

如前所述，唯一的隐私特征是私有。因此，$P = \{private\}$。

在初始种群中，具有隐私特征（即对私有回答 1）的候选人比例为 $|S|/(\Omega^2 + |U| * \Omega + |S|)$，因为有 $|S|$集合 个候选人对私有回答 1，有 Ω^2 空 个候选人，以及 $|U| * \Omega$ 个元素 候选人。定义在不推断出超出允许范围的更多敏感信息的前提下，对私有 回答 1 的候选人比例所能达到的最小值和最大值的界限 $a, b \in \mathbb{Q}$ 将为 $a = (|S| - k)/(|U| * \Omega + \Omega^2 + |S| - k)$ 和 $b = 1$。因此，可以接受与受访者匹配的所有候选人都回答 1。最后，定义最终群体所需适应度的界限 $x, y \in \mathbb{Q}$ 将为 $x = 0$ 和 $y = \Omega^2/(\Omega^2 + |S|)$。请记住，在所有情况下，我们都需要使适应个体比例足够低（低于 x）或足够高（高于 y）。

函数 f 和 g 有定义的输入值数量相对于输入大小（$|U| + |S| + 1$）是多项式的，并且我们可以在常数时间内计算这两个函数在任意这些输入值上的返回值。由于 P 只包含一个问题，因此可以在常数时间内构造。最后，只需遍历集合 U 和 S 中的元素，即可在线性时间内查询它们的基数，因此可以在多项式时间内计算 a、b、x和y 的值。由于元组（f、g、P、a、b、x、y）的所有分量都能在相对于（U、S、k）大小的多项式时间内计算，因此我们定义的变换是多项式时间的。

为清晰起见，接下来我们举例说明这一归约。提醒一下，形式化证明见附录 A。

3.2 到 CDPC 的变换示例

本节将通过一个示例说明该变换的工作方式。我们从以下集合覆盖判定问题的实例（U、S、k）开始:

```
U = {1, 2, 3, 4}
S = { {1,2,4}, {1,3}, {2} }
k = 2
```

请注意，这个决策问题在于确定是否存在一个基数小于或等于 k 的 S 子集，使得其元素的并集为 U。在我们的示例中，结果将是 $\top$，因为 $|\{\{1,2,4\},\{1,3\}\}| = 2$ 且 $\{1,2,4\} \cup \{1,3\} = \{1,2,3,4\}$。

表2展示了**CDPC**的结果实例。行表示候选人类型，列表示问题的答案，各数值

候选人	问题				健康	数量
	{1,2,4}	{1,3}	{2}	私有	f	g
1	1	1	0	0	$\perp$	Ω
2	1	0	1	0	$\perp$	Ω
3	0	1	0	0	$\perp$	Ω
4	1	0	0	0	$\perp$	Ω
null	0	0	0	0	$\top$	Ω^2
{1,2,4}	1	0	0	1	$\perp$	1
{1,3}	0	1	0	1	$\perp$	1
{2}	0	0	1	1	$\perp$	1

表2：**CDPC** 的结果实例

分别由适应度函数 f 和数量函数 g给出。

例如， $f(c_1) = \perp$ 和 $g(c_1) = \Omega$，其中 c_1 是与集合覆盖问题 →答案实例中的元素 1 对应的候选人类型，也就是返回 $c_1(\{1,2,4\}) = 1, c_1(\{1,3\}) = 1$, $c_1(\{2\}) = 0$，以及 $c_1(private)$ =0 的候选人类型。

剩下的就是定义 Ω、 a、 b、 x 和 y的值。$\Omega = 2*|U|*|S|$ =24, $a = (|S| - k)/($ $|U| * \Omega + \Omega^2 + |S| - k) = 0.0015$, $b = 1$, $x = 0$， 以及$y = \Omega^2/(\Omega^2 + |S|) = 0.9948$。

此变换旨在确保存在一个S 的子集满足集合覆盖判定问题的实例，当且仅当存在一个交互式访谈满足**CDPC**的实例。

设策略 是一种构造交互式访谈的方法，其中下一个问题的选择由受访者之前的答案决定。我们的策略非常简单：我们会提出与**集合覆盖**的解相对应的任意问题，即解中的集合。如果受访者回答 1，我们就立即结束访谈，并据此判断其不合格。如果回答 0，我们就继续提出下一个问题并重复这一过程，直到受访者回答 1，或者与**集合覆盖**的解相对应的问题全部问完；后一种情况下，我们据此判断其健康。

在我们的示例中，按照上述策略进行的访谈包含问题 {1,2,4} 和 {1,3}。提问顺序并不重要。接下来，我们要确认无论受访者是谁，都能以足够高的准确度推断其适应度（即使其概率落在（x, y）范围之外），同时推断出的隐私信息不超过允许范围（即使其回答隐私问题时回答 1 的概率保持在$[a, b]$ 范围内）。

如果受访者健康，那么他们只能属于空类型。问题 {1,2,4}排除了候选类型 1、2、4 和 {1,2,4}。随后提出的问题 {1,3}排除了类型 1（已被排除）、3 和 {1,3}。两个问题之后，尚未被排除的候选类型仅有空 和 {2}。目标类型（空）有 Ω^2 名候选人，而类型 {2}只有一名候选人。

因此，剩余人口的适应度比为 $\Omega^2/(\Omega^2+1) = 0.9983$，高于最低要求值（因为 $y = 0.9948$）。

另一方面，如果受访者不合格，即不是空，那么我们需要考虑两种可能情况:

- 如果受访者属于类型 $\{2\}$，那么他们对问题{1,2,4} 和 $\{1,3\}$的答案将是 0。由于只有候选类型 $\{2\}$ 和空能对这两个问题都回答 0，因此情况与上文类似。剩余总体的适应度比为 0。9983，高于y。在这种情况下，依靠该比值来预测候选人健康是不正确的，因为很遗憾，受访者并不合格。尽管如此，这次访谈仍然是正确的，因为我们已成功达到所需的适应度比，这一比值允许存在误差范围。

- 如果受访者不属于类型 $\{2\}$，那么他们对其中一个问题的答案将是 1，因为只有两种候选类型对问题 $\{1,2,4\}$ 和 $\{1,3\}$ 都回答 0：空 和 $\{2\}$。因此，我们排除了空类型的候选人，而这些候选人是唯一健康的候选人；剩余总体的适应度比将足够低（$0 \leq x = 0$）。

无论受访者是谁，我们都可以保证适应度比小于或等于 $x = 0$，或大于或等于 $y = 0.9948$。

为了得出变换对我们的示例有效这一结论，我们需要检查访谈不会推断出超出允许范围的隐私信息。具有隐私特征的初始种群（即对隐私问题回答 1 的人）的比例为 $|S|/(|U| * \Omega + \Omega^2 + |S|) = 0.0044$，我们需要确保该比例仍处于 $a = (|S| - k)/(|U| * \Omega + \Omega^2 + |S| - k) = 0.0015$ 和 $b = 1$ 之间。

首先，假设对问题 $\{1, 2, 4\}$的答案是 0。具有隐私特征的剩余人口（由类型 3、空、 $\{1,3\}$ 和 $\{2\}$组成）的比例为 $2/(\Omega^2 + \Omega + 2) = 0.0033$，仍处于区间 $[a, b]$内。关于下一个问题，即 $\{1, 3\}$,

- 如果答案为 0，那么具有隐私特征的剩余人口（即类型 空和 $\{2\}$）所占比例为 $1/(\Omega^2+1) = 0.017 \in [0.0015, 1]$，并且

- 如果问题 $\{1, 3\}$ 的答案为 1，那么具有隐私特征的剩余人口（类型 3 和 $\{1, 3\}$ ）所占比例为 $1/(\Omega + 1) = 0.04 \in [0.0015, 1]$。

另一方面，假设对第一个问题 $\{1, 2, 4\}$给出的答案为 1。具有隐私特征的剩余人口（类型 1、2、4、 $\{1,2,4\}$）所占比例为 $1/(3 * \Omega + 1)$ $= 0.0137 \in [0.0015, 1]$，我们将按照策略所示结束访谈。

可以用类似的方法证明，在 $\{1, 3\}$ 之前$\{1, 2, 4\}$ 提出该问题不会影响结果。由于证明内容重复，故省略。附录 A 在证明我们定义的归约正确的过程中表明，

提出问题的顺序并不重要。

如我们所见，遵循既定策略不可能推断出超出允许范围的隐私信息，并且我们可以确保，对于集合覆盖判定问题的这一特定实例，该转换是正确的。

通过这个示例，我们可以初步了解所提出的归约为何有效：

- 在大多数情况下，无法提出问题 隐私，因为这会直接推断出隐私信息。唯一的例外是，我们已经排除了所有对 隐私回答为 0 的候选人。附录A.1中介绍的引理1考虑到了这一特殊情况。

- 我们最多只能提出 k 个问题。否则，我们排除的集合 候选人数量将超过允许范围，因为每个问题恰好会排除 1 个此类候选人。如果我们排除的集合 候选人超过 k，那么具有隐私特征的剩余人口比例将过低（因为 $a = (|S| - k)/(|U| * \Omega + \Omega^2 + |S| - k)$）。[6]

- 当且仅当能够使用 k 集合问题排除所有元素候选人时，才能使用 k 个集合覆盖**SC**实例中 U 的所有元素。

- 如果要满足适应度要求（$y = \Omega^2/(\Omega^2 + |S|)$），就必须排除所有元素候选人。

3.3 分析结果

在对结果进行非正式介绍之后，我们总结复杂度分析，该分析在附录 A 中进行了完整阐述：我们定义了一个将集合覆盖判定问题**SC**的实例变换为**CDPC** 实例的算法，并证明该算法为多项式时间算法（见第3.1节）；我们证明，如果 **SC** 实例的结果为 $\top$，则其变换后实例的结果也将为 $\top$（附录A.1.1）；我们证明，如果**SC** 实例的结果为 $\bot$，则其变换后实例的结果将为 $\bot$（附录A.1.2）；最后，我们证明**CDPC** 属于NP（附录A.2）。[7] 因此，我们得出结论：所定义的多项式归约是正确的，并且**CDPC** 是NP完全问题。

既然已经证明 **CDPC** 是 NP困难的，因此其优化版本以及更一般的变体 **GCOPC** 也都是 NP困难的。如前所述，问题的困难性是从其更具体的变体

[6]请注意，只要受访者一直回答 0，因此 $null$ 尚未被排除，排除单个 set 候选人就总会降低具有隐私特征的总体占比（其初始值为 $|S|/(|U| * \Omega + \Omega^2 + |S|)$），无论是否同时排除了所有 $element$ 候选人。这是因为 Ω 的值经过选择，足够高，使得 $1/|S| > (|U| * \Omega + |S|)/(|U| * \Omega + \Omega^2 + |S|)$成立。[7]由于该问题的解是 $trees$，乍看之下，将其归入 NP 可能会令人惊讶。关键在于，这些树足够小，尤其是其大小为多项式级。

传递到更一般的变体中的。形式上，我们可以通过一种非常简单的多项式归约证明 GCOPC 的 NP困难性。回顾一下，CDPC 和 GCOPC 的实例分别是形如 (f, g, P, a, b, x, y) 和 (f_g, g, P_g, k, x, y) 的元组，其中 f:（问题 $\to$ 答案） $\rightharpoonup \{\top, \bot\}$, f_g:（问题 $\to$ 答案） $\rightharpoonup \{q \mid q \in \mathbb{Q}, 0 \leq q \leq 1\}$，以及 $P_g \subseteq$ 问题 $\times$答案 $\times \{a \mid a \in \mathbb{Q}, 0 \leq a \leq 1\} \times \{b \mid b \in \mathbb{Q}, 0 \leq b \leq 1\}$。

要在多项式时间内将**CDPC** 的实例转换为**GCOPC**的实例，首先，我们将适应度函数 f 的像中所有 $\top$ 和 $\bot$ 的出现分别替换为 1 和 0，以构造 f_g。请记住，尽管 f 的可能输出为 $\top$ （合格）和 $\bot$ （不合格），函数 f_g 返回的是介于 0 和 1 之间的有理数。然后，为了定义 P_g ，我们将每个隐私特征替换为一个元组，其中包含该特征、答案 1（如第2.2节所述，在**CDPC**中，我们需要确保在整个访谈过程中，对每个隐私特征回答 1 的总体比例保持不变）以及界限 a和 b。也就是说， $P_g = \{(p, 1, a, b) \mid p \in P\}$。例如，根据 $P = \{male, religious\}$ ，我们将构造 $P_g = \{(male, 1, a, b), (religious, 1, a, b)\}$。最后，问题限制 k将设为可用问题的总数，这相当于没有问题限制。

显然，相对于输入规模而言，这种变换可以在多项式时间内完成。显然，所得实例与原始实例等价，因此，这种从 **CDPC** 到 **GCOPC**的简单多项式归约是正确的，而且 **GCOPC**是 NP-困难的。

4 实际求解

带有私有特征的数据分类问题是 NP-困难的，因此无法通过精确方式求解大型实例。为了处理大型实例，必须采用启发式方法。这些方法无法保证找到完美的解，但能在合理的时间内提供足够好的解。

尽管 **CDPC**具有更高的特异性，因此在 NP-困难性分类方面更有研究价值，但 **GCOPC**更适合处理实际情境，因此实验中将采用 **GCOPC**。实际求解所用的代码可在在线代码仓库中获取 [21]。

在深入探讨算法之前，有必要明确“适应度”一词的用法。到目前为止，我们一直使用“适应度”来指代候选人或候选人群体的能力倾向。在讨论遗传算法和其他启发式算法时，“适应度”通常指问题实例的某个解的优良程度。为避免歧义，我们将使用 优良度 来指代一次访谈的优良程度。请回想一下，我们旨在找到一次不会推断出隐私信息、并能尽可能区分适任候选人与不适任候选人的访谈，从而使访谈树叶节点中候选人的适应度达到极值（即最大化或最小化）。

问题数量	运行时间（秒）
1	0.0009
2	0.049
3	5.8657
4	9.2263
5	70.0895
6	176.3103
7	241.6572
8	687.6727
9	1348.9556
10	1880.9390

表 3：精确算法的运行时间与问题数量的关系

4.1 精确算法

正如我们所讨论的，由于该问题的复杂度已得到证明，求解较大规模的问题实例是不可行的。为了说明这一点，我们使用 Python 实现了一种能够求解该问题实例的算法。这个简单的算法使用阿尔法-贝塔剪枝 [7] 来缩小需要评估的解空间。

为了测试算法在问题数量方面的可扩展性，我们针对 1 到 10 个问题的每种情况，在 5 个随机构造的**GCOPC**实例上运行该算法，并计算平均运行时间。这些随机构造的问题实例包含 500 种不同的候选人类型，每次访谈最多有 10 个问题（$k = 10$）。表 3 展示了结果。

这些简单的测试表明，该问题难以处理——照此速度，即使只有 30 个问题，也需要数年运行时间。作为参考，在以下章节中，我们将近似求解包含 3000 多种候选人类型和 200 多个问题的**GCOPC**实例。因此，使用精确方法处理此类实例完全不可能。

4.2 贪心算法

我们使用 Python 实现了一种简单的贪心算法，用于近似求解 **GCOPC** 的实例。该算法在构建交互式面试树的每个节点处，贪心地选择剩余问题中能够在不推断隐私信息的情况下最大化访谈优良度的问题。作出这一选择时，并未考虑到通常情况下，访谈（即树）将在后续节点中以相同方式继续增长。

具体而言，为了选择与树中每个节点关联的问题，我们假设访谈会以该问题结束。我们按照这些问题所能带来的访谈优良度对可能的问题排序，然后依次检查排序后的问题列表，直到找到一个不会推断出隐私信息的问题（如果存在的话）。如果我们

找到这样的问题，就在当前节点选择该问题，并继续执行算法，以相同的方式选择树中其余的问题。如果一个也找不到，我们就结束访谈的这条路径。为加快算法运行，我们会维护一个列表，记录当前正在探索的路径上尚未提出的问题。下面的伪代码说明了这一算法：

贪心算法（候选人、问题上限、问题）_如果没有剩余问题或已达到问题上限，则返回空。采访对象按每个问题后的群体适配度对问题排序。对于每个剩余问题，对于该问题的每个可能答案，如果答案无法推断出私人信息，则退出循环。如果每个问题都能推断出私人信息，则返回空采访

子代 = 空映射 对每个可能的答案，匹配候选者与问题 = 那些回答了问题的人
子代 [答案] = 贪心算法（匹配候选者，问题限制 −1，问题 − 问题）
-
-
-

返回 Interview(question,children)

4.3 遗传算法

遗传算法 [14, 1] 是一种受生物进化和遗传学启发的启发式优化算法，其核心是通过交叉和变异来进化一个由个体组成的总体，其中每个个体都编码了该问题的一个可能解。简而言之，遗传算法的基本操作如下：

1. 首先，创建初始种群。2. 将总体中的个体两两交叉，生成具有双方亲代特征的新个体。3. 每个个体都有较小的概率发生变异，从而轻微改变其某些特征。4. 根据个体的优良度，选出将进入下一次迭代总体的个体。5. 这一过程反复进行，返回第 2 步。

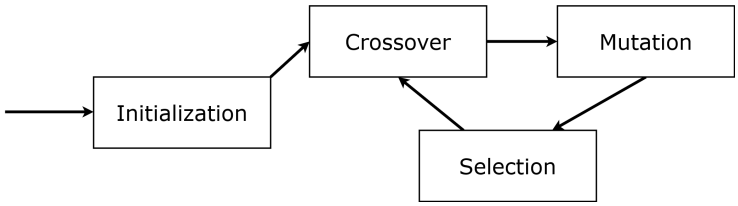


图2：基本遗传算法示意图

经过给定次数的迭代后，算法将终止，并返回由最佳个体编码的解。

在我们用 Python 实现遗传算法时，会将所研究的 **GCOPC** 实例的解（即交互式访谈）编码为树，其中节点的值是问题。总体中的任何访谈都不会推断出隐私信息。

使用遗传算法解决我们的数据分类问题并非易事。我们必须考虑到，通常，任意构造的大多数访谈都会在树的至少一条路径中推断出隐私信息。此外，访谈的效果取决于其最差路径。若某个访谈对大多数候选人都能取得良好结果，却会推断出少数候选人的隐私信息，那么这个访谈就是不正确的。当初始种群由不正确的访谈组成时，遗传算法最终难以生成正确的访谈。

因此，我们将确保初始化、交叉和变异方法不会生成不正确的访谈。如果在访谈中添加的问题会推断出隐私信息，我们就会尝试寻找其他问题。鉴于访谈的规模，这个过程可能会影响遗传算法的性能。我们将每次初始化或变异访谈时，寻找正确问题的尝试次数限制为 100 次。如果尝试次数用尽，或没有剩余问题，访谈的路径就会终止。

总体的初始化包括随机创建问题树和答案树，同时确保生成的访谈不会出错。

为了通过交叉两个个体生成新个体，我们会创建一棵新树，其根节点上的问题不会出现在亲代的访谈中。对于该问题的每个可能答案，生成的子树都是随机选取的其中一个亲代的子树。为确保生成的树正确无误，会剪除推断出隐私信息的路径。

在每次迭代中，对于每次访谈，变异函数会以一定概率，将访谈树中某个节点（位于随机深度）的子树替换为一棵随机子树，以确保访谈正确无误。

作为选择函数，我们采用锦标赛法。也就是说，我们随机将个体两两配对，并从每对中保留优良度最高的个体，用于算法的下一次迭代。

优度函数（遗传算法中个体的适应度函数）根据在最坏情况下正确预测候选人适应度的概率，为访谈赋值。为此，它会在树中适任候选人与不适任候选人比例最接近的叶节点，返回适任候选人比例与不适任候选人比例中较高的一个。

除了这种基本遗传算法外，我们还实现了一种由前文介绍的贪心算法增强的遗传算法（见第4.2节）。该算法与前一种算法的不同之处在于：初始化时，一份访谈由贪心算法直接生成，另一份则是对贪心解进行变异得到的。总体中的其余部分采用与基本遗传算法相同的方式初始化。此外，一半的变异操作会用从该位置开始应用贪心算法所得的解替换访谈的一棵子树，而不是替换为随机生成的子树。

4.3.1 参数调优

本节将说明我们如何调整遗传算法的参数。为此，我们生成了 GCOPC 的随机实例。这些构造的实例中，每个问题最多有5个答案，每种候选人类型有100名候选人，且隐私特征较少（具体而言，为4至9个）；这也是实际访谈中最常见的情形。此外，具有许多隐私特征的实例不太适合用于比较，因为这会严格限制访谈的规模，以免推断出隐私特征。

我们在 10 个随机生成的不同 **GCOPC** 实例上运行基本遗传算法，以确定种群规模。我们采用 0.1 的变异率，并运行遗传算法 200 次迭代。表4列出了这些实例的规模，表5展示了所得结果。结果表明，使用 20 个个体明显优于仅采用 10 个元素的种群规模。然而，将种群规模增至 30 或 40 个个体并未提升结果质量。事实上，我们采用的统计检验表明，10 个与 20 个个体之间确实存在统计学上的显著差异，但 20 个与 30 个或 40 个个体之间则不存在差异。因此，在后续实验中，我们决定采用 20 的种群规模。

同样，为了确定迭代次数，我们对 10 个随机实例进行测试，采用 0.1 的变异率和 20 的种群规模，并定期测量得到的最佳解。表6列出了这些实例的规模，图3展示了算法性能随迭代次数的变化情况。可以看出，随着迭代次数增加，改进幅度显著减小；迭代 400 次后，继续迭代已不会带来实质性差异。因此，我们将遗传算法的最大迭代次数设为 400。

最后，为了选择遗传算法的变异率，我们运行了 10 个随机的 **GCOPC**实例，种群规模为 20 个个体，迭代 400 次，变异率介于 0 到 0.5 之间。表7 显示了各个规模

实例	候选类型	字符	私有字符	问题限制
1	3097	225	7	10
2	3303	224	6	11
3	2165	227	8	12
4	2254	180	6	13
5	2602	195	6	14
6	3536	152	9	12
7	3954	239	6	13
8	2158	235	8	12
9	3295	235	9	14
10	3963	262	6	15

表4：用于调整总体规模的实例概览

实例	大小 10	大小 20	大小 30	大小 40
1	0.8463	0.8827	0.8669	0.9102
2	0.9380	0.9050	0.9079	0.9334
3	0.8680	0.8910	0.9051	0.8351
4	0.9218	0.9845	0.9555	0.9284
5	0.8851	0.9690	0.9317	0.9534
6	0.7806	0.8209	0.9669	0.8783
7	0.8628	0.8660	0.8138	0.9023
8	0.8582	0.9021	0.8965	0.8889
9	0.8043	0.8714	0.9507	0.8767
10	0.8268	0.8716	0.8394	0.9071
平均值	0.8592	0.8964	0.9035	0.9014

表5：遗传算法的性能与总体规模的关系

实例	候选类型	字符	私有字符	问题限制
1	2823	168	9	14
2	3962	251	4	14
3	3199	298	6	14
4	2275	180	7	15
5	3702	261	9	11
6	3290	181	4	14
7	3552	259	8	15
8	2684	243	8	15
9	3417	210	4	11
10	3492	200	5	11

表6：用于调整迭代次数的实例概览

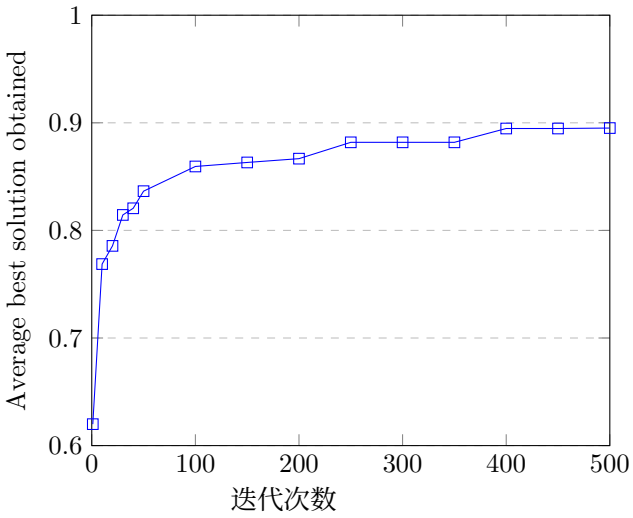


图3：遗传算法的性能随迭代次数的变化

这些实例的结果如图4所示，图中展示了各次运行所获得的结果。可以看出，变异率为 0.2 时取得的结果最佳。不过，不同的变异率似乎对算法性能的影响很小。遗传算法中个体变异的主要目的是促进对解空间的探索。由于具有随机性，交叉算法本身已能实现这一目的。此外，访谈的质量取决于其最差路径；而且，由于访谈树的规模较大，访谈中的最差路径很可能在变异后保持不变。因此，图4所示的结果没有呈现明显趋势，但变异率为 0.2 时取得的结果最佳。

实例	候选类型	字符	私有字符	问题限制
1	2470	181	9	15
2	2251	217	4	15
3	3999	301	8	11
4	2287	279	4	15
5	2852	301	6	15
6	2927	187	4	11
7	2341	252	5	12
8	2796	270	6	14
9	2838	174	9	13
10	2434	153	9	13

表7：用于调节变异率的实例概览

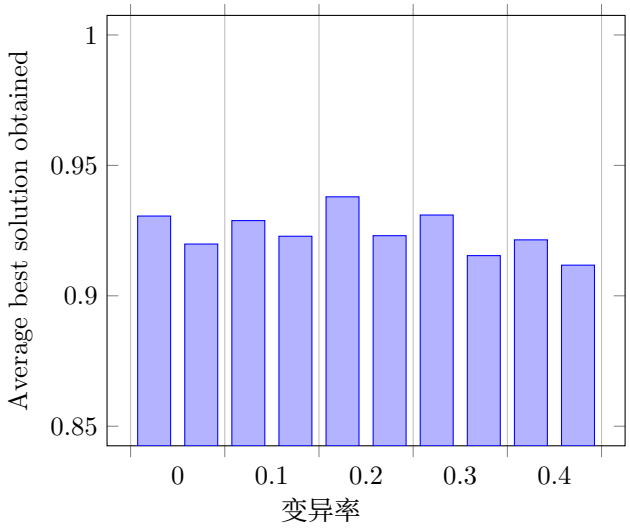


图4：变异率下的平均性能

根据所得结果，我们决定在后续的实际求解中采用以下参数：迭代次数为 400，种群规模为 20，变异率为 0.2。

4.4 算法性能比较

本节旨在展示并比较所实现的启发式算法的性能。我们使用遗传算法中选定的参数，针对 50 个随机构造的 **GCOPC** 实例，测试了贪心算法、基本遗传算法和增强遗传算法的性能——请注意，后者是在初始种群中加入贪心解及其变异版本的遗传算法。对于每个实例，我们分别运行每种遗传算法 3 次，并保留所得的最佳结果。表8和表9分别展示了实例的规模和所得结果。

值得注意的是，贪心算法的性能明显差于遗传算法。请注意，对于该问题的任何给定实例，即使是最差的正确访谈，其优良度也至少为 0.5。这种情况只会出现在恰好有 50% 的候选人适任，且在最坏情况下访谈无法以任何方式区分适任候选人与不适任候选人的实例中（访谈树中存在一个叶节点，其中 50% 的候选人适任）。此外，有些实例无法在不推断隐私信息的情况下完全区分适任和不适任个体，因此不存在优良度为 1 的访谈。因此，平均优良度从 0.5501 提高到 0.9453 或 0.9577，实际意义远比乍看之下更显著。然而，基本遗传算法与增强遗传算法之间的差异并不那么明显。事实上，一项统计

实例	候选类型	字符	私有字符	问题限制
1	2074	295	4	15
2	2748	279	7	15
3	2247	240	6	14
4	2144	184	7	11
5	3248	267	6	11
6	2084	157	7	10
7	2058	284	7	15
8	2722	189	8	12
9	3614	281	7	14
10	2349	257	6	12
11	3096	191	5	10
12	2831	283	7	10
13	2086	221	6	14
14	3852	159	9	13
15	2622	229	6	14
16	2164	199	8	10
17	2089	274	8	11
18	2406	152	9	11
19	2916	254	9	14
20	2839	184	9	12
21	2551	231	6	14
22	2272	161	4	12
23	3808	240	8	13
24	3208	282	5	11
25	3313	230	7	11
26	3487	178	4	15
27	2056	241	7	14
28	2608	208	9	10
29	3559	166	6	15
30	2406	281	4	14
31	2169	156	4	14
32	3257	221	7	12
33	3065	223	8	13
34	3025	301	5	10
35	2593	235	8	15
36	3794	298	9	12
37	2471	152	4	15
38	3714	281	9	15
39	2924	193	6	10
40	2596	286	5	15
41	2841	214	9	14
42	3601	211	9	11
43	2630	278	8	13
44	2204	178	4	14
45	2851	290	9	14
46	3218	221	8	13
47	2076	302	9	13
48	2337	231	8	13
49	3709	172	6	14
50	2936	289	6	15

表 8：用于比较启发式算法的实例概览

实例	贪心算法	基础遗传算法	强化遗传算法
1	0.7285	1	1
2	0.522	0.9573	1
3	0.7145	1	1
4	0.5098	1	1
5	0.6053	0.9637	1
6	0.6368	1	1
7	0.5015	0.9396	0.9562
8	0.5314	0.912	1
9	0.5257	0.9137	0.9188
10	0.5163	0.9632	0.9266
11	0.5471	0.9455	1
12	0.5147	0.9343	0.9537
13	0.5724	1	0.9917
14	0.5009	0.8708	0.9462
15	0.609	1	1
16	0.5152	0.8532	0.848
17	0.5095	0.9347	0.9434
18	0.623	0.9495	0.9021
19	0.5127	0.9105	0.9095
20	0.5071	0.9613	0.9039
21	0.6304	0.9456	0.9362
22	0.6152	1	1
23	0.5534	0.9096	1
24	0.6173	0.9074	0.9416
25	0.586	0.9313	1
26	0.5014	1	0.993
27	0.6054	0.9361	1
28	0.5344	0.9731	1
29	0.5403	0.9182	1
30	0.5066	0.9432	0.9212
31	0.503	0.99	0.9684
32	0.5011	1	1
33	0.5417	0.9649	0.9067
34	0.5151	0.8824	0.9184
35	0.5091	0.9223	0.9255
36	0.5112	0.9012	0.8874
37	0.5088	0.906	0.9255
38	0.5008	0.8821	0.9114
39	0.5017	0.9716	0.8871
40	0.5089	0.9403	0.9895
41	0.6156	0.9695	1
42	0.5025	0.9445	0.8876
43	0.5602	0.9057	0.8735
44	0.5107	1	1
45	0.5089	0.9925	0.9632
46	0.5726	0.8957	0.9465
47	0.5162	0.9308	0.9416
48	0.5075	0.8919	0.9606
49	0.5554	0.9289	1
50	0.6585	0.9721	1
平均适应度	0.5501	0.9453	0.9577

表 9：贪心算法、基础遗传算法和强化遗传算法的性能比较

二项符号检验（$\alpha = 0.1$）		
统计量	**p 值**	**结果**
26	0.088	H_0 被拒绝

表10：二项符号检验的结果

需要进行测试，以评估两种方法之间是否存在统计学显著差异。

为此，我们使用 STAC 工具 [26]比较了获得的结果。STAC 是一个网络平台和 Python 库，可通过统计检验比较算法。我们使用二项符号检验 [3]比较了遗传算法。该非参数双组假设检验（不假设数据遵循任何特定分布）比较配对观测值；在我们的案例中，这些配对观测值是遗传算法针对每个实例找到的最佳访谈的优良度。该检验不假设观测值之间的差异相对于中位数呈对称分布。我们使用二项符号检验来检测遗传算法性能是否存在显著差异。

表10显示了获得的结果。统计量 是根据算法结果计算得出的度量，用于判断零假设 H_0 （即比较中不存在统计差异）是否合理。显著性水平（α）是在 H_0 为真时拒绝它的概率，也就是出现假阳性的概率。值 p 表示在假设 H_0 为真的情况下，观察到算法所得结果的概率。当该值小于 α时，检验被认为具有统计显著性，并且H_0 会被拒绝。

最后，为了确保增强遗传算法能够为精确算法可以求解的小规模问题实例返回正确结果，我们将两者的结果与 10 个随机生成的实例进行了比较，每个实例包含 500 种候选类型和 10 个问题，结果完全相同。

通过这些测试，我们表明，增强遗传算法的表现分别显著优于基本遗传算法和贪心算法。当遗传算法无法找到改进其总体访谈的方法时，贪心变异以及使用贪心子树补全解有助于引导遗传算法。总而言之，我们可以得出结论：增强遗传算法明显优于另外两种方法。

5 结论与未来工作

本文正式介绍了带有隐私特征的数据分类问题及其优化变体。我们分析了该问题的计算复杂性，并为此构造了从 NP完全集合覆盖判定问题出发的多项式归约。除了所提出问题本身的分类意义之外，这还可以

帮助确定未来其他（明显或隐约）相关问题的复杂度。

由于该问题的计算复杂性，无法在合理时间内对大型实例求得最优解，因此我们实现了能够在合理时间内寻找次优解的启发式算法。我们实现了贪心算法、遗传算法，以及由贪心算法增强的遗传算法，并将它们的性能与一个具体基准进行了比较。我们发现，由贪心算法增强的遗传算法表现优于其他算法，且差异具有统计显著性。

我们证明该问题是NP完全问题，这表明不存在简单的策略能够确保在隐私保护与分类准确率之间实现最佳权衡。此外，我们的实验结果证实，采用简单策略（例如我们的贪心算法）所得结果相当不理想。因此，解决之道是寻找能够适应具体情形的启发式方法，并针对不同情况利用各自的特征。在此背景下，遗传算法是非常理想的折中选择，因为它们能提供一种通用方法，并适应每个具体案例。因此，我们建议，只要遇到需要在隐私保护与分类准确率之间保持平衡的问题，就直接采用我们的遗传算法。

本文提出的问题，即带有私有特征的数据分类问题，在当前背景下具有重要意义，因为它有着广泛的应用。它有助于在多个领域开发更高效、更合乎伦理的解，例如设计授予机会的访谈（如决定是否授予求职、入学、福利或收养机会的访谈）、用户数据收集、保障隐私的认证流程等。

值得注意的是，在我们的问题背景下，可以考虑需要保护的两类不同数据：（a）关于总体人群的统计信息；以及（b）关于特定受访者本人的信息。我们的问题只涉及保护（b），因为（a）被假定为常识。我们回顾一下，总体人群统计数据集表示具有每种特定特征组合的人所占的比例（例如具有特征 A、B 和 C，但不具有 D、E 和 F），其中既包括符合我们希望了解的目标属性的人所占的比例，也包括符合我们不希望了解的“禁止”属性的人所占的比例。如前所述，这些数据（或至少其总体趋势）要么已为相关领域的专家充分了解，要么可通过民意调查公开获取。例如，教育水平与工作表现之间的统计关系在相关文献中已有记载；某些宗教信徒通常遵守的饮食限制是任何受过教育的人都熟知的；营销专家以及任何希望查阅相关文献的人，都知道不同性别、年龄和财富水平人群在消费习惯方面的总体差异；等等。我们无法对这些数据采取任何数据隐私保护措施，因为它们属于常识，通常也不属于开发自适应面试的机构。该机构可以添加一些关于其自身员工或

以往申请人的额外信息，但只要总体人群中的趋势通常适用于该机构预期的申请人，那么导致目标属性或禁止属性潜在不确定性降低的相关性，就都是众所周知的。也就是说，不能为了隐私保护而更改统计数据集，因为实际上几乎所有人都已经知道它，或者只需进行一次互联网搜索就能了解到。

关于（b），即对受访者的隐私保护，这正是该问题以及本文所提出方法的目标：在大幅降低与访谈正当目标相关特征（例如申请人的工作能力倾向）不确定性的同时，尽可能降低被视为私有的特征（例如性取向）的不确定性。将设计好的自适应面试应用于申请人后，由于其构造方式，申请人所提供的答案导致的私有特征部分泄露将始终低于既定阈值。只要该阈值在构建访谈之前就已设定为期望值，那么在访谈结束后，无需对申请人提供的答案进行任何特殊处理：根据定义，这些答案符合既定阈值。因此，无需进一步进行后处理即可满足所期望的隐私水平。

本研究未来可能的工作范围十分广泛。应定义并分析更多问题变体，以便更好地适应其他可能的实际应用（例如，考虑访谈不是自适应的特殊情况）。

未来研究的另一个可能方向是开发并扩展实际求解方法。可以实现更多启发式算法，例如粒子群优化算法 [30], ，来解决这一问题。在这方面，探索基于图的多维约束机器学习方法的应用也会很有意思（参见例如 [28]）。需要注意的是，虽然利用这种基于图的方法为我们的问题提供一种通用求解方法可能较为复杂，但我们问题中的某些特定实例或许能够从这种方法中受益。

可以实现更多启发式算法，例如粒子群优化算法，来解决这一问题。此外，我们可以用更多经过有意构造的实例来测试已实现的算法。实例的难度并非仅取决于其规模；在许多问题中，例如 3-SAT，实例的难度会受到其参数特定模式的影响，因此，研究我们问题的实例中候选人与问题之间的关系对启发式算法性能的影响，或许会很有意义。

代码可用性： 算法、数据和数据生成代码可在 https://github.com/davidpantojasanchez/Data-classification-problem-with-private-characteristics.git获取。

作者贡献：David Pantoja： 撰写——审阅与编辑、撰写——初稿、可视化、验证、软件、资源、方法学、调查、形式分析、数据整理、概念化。**Ismael**

Rodrıguez：撰写——审阅与编辑、撰写——初稿、验证、监督、资源、方法学、调查、形式分析、概念化。**Fernando Rubio：**撰写——初稿、验证、监督、资源、项目管理、方法学、调查、经费获取、形式分析、概念化、撰写——审阅与编辑。**Clara Segura：**撰写——审阅与编辑、撰写——初稿、验证、监督、资源、方法学、调查、形式分析、概念化。

利益冲突

作者声明，他们不存在利益冲突。

参考文献

[1] Alhijawi, B., Awajan, A.：遗传算法：理论、遗传算子、解与应用。《Evolutionary Intelligence》**17**(3), 1245–1256 (2024)[2] Caballero, R., Sen, S., Nyg ̊ard, J.F.：预见筛查项目数据库中的匿名性。International Journal of Medical Informatics **103**, 20– 31 (2017)[3] Conover, W.J.：《实用非参数统计学》，第350卷。John Wiley & Sons (1999)[4] Dattner, B., Chamorro-Premuzic, T., Buchband, R., Schettler, L.：在招聘中使用人工智能的法律与伦理影响。《哈佛商业评论》**25**, 1–7 (2019)[5] Esponda, F.：负向调查。arXiv预印本 math/0608176 (2006)[6] Estrada-Jim´enez, J., Parra-Arnau, J., Rodr´ıguez-Hoyos, A., Forn´e, J.：在线广告：隐私威胁与保护方法分析。Computer Communications**100**, 32–51 (2017)[7] Felstiner, C.：阿尔法-贝塔剪枝。惠特曼学院 (2019)[8] Fletcher, S., Islam, M.Z.：差分隐私下的决策树分类：一项调查。ACM Comput. Surv. **52**(4) (2019). DOI 10.1145/3337064. URL https://doi.org/10.1145/3337064[9] Fung, B.C., Wang, K., Yu, P.S.：用于隐私保护的分类数据匿名化。IEEE Transactions on Knowledge and Data Engineering**19**(5), 711–725 (2007). DOI 10.1109/TKDE.2007.1015

[10] Galiana, J., Rodr´ıguez, I., Rubio, F.：利用双层遗传算法阻止不必要的传播。《应用软计算》**136**, 110094 (2023)[11] Godoy, A., Rodr´ıguez, I., Rubio, F.：根据政治立场进行投票并非易事：问题定义、计算困难性与近似解。《计算科学杂志》**80**, 102328 (2024)[12] Greenberg, B.G., Abul-Ela, A.L.A., Simmons, W.R., Horvitz, D.G.：无关问题随机应答模型：理论框架。《美国统计学会会刊》**64**(326), 520–539 (1969)[13]Karp, R.M.：组合问题之间的可归约性，第85–103页。施普林格美国，波士顿，马萨诸塞州 (1972)[14] Katoch, S., Chauhan, S.S., Kumar, V.：遗传算法综述：过去、现在与未来。Multimedia tools and applications**80**, 8091–8126 (2021)[15] Kennedy, M.C., O'Hagan, A.：计算机模型的贝叶斯校准。皇家统计学会会刊B辑：统计方法学**63**(3), 425–464 (2002). DOI 10.1111/1467-9868.00294. URL https://doi. org/10.1111/1467-9868.00294[16] Kodiyan, A.A.：使用人工智能系统进行招聘的伦理问题概述：以亚马逊的基于人工智能的招聘工具为例。ResearchGate预印本，第1–19页 (2019)[17] Kroll, E., Ziegler, M.：因族裔和性别造成的歧视：基于视频的求职面试容易受到多大影响？International Journal of Selection and Assessment**24**(2), 161–171 (2016)[18] Laurent, H., Rivest, R.L.：构造最优二叉决策树是NP完全问题。《信息处理快报》**5**(1), 15–17 (1976)[19] Macan, T., Merritt, S.：行动也说明问题：揭示就业面试过程中可能存在的隐性与显性歧视。《工业与组织心理学国际评论》201**126** (2011)[20] Mu˜noz, A., Rubio, F.：通过可近似性层级评估遗传算法。《计算科学杂志》**53**, 101388 (2021)[21]Pantoja, D.：带有私有特征的数据分类问题的复杂度分析与实际求解 (2024)。URL https://github.com/davidpantojasanchez/Data-classification-problem-with-private-characteristics.git[22] Paschos, V.T.：NP-困难问题多项式近似概述。《南斯拉夫运筹学杂志》**19**(1) (2016)

[23] Patil, D.D., Wadhai, V., Gokhale, J.：决策树剪枝算法在复杂度和分类准确率方面的评估。《国际计算机应用杂志》**11**(2), 23–30 (2010)[24] Purkiss, S.L.S., Perrew´e, P.L., Gillespie, T.L., Mayes, B.T., Ferris, G.R.：就业面试判断与决策中偏差的隐性来源。《组织行为与人类决策过程》**101**(2), 152–167 (2006)[25] Rodr´ıguez, I., Rubio, D., Rubio, F.：外延定义情境下自适应测试的复杂度。《计算机科学前沿》**17**(3), 173206 (2023)[26] Rodr´ıguez-Fdez, I., Canosa, A., Mucientes, M., Bugar´ın, A.：STAC：一个用于通过统计检验比较算法的网络平台。载于：2015年IEEE国际模糊系统会议（FUZZ-IEEE），第1–8页。IEEE (2015)[27] Saravanan, R., Sujatha, P.：机器学习算法的先进技术：监督学习方法在数据分类中的应用视角。载于：2018年第二届智能计算与控制系统国际会议（ICICCS），第945–949页 (2018)。DOI 10.1109/ICCONS.2018.8663155[28] Tutsoy, O.：基于图论的大规模机器学习，采用多维约束优化方法对大流行病进行精确的流行病学建模。《IEEE模式分析与机器智能汇刊》**45**(8), 9836–9845 (2023)[29] Ullah, I., Boreli, R., Kanhere, S.S.：定向广告中的隐私：综述。arXiv预印本 arXiv:2009.06861 (2020)[30] Wang, D., Tan, D., Liu, L.：粒子群优化算法：综述。《软计算》**22**(2), 387–408 (2018)[31]Warner, S.L.：随机应答：一种消除回避回答偏差的调查技术。《美国统计学会会刊》**60**(309), 63–69 (1965).[32] Zorarpacı, E., Ozel, S.A.：针对差分隐私数据的隐私保护分类。《Wiley跨学科评论：数据挖掘与知识发现》**11**(3), e1399 (2021)

该问题 NP 完全性的证明

在本附录中，我们给出 **CDPC** 的 NP完全性的形式化证明。首先，我们通过证明第3节中概述的变换正确，来证明其 NP困难性。然后，我们证明该问题属于 NP，从而说明它不仅是 NP-困难的，而且是 NP完全的。

A.1 变换正确性的证明

本节将证明第3.1节中定义、并在第3.2节示例中说明的从 **集合覆盖** 到 **CDPC**的归约有效。回顾一下，**集合覆盖** 的一个实例是一个元组 $(U,\ S,\ k)$，**CDPC** 的一个实例是一个元组 $(f,\ g,\ P,\ a,\ b,\ x,\ y)$，并且我们假设 $|S| \neq k$。

我们按如下方式组织证明 ：

- 我们定义并验证两个有用的引理。
- 我们证明，如果 **SC** 的任一实例的结果为 $\top$（肯定实例），那么在变换后的 **CDPC**实例中，我们定义的策略将产生一次正确的访谈，且结果将为 $\top$。
- 我们证明，如果 **SC**的任一实例的结果为 $\bot$ （否定实例），那么不可能在变换后的 **CDPC**实例中生成满足要求的访谈。

引理1 如果存在一次包含隐私问题的正确访谈，那么必然存在一次与原访谈完全相同、但不包含隐私问题的正确访谈。

证明。 如果在提出名为私有问题的问题时，没有候选人会回答 *0*，也没有候选人会回答 *1*，那么该问题的答案就无法缩小剩余总体的范围，因为所有剩余候选人都会以相同的方式回答私有问题。因此，该问题是多余的，可以删除。

另一方面，假设提出私有问题时，有候选人会回答 *0*，也有候选人会回答 *1*。我们知道 $a > 0$ ，因为我们假设 $|S| \neq k$，问题定义保证 $k \in \mathbb{N}$ 和 $k \leq |S|$，且 $a = (|S|-k)/(|U|*\Omega + \Omega^2 + |S| - k)$。由于 $a > 0$，且至少有一名候选人回答私有问题时会答 *0*，另有一名候选人会答 *1*，如果某次访谈包含私有问题，那么在受访者回答该问题后，我们就无法保证会回答私有问题为 *1* 的候选人所占比例大于或等于 a；当受访者回答 *0* 时，这一点并不成立。因此，所有包含私有问题的访谈，都会在至少一条路径上推断出超出允许范围的隐私信息，因而是不正确的。

既然我们已经证明：*(a)* 当仍有候选人会回答 *0*，也有候选人会回答 *1* 时，任何提出私有问题的访谈都不正确；并且 *(b)* 当剩余候选人的回答只可能是 *0* 或 *1* 时，任何提出私有问题的访谈都可以转化为一个等价且不包含私有问题的访谈，因此我们可以确保结果。□

引理 2 对空候选人进行至多包含 k 个问题的访谈，且不询问私有问题，当且仅当访谈中的问题对应于原始集合覆盖实例解中的集合时，才能排除所有元素候选者。

证明。 元素候选者对与其所属集合相关联的问题回答 *1*，对其他问题回答 *0*。因此，每当空候选受访者回答与集合 $C \in S$ 相关联的问题时，我们就排除所有其关联元素属于 C 的元素候选类型，因为我们知道这些候选人会对该问题回答 *1*，而空候选人对所有问题都回答 *0*；其他元素候选者则不会被排除。此外，由于访谈不能包含私有问题，有效访谈中的所有问题都必须对应于集合。因此，若要通过至多包含 k 个问题的访谈排除所有元素候选者，访谈中各问题对应集合的并集必须等于U，故结论成立。□

A.1.1 集合覆盖判定问题的正实例分析

对于**SC**的正实例，必须至少存在一个元素集合的集合$R \in S$、 $|R| \leq k$，使得 $\cup_{A \in R} A = U$。正如我们在第3.2节给出的示例中所看到的，如果受访者健康，那么其候选人类型只能是空。根据我们定义的策略进行的访谈，将是一系列与属于 R的集合相关联的问题。

我们首先研究受访者健康时的正实例。由于空 候选人对所有问题的回答均为0，并且根据我们的策略，除非已经询问了 R 中的所有问题或收到回答1，否则我们不会结束访谈，因此可以确定我们将询问R中的所有问题。由于这是一个正实例，引理 2保证访谈将排除所有元素 候选人。

除私有情况外，每个问题都会排除恰好一种集合候选类型，具体来说，就是与之关联的集合相同的类型。例如，问题 $\{2\}$ 会排除类型为 $\{2\}$ 的候选人。因此，我们总共最多会排除 k 种集合候选类型。

初始人群中具有隐私特征的候选人比例为 $|S|/(|U| * \Omega + \Omega^2 + |S|)$。访谈后，我们将排除所有的 元素 候选人和 $1 \leq k' \leq k$ 集合 候选人类型（请回忆，我们假设 $U \neq \emptyset$ ，因为如第3.1节所见，我们在变换中将这一平凡情况单独处理，因此 $0 < k'$） ，比例将为 $(|S| - k')/(\Omega^2 + |S| - k')$。该比例介于 $a = (|S| - k)/(|U| * \Omega + \Omega^2 + |S|)$ 和 $b = 1$ 之间：

$$k, k', |S|, |U| \in \mathbb{N}; \quad k' \leq k < |S| \tag{1}$$

$$(|S| - k)/(|U| * \Omega + \Omega^2 + |S|) \leq (|S| - k)/(\Omega^2 + |S| - k') \tag{2}$$

$$(|S| - k)/(\Omega^2 + |S| - k') \leq (|S| - k')/(\Omega^2 + |S| - k') \tag{3}$$

$$(|S| - k)/(|U| * \Omega + \Omega^2 + |S|) \leq (|S| - k')/(\Omega^2 + |S| - k') \leq 1 \tag{4}$$

我们知道 (1) 为真；如果 (1) 为真，则 (2) 和 (3) 成立；如果 (2) 和 (3) 成立，则 (4) 成立。

初始总体中适任候选人的比例为 $\Omega^2/(|U|*\Omega+\Omega^2+|S|)$。访谈后，该比例将为 $\Omega^2/(\Omega^2+|S|-k)$，大于或等于 $y=\Omega^2/(\Omega^2+|S|)$。因此，当受访者适任时，该变换对于正实例是正确的。

接下来，我们考察受访者不适任时的正实例。不适任的候选人类型可以是元素 类型，也可以是集合 类型。我们先来看受访者属于元素类型的情况。

按照我们定义的策略，我们将询问与属于 R 的集合相关的问题，直到得到答案 1。引理 2 保证，如果向空 候选人提问，这些访谈问题会排除所有元素 候选人，而空 候选人对所有问题的答案都是 0。因此，我们可以确定，访谈元素候选人时，他们最终会回答 1。否则，若向空候选人提问，这些访谈问题就无法排除他们。

由于它们最终必须回答 1，因此空 候选人会被排除。我们不会排除至少一种元素 候选人类型（即受访者的类型）。如上所述，每个非私有 问题恰好排除一种集合候选人类型。因此，访谈会排除 k' 集合 候选人类型，其中$k'\in\mathbb{N}$ 且 $k'\leq k$。

访谈后，剩余人口中具有隐私特征的候选人比例为 $(|S|-k')/((|U|-u')*\Omega+|S|-k')$，其中 $u'\in\mathbb{N}$y $u'<|U|$。我们需要证明 $(|S|-k')/((|U|-u')*\Omega+|S|-k')\in[a,b]$:

$$(|S|-k')/((|U|-u')*\Omega+|S|-k')\geq(|S|-k')/(|U|*\Omega+|S|) \tag{5}$$

$$(|S|-k')/(|U|*\Omega+|S|)\geq(|S|-k)/(|U|*\Omega+\Omega^2+|S|)=a \tag{6}$$

由于 $b=1$，且显然具有隐私特征的候选人比例不会大于 1，因此该值属于 $[a,b]$。

剩余总体中适任候选人的比例为 0，因为我们已排除所有适任候选人以及 $0\leq x$。当受访者属于元素类型时，该变换对正实例是正确的。

由此可见，研究受访者属于集合类型的情况很容易。这些类型的候选人会对隐私问题回答 1，并且只会对另一个问题回答 1。因此，他们可以对所有面试问题都回答 0，或对除一个问题外的所有面试问题回答 0。

如果受访者对任何问题回答 1，那么我们就处于一种等价情形：受访者属于元素类型。唯一的区别是，现在可以排除所有元素候选人，而不是 $u'<|U|$，$u'\leq|U|$。这一变化不影响先前的论证。

如果受访者对所有问题的答案都是 0，那么我们就处于一种等同于受访者属于空 类型的情形。我们排除相同的候选人，适任候选人与具有隐私

特征的候选人的比例相同。正如第3.2节中的示例所示，在这种情况下，访谈的适应度预测不正确，但**CDPC** 的定义允许存在一定的误差范围。

因此，我们证明了：如果**集合覆盖** 实例是肯定的，那么变换后的**CDPC** 实例也为肯定。

A.1.2 集合覆盖判定问题否定实例的分析

在这种情况下，我们需要证明：如果某个**集合覆盖** 实例的结果为 $\perp$，那么由其变换得到的**CDPC** 实例的结果也为 $\perp$。

首先，我们证明，不包含隐私问题的任何正确访谈都不会超过 k 个问题。显然，任何包含超过 k 个问题的访谈都必须提出第（$k+1$）个问题，其中 $k+1 \leq |S|$。如果我们证明，所有包含 $k+1$ 个问题的访谈所推断出的隐私信息都超出允许范围，那么就能证明，所有包含超过 k 个问题的访谈都是不正确的。

要使一次访谈正确，无论总体中的哪位候选人作为受访者，访谈都必须正确。假设受访者的类型为空。如上所述，每个非私有 问题都恰好排除一种集合候选类型。总计而言，我们排除 $k+1$ 种集合 候选类型、不排除任何空候选类型，以及 $u' \in \mathbb{N}$ 种元素 候选类型，其中 $0 \leq u' \leq |U|$。

回顾一下，初始种群中具有隐私特征的候选人比例为 $|S|/(|U| * \Omega + \Omega^2 + |S|)$，且 $a = (|S| - k)/(|U| * \Omega + \Omega^2 + |S| - k)$ 和 $b = 1$。访谈后，剩余人口中具有隐私特征的候选人比例将为 $(|S| - k - 1)/((|U| - u') * \Omega + \Omega^2 + |S| - k - 1)$。由于 $\Omega = 2 * |U| * |S| > 0$，该比例小于$b$。为了证明它不可能属于 $[a, b]$，我们需要证明 $(|S| - k - 1)/((|U| - u') * \Omega + \Omega^2 + |S| - k - 1) < a$。已知：

$$|U|, |S| \in \mathbb{N}^+; \quad \Omega = 2 * |U| * |S|; \quad 0 \leq u' \leq |U|; \quad 0 \leq k < |S| \tag{7}$$

以下结论成立，因为 $|U|$ 和 $|S|$ 均为正数：

$$|S| < 2 * |U|^2 * |S| + 2 * |U|^2 * |S|^2; \tag{8}$$

我们可以在不等式两边加上 $2 * |U|^2 * |S|^2$，然后进行几项基本运算：

$$2 * |U|^2 * |S|^2 + |S| < 2 * |U|^2 * |S| + 4 * |U|^2 * |S|^2; \tag{9}$$

$$(2 * |U|^2 * |S| + 1) * |S| < |U| * (2 * |U| * |S|) + (2 * |U| * |S|)^2; \tag{10}$$

$$|S| < \frac{|U| * \Omega + \Omega^2}{|U| * \Omega + 1}; \tag{11}$$

如果我们的不等式成立，那么以下结论也必须成立：

$$|S| - k \leq |S| < \frac{|U| * \Omega + \Omega^2}{|U| * \Omega + 1} \leq \frac{|U| * \Omega + \Omega^2 + |S| - k}{|U| * \Omega + 1}; \tag{12}$$

$$|S| - k < \frac{|U| * \Omega + \Omega^2 + |S| - k}{|U| * \Omega + 1}; \tag{13}$$

$$\frac{|U| * \Omega + 1}{|U| * \Omega + \Omega^2 + |S| - k} < \frac{1}{|S| - k}; \tag{14}$$

注意，$|U| * \Omega + 1 = (|U| * \Omega + \Omega^2 + |S| - k) - (\Omega^2 + |S| - k - 1)$：

$$\frac{(|U| * \Omega + \Omega^2 + |S| - k) - (\Omega^2 + |S| - k - 1)}{|U| * \Omega + \Omega^2 + |S| - k} < \frac{1}{|S| - k}; \tag{15}$$

$$1 - \frac{\Omega^2 + |S| - k - 1}{|U| * \Omega + \Omega^2 + |S| - k} < \frac{1}{|S| - k}; \tag{16}$$

我们可以在被减数的分数分子中加上 $(|U| - u') * \Omega$，同时保持不等式成立。然后，我们将重写不等式右侧，并继续进行基本运算：

$$1 - \frac{(|U| - u') * \Omega + \Omega^2 + |S| - k - 1}{|U| * \Omega + \Omega^2 + |S| - k} < \frac{(|S| - k) - (|S| - k - 1)}{|S| - k}; \tag{17}$$

$$1 - \frac{(|U| - u') * \Omega + \Omega^2 + |S| - k - 1}{|U| * \Omega + \Omega^2 + |S| - k} < 1 - \frac{|S| - k - 1}{|S| - k}; \tag{18}$$

$$\frac{|S| - k - 1}{|S| - k} < \frac{(|U| - u') * \Omega + \Omega^2 + |S| - k - 1}{|U| * \Omega + \Omega^2 + |S| - k}; \tag{19}$$

最后，我们可以得出以下不等式成立：

$$\frac{|S| - k - 1}{(|U| - u') * \Omega + \Omega^2 + |S| - k - 1} < \frac{|S| - k}{|U| * \Omega + \Omega^2 + |S| - k} \tag{20}$$

结果表明，剩余总体中具有隐私特征的候选人比例不在 $[a, b]$ 之外。

此外，引理1保证，如果存在一个包含隐私问题的正确访谈，那么必定也存在一个正确访谈，除不包含隐私问题外，其余均与原访谈相同。有鉴于此，要考虑由否定实例的集合覆盖经变换得到的 CDPC 实例为正实例的情形，唯一剩下的可能是：访谈包含至多 k 个问题，且其中没有任何一个是隐私问题。

根据引理1，排除这种情形也就排除了所有包含隐私问题的情形。如前所述，如果访谈推断出隐私信息，或未能足够准确地确定受访者的适应度，那么该访谈就是不正确的。即使访谈对其余候选人都有效，只要对一名候选人无效，该访谈就是不正确的。我们通过聚焦于受访者属于空类型的情形，证明至多包含 k 个问题且没有任何一个是隐私问题的访谈绝不可能正确。

引理 2 告诉我们，由于原始集合覆盖实例的答案是否定的，这次访谈不可能排除所有的元素候选人。它不会排除任何空

候选人和 $k' \in \mathbb{N}$ 集合候选人，其中 $0 \leq k' \leq k$，因为与原始实例中的某个集合相关联的每个问题（除私有问题外）都会恰好排除一个集合候选人。

初始总体中适任候选人的比例为 $\Omega^2/(|U|*\Omega + \Omega^2 + |S|)$。访谈后，该比例将为 $\Omega^2/((|U| - u') * \Omega + \Omega^2 + |S| - k')$，其中$u' \in \mathbb{N}$ 和 $u' < |U|$。

$$\Omega^2/((|U| - u') * \Omega + \Omega^2 + |S| - k') \leq \Omega^2/(\Omega + \Omega^2 + |S| - k') = \tag{21}$$

$$\Omega^2/(2 * |U| * |S| + \Omega^2 + |S| - k') \tag{22}$$

并且，由于 $0 \leq k' \leq k \leq |S|$ 和 $|U| > 0$:

$$\Omega^2/(2 * |U| * |S| + \Omega^2 + |S| - k') \leq \Omega^2/(2 * |S| + \Omega^2 + |S| - |S|) \leq \tag{23}$$

$$\Omega^2/(\Omega^2 + 2 * |S|) < \Omega^2/(\Omega^2 + |S|) = y \tag{24}$$

最后，由于该比例大于 $x = 0$ （因为 $\Omega > 0$），它属于 $[x, y]$，因此访谈不会将总体限定在适任候选人比例足够低（$[0, x]$）或足够高（$[y, 1]$）的子集内，因而是不正确的。我们已经证明，如果 **SC** 的实例为否定实例，那么转换后的 **CDPC** 实例也为否定实例。

A.2 NP 中的成员资格

如果对于每个问题实例，在给定一个解的假设（证书）后，能够在多项式时间内验证该解的有效性，则该问题属于 NP。**CDPC** 的证书将是问题树。受访者的答案是确定性的，即他们总是以相同的方式回答相同的问题，因此，一个问题的可能答案会将剩余总体（与该受访者匹配的候选人类型）划分为两两不相交的集合。树中的每条路径都表示一系列问题（顶点）和答案（边），而每个节点表示初始总体中会按照各自答案所指示的方式回答从根节点到该节点路径上问题的候选人的子集。

在证明**CDPC**和**GCOPC**属于 NP 的其余部分中，设初始种群中的候选类型数量为 $n \in \mathbb{N}$，问题总数为 $m \in \mathbb{N}$。

对于证明该问题的解的树中的每个节点，其子节点的剩余总体两两不相交；这些集合由候选类型组成，而候选类型会按照相应子节点路径所指示的方式作答。显然，这些集合的并集等于父节点的剩余总体。通过归纳推理，我们可以推断，任何正确证书的叶节点所对应的剩余总体两两不相交，且其并集为初始种群。因此，叶节点的数量不会超过候选类型的数量（n）。图1曾作为交互式访谈的示例使用，它展示了这一性质。

由于我们的问题定义不允许同一路径上出现重复问题，且有 m 个不同的问题，因此树的最大深度为 m。此外，路径数与叶节点数相同，因此树最多有 $n*m$ 个节点。任何问题实例的规模下界必须为 $n*m$，因为必须为初始种群中的所有候选类型定义适应度函数和数量函数，并且每种候选类型都需要 m 的空间来表示（候选类型是全函数问题 $\rightarrow$ 答案，且共有 m 个问题）。因此，树中的节点数相对于其所证明的问题实例规模呈线性关系。

鉴于此，我们定义一个多项式验证算法。假设我们正在验证一棵维度适当的树——如果并非如此，那么我们可以确定它不可能是一个正确的证书。该算法包括遍历这棵树，并存储当前处理路径上的答案。到达每个叶节点时，我们必须确定哪些候选类型可能以所存储的方式作答。为此，我们遍历候选类型针对最多 m个问题所给出的答案，其中函数 g 有定义。对于某个叶节点的剩余候选类型，只需应用适应度函数和数量函数，即可在线性时间内（相对于剩余候选类型的数量）检查它们是否满足问题实例中规定的适应度和隐私要求。

请注意，无需在树的每个节点处检查隐私要求是否成立。如果某个节点的所有子节点都满足该要求，那么其父节点也满足隐私要求。我们通过反证法证明这一性质。假设某个节点的所有子节点都满足隐私要求。如果父节点不满足该要求，那么父节点中以某种方式回答某个隐私问题的候选人的比例将落在某个区间之外（该区间取决于 a 和 b）。然而，由于子节点满足该要求，我们知道，在子节点中以相同方式回答同一个隐私问题的候选人比例必须处于该区间内。由于父节点候选人的比例是各子节点对应比例的加权平均值，因此父节点的比例不可能落在该区间之外。我们由此得出矛盾，并可以断定所要证明的性质成立。因此，通过应用归纳推理，我们可以看出，只需在叶节点处检查隐私要求即可。

每个叶节点都可以在 $n*m$时间内处理，因此验证算法可以在 $n*m+n^2*m$ 时间内完成。令 $t \geq n*m$ 为输入规模。那么，验证算法的时间复杂度为 $\Theta(t^2)$。

因此，已证明**CDPC** 和**GCOPC** 属于 NP。