第 1 章 最优化问题的数学模型与基本理论

数学是解决问题的工具, 而最优化则是这个工具中最锋利的刀刃。

最优化是大自然的选择, 也是人类的执着追求。本章首先介绍最优化问题的一些实际案例, 进而引出最优化问题的数学模型, 并指出区分全局最优化和局部最优化是重要的也是无奈的。然后介绍最优化问题的基本理论, 特别是全局最优解的存在性, 以及从最优性条件的角度介绍全局最优解的几乎不可知性。

1.1 最优化问题及其数学模型

本节先介绍最优化问题的一些实例, 从中可以初步感受到最优化为何是大自然的选择和人类的执着追求。然后, 在此基础上, 介绍最优化问题的数学模型, 为后续内容提供基本的研讨平台。

1.1.1 最优化问题举例

1) 封闭系统的熵增定律

物理学中有一个非常重要的熵增定律, 它描述的是封闭系统总是朝着熵最大化的方向发展的。该定律来自于热力学第二定律, 表明了物理系统在没有外力作用的条件下, 熵 (entropy) 必定会越来越大, 也即系统变得越来越混乱。因此, 如果我们的宇宙是孤立的封闭系统, 熵最大化是其不二的选择或 "努力" 的方向。其数学模型可表示为

$$\max_{t} \quad E(t) \tag{1.1}$$

其中，$E(t)$ 表示 t 时刻系统的熵。由于宇宙系统的熵总在增加, 说明自然过程是不可逆的, 这也引出了 "时间之箭" 的概念。总之, 模型 (1.1) 表明了最优化是大自然的选择, 后面的例子又表明, 最优化也是人类的不懈追求。

2) 消费者的效用最大化问题

经济系统是人类最基础、最重要的系统之一。经济学的研究表明, 经济系统中的所有参与者 (消费者、生产者、政府等) 都在追求最优化。比如, 消费者追求效用 (utility) 最大化, 生产者追求利润最大化和成本最小化, 而政府借助 "看得见的手" 调控经济系统, 努力使其状态最佳。

这里先看消费者的选择问题。在微观经济学中, 理性的消费者在商品价格给定、可支配收入给定的情况下, 追求总效用的最大化。其数学模型可表述为

$$\begin{aligned} \max_{\boldsymbol{x}\in\mathbb{R}^n} \quad & U(\boldsymbol{x}) \\ \text{s.t.} \quad & \boldsymbol{p}^{\mathrm{T}}\boldsymbol{x} \leqslant I \\ & \boldsymbol{x} \geqslant 0 \end{aligned} \tag{1.2}$$

其中, $\boldsymbol{x}$ 是 n 维向量, 描述消费者对 n 种商品的购买数量; $U(\boldsymbol{x})$ 是这些商品带给消费者的总效用。每种商品的价格组成了向量 $\boldsymbol{p}$, 消费者的可支配收入用 I 表示。模型 (1.2) 描述了理性的消费者在量入为出 (不使用金融手段) 的条件下, 努力实现效用的最大化。通过对该模型的调整或修改, 经济学家们可以进一步考虑更多的因素, 比如非理性行为、效用在时间跨度上的折旧、收入随时间的变化, 等等。总之, 消费者在各种情况下都追求效用的最大化。

3) 生产者的利润最大化问题和成本最小化问题

下面考虑经济系统中生产者的选择。在微观经济学中, 理性的生产者在成本一定的条件下追求企业利润的最大化。其数学模型可描述为

$$\begin{aligned} \max_{\boldsymbol{x}\in\mathbb{R}^n} \quad & py - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} \\ \text{s.t.} \quad & \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} \leqslant C \\ & \boldsymbol{x} \geqslant 0 \end{aligned} \tag{1.3}$$

其中, $\boldsymbol{x}$ 是 n 维向量, 描述 n 种原材料的购买数量; $\boldsymbol{w}$ 表示这些原材料的价格向量。企业利用这些原材料生产出商品, 其售价为 p, 产量为 $y = g(\boldsymbol{x})$, 这里 g 表示其生产函数。由于资源是有限的, 企业成本也是有限的, 企业在一定时间内能够使用的成本 C 是给定的。

经济学上已经证明, 利润最大化模型 (1.3) 与下面的产量一定的条件下的成本最小化模型等价。

$$\begin{aligned} \min_{\boldsymbol{x}\in\mathbb{R}^n} \quad & \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} \\ \text{s.t.} \quad & g(\boldsymbol{x}) \geqslant Y \\ & \boldsymbol{x} \geqslant 0 \end{aligned} \tag{1.4}$$

其中, Y 表示产量, 这里是给定的常数。如果这个产量等于利润最大化模型 (1.3) 中的最优产量, 即 $Y = g(\boldsymbol{x}^*)$, 那么可以证明, 成本最小化模型 (1.4) 的解就等于利润最大化模型中的 C。这一结果说明, 理性生产者的利润最大化要求企业成本最小化, 反之亦然。

以上两个模型表明, 理性的生产者在追求利润最大化和成本最小化, 且这两个追求在数学模型上是相互等价的。当然, 经济研究可能会考虑更多的现实因素, 但都是对模型 (1.3) 和模型 (1.4) 的调整或修订, 本质上仍然是求解各种最优化问题。

下面给出一些具体经济场景中的最优化问题。

4) 生产计划问题

某企业用 m 种原材料来生产 n 种产品, 为了生产 1 单位第 i 种产品需要 a_{ij} 单位的第 j 种原材料, $i = 1, 2, \cdots, n; j = 1, 2, \cdots, m$。已知第 j 种原材料只有 b_j 单位, 而卖出 1 单位第 i 种产品可以获利 c_i 元。请问该企业的最优生产计划是什么?

在利润最大化的假设下, 该问题的数学模型如下:

$$
\begin{aligned}
\max_{\boldsymbol{x}} \quad & \sum_{i=1}^{n} c_i x_i \\
\text{s.t.} \quad & \sum_{i=1}^{n} a_{ij} x_i \leqslant b_j, \quad j = 1, 2, \cdots, m \\
& x_i \geqslant 0, \qquad\qquad i = 1, 2, \cdots, n
\end{aligned} \tag{1.5}
$$

其中, x_i 表示第 i 种产品的产量。模型 (1.5) 具有广泛的适用性, 大量工厂的生产计划都可以描述成这一模型或其变形。这类问题被称为线性规划问题, 在 20 世纪上半叶得到了空前的重视和发展, 能有效求解这类问题的单纯形法, 被认为是人类有史以来产生最大经济价值的算法。

5) 食谱问题

为了达到身体健康, 人必须每天摄入 m 种营养成分, 且每种至少需要数量 $b_i(i = 1, 2, \cdots, m)$。假设人经常食用的食物有 n 种, 每种的单位价格为 $C_j(j = 1, 2, \cdots, n)$。食谱问题要求用最少的成本达到人每天的营养需要。其数学模型可表述如下:

$$
\begin{aligned}
\min_{\boldsymbol{x}} \quad & \sum_{j=1}^{n} C_j x_j \\
\text{s.t.} \quad & \sum_{j=1}^{n} a_{ij} x_j \geqslant b_i, \quad i = 1, 2, \cdots, m \\
& x_j \geqslant 0, \qquad\qquad j = 1, 2, \cdots, n
\end{aligned} \tag{1.6}
$$

其中, x_j 表示第 j 种食物的数量; a_{ij} 表示每单位第 j 种食物含有第 i 种营养成分的数量。

可以看出, 食谱问题的数学模型 (1.6) 与生产计划的数学模型 (1.5) 类似, 它们都属于线性规划问题, 是最优化问题中最早被研究的一类, 也是发展最成熟的一个分支。

6) 物流运输问题

某物流公司负责的商品有 m 个产地和 n 个销售地, 每个产地的供应量为 $a_i(i = 1, 2, \cdots, m)$, 每个销售地的销售量为 $b_j(j = 1, 2, \cdots, n)$。假设从第 i 个产地到第 j 个销售地的单位运价为 $c_{ij}(i = 1, 2, \cdots, m; j = 1, 2, \cdots, n)$, 并假设总供应量大于总销售量, 要求用最小的成本完成整个运输计划。该问题的数学模型可表述如下:

$$
\begin{aligned}
\min_{\boldsymbol{x}} \quad & \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{i=1}^{m} x_{ij} = b_j, \quad j = 1, 2, \cdots, n \\
& \sum_{j=1}^{n} x_{ij} \leqslant a_i, \quad i = 1, 2, \cdots, m \\
& x_{ij} \geqslant 0, \qquad i = 1, 2, \cdots, m; j = 1, 2, \cdots, n
\end{aligned} \tag{1.7}
$$

其中, x_{ij} 表示从第 i 个产地到第 j 个销售地的运输量。

模型 (1.7) 仍然属于线性规划问题, 在物流运输领域具有重要应用。

7) 指派 (分配) 问题

某部门有 n 项任务, 需要从 m 个人选出 n 个人去完成 $(m > n)$, 每个任务仅需要一人。由于任务的性质和每个人的专长不同, 假设安排第 i 个人去完成第 j 项任务所需成本为 $c_{ij}(i = 1, 2, \cdots, m; j = 1, 2, \cdots, n)$。如何分配任务使得总成本最小呢? 指派问题的数学模型可表述如下:

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \\ \text{s.t.} \quad & \sum_{i=1}^{m} x_{ij} = 1, \qquad j = 1, 2, \cdots, n \\ & \sum_{j=1}^{n} x_{ij} \leqslant 1, \qquad i = 1, 2, \cdots, m \\ & x_{ij} = 0, 1, \qquad i = 1, 2, \cdots, m; j = 1, 2, \cdots, n \end{aligned} \tag{1.8}$$

其中, 如果指派第 i 个人去完成第 j 项任务则 $x_{ij} = 1$; 否则 $x_{ij} = 0$。

从数学模型可以看出, 指派问题非常类似于运输问题, 本质上是运输问题的一个特例。它们都属于线性规划领域的经典问题。

8) 数据拟合问题/线性回归问题/最小二乘问题/机器学习问题

在大量的科学和工程领域, 经常要对已知数据进行曲线拟合或规律发现 (知识发现), 其本质也是一个最优化问题。比如, 在经典的线性回归问题中, 有 n 组数据 $(\boldsymbol{x}_i, \boldsymbol{y}_i), i = 1, 2, \cdots, n$, 要求用一条直线 $\hat{\boldsymbol{y}} = \boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{b}$ 来拟合这些数据。根据高斯的理念, 拟合的目标是使得残差 $\boldsymbol{y} - \hat{\boldsymbol{y}}$ 的平方和最小。因此, 该问题的数学模型可表述为

$$\min_{\boldsymbol{a},\boldsymbol{b}} \quad \sum_{i=1}^{n} \left[\boldsymbol{y}_i - (\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x}_i + \boldsymbol{b})\right]^2 \tag{1.9}$$

求解出模型 (1.9) 后, 得到的回归直线 $\hat{\boldsymbol{y}} = \boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{b}$ 也被称为线性分类器, 后者是机器学习的概念。可以把线性分类推广到一般的非线性分类问题, 该问题的数学模型可描述为

$$\min \quad \sum_{i=1}^{n} [\boldsymbol{y}_i - f(\boldsymbol{x}_i)]^2 \tag{1.10}$$

这里 $\hat{\boldsymbol{y}} = f(\boldsymbol{x})$ 是一个一般的分类器。事实上, 机器学习研究的一个重要内容, 就是开发不同类型的分类算法, 而其本质就是建立合适的最优化模型, 并想办法求解该模型。

以上例子只是最优化问题广泛应用的一些缩影, 可以说, 在任何的经济系统、社会系统或物理系统中都存在大量的最优化问题。再强调一遍, 最优化是大自然的选择, 也是人类的不懈追求。研究最优化问题具有重要的科学意义和工程价值。

1.1.2　最优化问题的数学模型

从以上最优化问题的案例中, 我们可以提炼出最优化问题的如下一般模型:

$$
\begin{aligned}
\min_{\boldsymbol{x}} \quad & f(\boldsymbol{x}) \\
\text{s.t.} \quad & \boldsymbol{x} \in \Omega \subseteq \mathbb{R}^n
\end{aligned}
\tag{1.11}
$$

其中, n 元函数 $f(\boldsymbol{x})$ 称为**目标函数**; $\boldsymbol{x}$ 称为**决策变量**, 是一个 n 维向量; Ω 称为**可行域**, 可行域中的点称为该问题的**可行解**。如果 $\Omega = \mathbb{R}^n$, 则称模型 (1.11) 为**无约束最优化问题**, 否则称其为**约束最优化问题**。

可行域是约束条件的交集或公共部分, 如果把约束条件写出来, 可以得到下面的模型:

$$
\begin{aligned}
\min_{\boldsymbol{x}\in\mathbb{R}^n} \quad & f(\boldsymbol{x}) \\
\text{s.t.} \quad & \begin{cases} c_i(\boldsymbol{x}) = 0, & i \in \mathcal{E} \\ c_j(\boldsymbol{x}) \geqslant 0, & j \in \mathcal{I} \end{cases}
\end{aligned}
\tag{1.12}
$$

其中, $c_i(\boldsymbol{x}), c_j(\boldsymbol{x})$ 分别被称为**等式约束函数和不等式约束函数**。集合 $\mathcal{E}, \mathcal{I}$ 可以为空集, 此时表示没有此类约束。

模型 (1.11) 和模型 (1.12) 都是最小化问题, 那如何描述最大化问题呢？只需要在目标函数前加一个负号, 此时的最小化问题本质上就是一个最大化问题。也就是说, 最大化问题可以描述为最小化问题, 当然, 反之亦然。由于利用计算机求解最大化问题时容易产生 “溢出”, 所以一般将最大化问题改写为最小化问题。因此, 本书中的最优化问题除非特别指出一般指最小化问题。

1) 最优化模型的分类

借助模型 (1.11) 和模型 (1.12), 可以解释何为线性规划, 何为非线性规划。线性规划 (linear programming, LP) 模型指的是, 目标函数和约束函数都是线性函数; 而非线性规划 (nonlinear programming, NLP) 模型指的是, 目标函数或者约束函数中至少有一个是非线性函数。因此, 在前面举的例子中, 模型 (1.1), 模型 (1.2), 模型 (1.9), 模型 (1.10) 一般都是非线性规划问题。

除了线性规划和非线性规划这一重要分类外, 最优化问题还有其他一些分类方法。比如, 二次规划 (quadratic programming, QP) 模型指的是目标函数是二次函数, 而约束函数都是线性函数; 整数规划 (integer programming, IP) 模型指的是决策变量只能取值为整数; 特别地, 如果决策变量只能取值为 0 和 1, 则称为 0-1 规划。整数规划是组合优化问题的一个分支, 后者又叫作离散优化, 指的是可行域中只有有限多个点或者可列无限多个点。与离散优化对应的是连续优化, 指的是可行域中有不可列无限多个点, 比如取值为实数。此外, 很多优化场景要求对多个目标同时进行优化, 这些目标不是相互独立的, 某个目标的选择会影响另一个目标的选择, 这类问题称为多目标优化问题。

最优化问题最重要的分类是凸优化和非凸优化。凸优化指的是目标函数为凸函数, 可行域为凸集 (关于凸函数和凸集的介绍可参阅 1.2.4 节); 而非凸优化指的是目标函数不是凸函

数, 或者可行域不是凸集。为什么说这一分类是最重要的呢? 主要是因为凸优化具有良好的数学结构, 人类已经在理论上很好地解决了这一问题; 而对于非凸优化问题, 人类还没有很好的解决办法, 是目前最优化领域的研究热点。另外, 所有的线性规划问题都是凸优化问题, 非凸优化问题必定是非线性规划问题 (图 1.1)。关于凸优化问题请读者参阅文献 [1]。

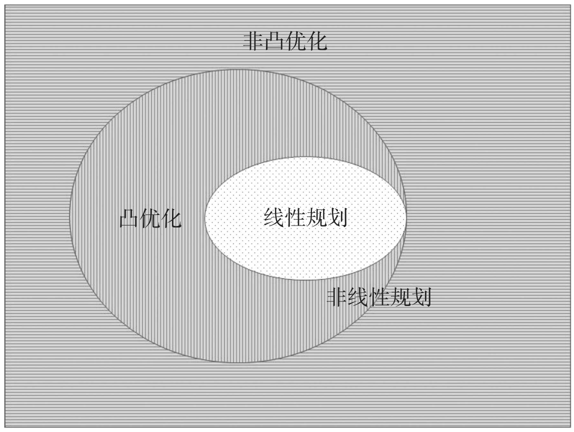

图 1.1　最优化问题的两大重要分类

本书主要研究全局最优化问题, 与它相对应的是局部最优化问题。顾名思义, 前者力图寻找最优化问题的全局最优解, 而后者试图找到最优化问题的局部最优解, 它们的具体含义详见 1.1.3 节。如果最优化问题是一个凸优化问题, 则它的所有局部最优解都是全局最优解。因此, 在凸优化领域, 全局最优化与局部最优化是等价的。所以, 全局最优化问题的研究主要关注的是如何找到非凸优化问题的全局最优解。

2) 最优化模型的作用

建立最优化问题的数学模型并划分那么多分类有何意义和作用呢? 这里我们需要提出一个重要观点, 那就是最优化问题的研究和学习类似于学医, 需要根据不同的病情开不同的药方。这里的 “病” 对应于最优化问题, 不同的病情对应于最优化模型的不同, 而 “药方” 则对应于不同的求解算法。换句话说, 正如没有万能的药物能够有效地治疗各种疾病, 也不存在一个万能的算法可以高效求解各种不同类型的最优化问题。

我们需要根据最优化模型的不同特点, 采用不同的算法来求解它。这就是我们建立最优化问题的数学模型, 并给它准确分类的意义和作用。这一步完成的好坏直接影响后续的算法设计以及算法的求解效果。关于这一点, 请读者务必铭记于心, 在本书的学习中不断对比和反思, 方能取得良好的学习效果。

3) 最优化问题建模的关键

既然最优化问题的数学模型如此重要, 下面给出建模的一些关键点。

首先, 要理解实际问题本身, 搞清楚它的目标有几个? 分别是什么? 这些目标是相互独立的吗? 如果只有一个目标, 则建立一个单目标最优化模型; 如果多个目标相互独立, 则可以建立多个单目标最优化模型; 否则, 建立一个多目标最优化模型。

其次, 在每个最优化模型中, 搞清楚决策 (控制) 变量有几个? 分别是哪些? 它们的取值是离散的还是连续的? 目标函数如何用决策变量表示出来?

再次, 在每个最优化模型中, 有哪些可能的约束? 每个约束函数怎么由决策变量表示出来? 约束函数是等式约束还是不等式约束?

最后, 在以上每个步骤中, 如果有多种方案可供选择, 一般优先尝试把它建成凸优化问题, 因为有成熟的算法可以高效求解这类问题。另外, 如果存在可用的梯度信息也有助于设计更高效的算法。

需要指出的是, 为实际问题建立最优化模型的过程往往不是一次成功的, 而是需要多次尝试才能确定下来的。这主要是因为对实际问题的理解需要有一个过程。另外, 建模的几个关键环节也可能有多种组合, 需要逐个尝试才能确定最合适的方案。

4) “隐式” 最优化模型

在大量工程实践中, 并不是所有最优化问题都能很好地把模型显式地写出来。比如, 在大量的船舶设计或航空器设计中, 船体或表面体需要满足流体力学或空气动力学方程, 这些方程很难求解出显式表达式。所以, 当遇到这些问题时, 目标函数可能无法显式地写出来 (存在但人类写不出来), 目标函数值只能通过仿真的手段得到, 且通常成本高昂。

虽然没有显式的最优化模型, 但一般并不妨碍确定模型的决策变量。至于约束条件, 如果难以确定, 则可以选择一个较大的矩形或超矩形, 确保其包含整个可行域。等找到最优解后, 再结合实际判断是否可行, 如果不可行则调整矩形或超矩形的范围, 再次尝试。

1.1.3　全局最优化问题与局部最优化问题

本书主要关注全局最优化问题, 特别是非凸最优化问题的全局最优解的求解。这就需要搞清楚什么是最优化问题的全局最优解。

定义 1.1　如果存在 $\epsilon > 0$, 使得

$$f(\bar{\boldsymbol{x}}) \leqslant f(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in B(\bar{\boldsymbol{x}}, \epsilon) \subset \Omega$$

其中,

$$B(\bar{\boldsymbol{x}}, \epsilon) = \{\boldsymbol{x} | \; ||\boldsymbol{x} - \bar{\boldsymbol{x}}|| < \epsilon\}$$

则称 $\bar{\boldsymbol{x}}$ 为问题 (1.11) 的局部最优解。如果存在 $\boldsymbol{x}^* \in \Omega$, 使得

$$f(\boldsymbol{x}^*) \leqslant f(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \Omega$$

则称 $\boldsymbol{x}^*$ 为问题 (1.11) 的全局最优解。

简而言之, 全局最优解要求在可行域内没有别的位置的目标函数值比它的更小, 如图 1.2 中的 B 点; 而局部最优解只要求在可行域的某个局部没有别的更好位置, 如图 1.2 中的 A, C 点。

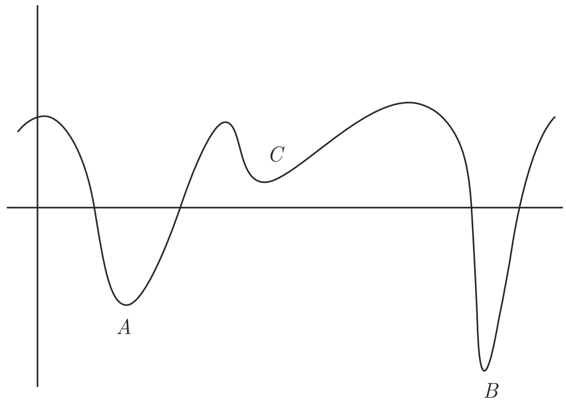

图 1.2　局部最优解 (A, B, C) 和全局最优解 (B)

在定义 1.1 中, 用到了球形邻域 $B(\bar{\boldsymbol{x}},\epsilon)\subset\Omega$, 这意味着局部最优解一定在可行域的内部, 而不会在边界上。反之, 全局最优解可能出现在可行域的内部, 也可能出现在可行域的边界。如果出现在可行域的内部, 则它一定也是一个局部最优解。寻找全局最优解只需要关注可行域内部的局部最优解以及可行域的边界点。

在最优化领域, 寻找全局最优解的分支称为全局最优化 (global optimization), 而聚焦于获得局部最优解的分支称为局部最优化 (local optimization)。从定义 1.1 可以看出, 最优化问题的局部最优解只是在一个局部 (可能是很小的一个邻域) 是最好的, 但是全局最优解则要求在整个可行域都是最好的。显然, 我们希望得到的是最优化问题的全局最优解。那么, 为什么还要引入所谓局部最优解这个概念呢? 还有一个重要分支聚焦于寻找它呢?

上述奇怪的情形反映了最优化领域的窘境: 求解最优化问题的全局最优解是一个很困难的任务, 一个原因是真正的全局最优解通常很难找到; 更窘迫的是, 即便算法找到了全局最优解, 也几乎无法确认这一点。于是只能退而求其次去寻找最优化问题的局部最优解, 并试图间接找到全局最优解。除此之外, 局部最优化这个分支大行其道的另一个原因是: 存在寻找局部最优解的引导信息, 在该信息的引导下通常可以找到一个局部最优解, 且这一机制的数学原理非常优美。

当然, 上一段话的信息量很大, 要充分理解并不容易。下面从最优化问题的基本理论出发来慢慢理解它们。

1.2　最优化问题的基本理论

本节主要介绍最优化问题 (1.11) 的几个基本理论结果: 最优解的存在性、最优化的必要条件和充分条件等, 特别关注全局最优化与局部最优化在最优性条件上的不同。

另外, 从本节开始, 凡是以 ♠ 开头的内容都表示铺垫性知识。

1.2.1 全局最优解的存在性

首先要指出, 最优化问题 (1.11) 并不总是存在最优解! 最优解的存在性需要考虑目标函数和可行域双方的特征。下面的 Weierstrass 定理 (即定理 1.1) 表明, 在一定条件下最优解总是存在。

定理 1.1 *紧集上的连续函数必定存在最大值和最小值。*

♠ 紧集指的是有界闭集。

该定理在高等数学中的版本为 “闭区间上的连续函数必定存在最大值和最小值”, 这里的闭区间就是常见的紧集。Weierstrass 定理的两个条件不是很强, 大多数常见问题的目标函数可以认为满足一定的连续性, 而可行域也可以满足一定的有界性和闭性 (从而满足一定的紧性)。因此, Weierstrass 定理为后续关于最优化问题的介绍提供了基本的前提和研讨平台。

1.2.2 全局最优解的 NP-hard 性质

要理解 NP-hard 这个概念, 需要先知道什么是算法的时间复杂度。算法的时间复杂度是算法计算时间的重要度量指标, 它指的是随着输入变量的增长, 算法的计算量 (一般用四则运算的计算次数来度量) 以什么样的相对规模增长, 特别关注当输入变量充分大 (甚至无穷大) 时的相对数量级。比如, 当采用高斯消元法来求解具有 n 个未知数的线性方程组时, 其时间复杂度为 $O(n^3)$, 这表明高斯消元法的计算量是 n^3 的常数倍。

在计算复杂度这个领域中, 一个非常重要的关注点是它的形式是不是 “多项式” 的。比如 $O(n^a)$ 对于任何常数 a 都是多项式复杂度, 然而 $O(2^n), O(n!)$ 就不是多项式复杂度。一般认为, 多项式复杂度是人类能够接受的最高等级的复杂度, 所以, 超过多项式复杂度的算法往往被认为是不可接受的。

1) P、NP、NP-complete 和 NP-hard

在人类探究算法时间复杂度的过程中, 人们发现可以根据时间复杂度对问题进行分类。最基本且重要的两个类是 P 类问题和 NP 类问题: 如果一个问题可以找到一个能在多项式时间内解决它的算法, 那么这个问题就属于 P 类问题; 如果可以在多项式时间内验证一个解, 那么这个问题就属于 NP 类问题[2]。很显然, 只有能够在多项式时间内验证一个解, 才有可能在多项式时间内求解该问题, 因此 P 类问题必定是 NP 类问题。然而, NP 类问题是否是 P 类问题尚不清楚。在信息科学与计算科学中, 一个著名的问题就是 “P 类问题是否等于 NP 类问题?”

更多的人相信 “P≠NP”, 因为有 NP-complete 类问题的存在。NP-complete 类问题是 NP 类问题中的一个超级子类, NP 类中的任何一个问题都可以在多项式时间内归约为 NP-complete 类中的问题。问题 A 可归约为 (reducible) 问题 B, 指的是可以用 B 的解法来求解 A 问题。比如, 一元一次方程 $kx + m = 0$ 就可归约为一元二次方程 $ax^2 + bx + c = 0$, 只需要取二次项的系数为 0, 后者的解法就变成了前者的解法。显然 “可归约” 是可传递的, 且归约后的问题往往具有更高的计算复杂度。也就是说, NP-complete 类问题是 NP 类问

题中计算复杂度最高的。NP-complete 类中的任何一个问题如果存在多项式时间算法, 那么就有 NP=P。不过, 这似乎令人难以置信。

NP-hard 类问题跟 NP-complete 类有点类似, NP 类中的任何问题都可以在多项式时间内归约为 NP-hard 类问题。它们的区别是, NP-complete 是 NP 的子类, 而 NP-hard 不是。因为 NP-complete 类问题可以归约为 NP-hard 类问题, 结果就导致 NP-hard 类问题比 NP-complete 类问题更困难。图 1.3 描述了 P、NP、NP-complete 和 NP-hard 四类问题 (基于 NP$\neq$P 这一假设) 的相互关系[3]。

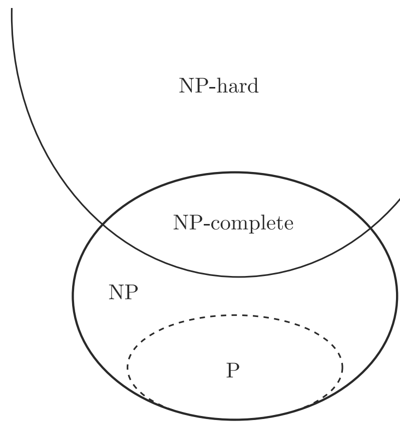

图 1.3 基于算法时间复杂度的四类问题的相互关系 (假定 NP$\neq$P)

2) 全局最优化问题可能是 NP-hard 问题

前面我们已经看到了, NP-hard 类问题是时间复杂度最高的一类问题。非常不幸的是, 本书研究的问题——非凸优化的全局最优化问题可能是一类 NP-hard 问题, 至少某些特定的全局优化问题已经被证明是 NP-hard 问题了[4-6]。

比如, 在文献 [4] 中列出了 12 大类 150 小类的全局优化问题, 它们都是 NP-hard 问题。下面给出来自文献 [6] 的两个命题。

命题 1.1 任何具有如下形式的函数的全局优化问题是一个 NP-hard 问题,

$$f(\boldsymbol{x})=a+b\sum_{i=1}^{n}\left(\frac{x_i-\alpha}{\beta-\alpha}\right)^2\left(\frac{x_i-\beta}{\beta-\alpha}\right)^2+\left[\sum_{i=1}^{n}s_i(\frac{x_i-\alpha}{\beta-\alpha})-s\right]^2 \tag{1.13}$$

其中, a,b,α,β 是实数; $x_i\in[\alpha,\beta]$; s_i,s 是整数, $i=1,2,\cdots,n$。

更进一步, 有以下结论。

命题 1.2 定义在单位单纯形上的函数 $f(\boldsymbol{x})$, 如果满足以下两个条件, 则其全局最优化问题是一个 NP-hard 问题。

- $f(x)$ 单调递增;
- $f(x)$ 满足正齐次性, 即对任何正数 a, 有 $f(a\boldsymbol{x})=af(\boldsymbol{x})$。

♠ n 维空间中的单纯形指的是由 $n+1$ 个点两两连接形成的凸多面体。比如, 1 维单纯形是线段, 2 维单纯形是三角形, 3 维单纯形是四面体。