一维非线性方程的求根

南子谦
信息与计算科学　3210104676

2022 年 7 月 4 日

摘要

未知数和一些常数不仅施行有限次代数运算，而且还要施行有限次指数、对数、三角函数等运算，这样的方程叫做超越方程。对于一般的超越方程，没有求根公式可以使用。对于代数方程，当次数大于5时无精确的求根公式。而实际问题中，欲求得方程的根时，不一定要得到根的准确值，而只需求得满足一定精度的近似值即可。本文章主要介绍求解近似根的各种方法。

方程求根问题包括三个问题，根的存在性，有根区间的确定和根的精确化。只要确定出若干个区间，使得每个区间内y=f(x)与x轴有且只有一个交点，再根据特定算法逼近根即可。

1　引言

随着科学技术的高速发展，非线性科学的应用已经涉及各个行业，例如气象资料分析、飞机，汽车及轮船的设计、石油地质、计算生物化学、航天航空领域和轨道设计、信息化援救等方面有着大量的实际问题，这些问题都要借助于非线性模型来描述，最终都可以归结为非线性方程和非线性方程组的求解问题。而对于次数大于 4 次的代数方程，它的精确解已经不能用解析方法求出，这时想要求出方程的近似解只能寻求某种数值方法，而非线性方程组的求解要更加困难。所以，无论在理论意义还是在实际应用中，运用数值方法求解非线性方程和非线性方程组都是非常重要的。

2　数学理论

2.1　定义

定义 1. 设函数$F: D \subset R^n \to R^n$,若对$\forall \xi > 0$, $\exists \delta > 0$，使得对任意的$x \in S(x_0, \delta) = |x - x_0| < \delta \subset D$都有

$$|F(x) - F(x_0)| < \xi \tag{1}$$

则称$F(x)$在点$x_0 \in int(D)$，若$F(x)$在D内每一点连续，则称$F(X)$在D内连续。

定义 2. 迭代法的定义

现在给定一个初始点z_0或者几个初始近似值$z_0, z_1, z_2, \cdots, z_r, \cdot$，其中$r$为有限个，同时保证$r$ 为正整数，可以依据某种格式产生一个有限或者无限序列

$$x_0, x_1, x_2, \cdots, x_n, x_{n+1} \cdots (n \in z). \tag{2}$$

使得生成的序列收敛于方程$f(x) = 0$一个根x^*，即

$$\lim_{x \to 0} = x^*. \tag{3}$$

此时当$n \to \inf$时，则x_n为方程的近似解，x_n无限趋近于真实解x^*。

2.2 定理

定理 1. 零点存在定理

若函数$f(x)$在闭区间$[a, b]$连续，且$f(a) \cdot f(b) < 0$，则一定存在$\xi \in (a, b)$，使$f(\xi) = 0$.

证明. 请参考数学分析I第93页。 □

定理 2. 零点存在定理的推论

若函数$f(x)$在闭区间$[a, b]$单调且连续，且$f(a) \cdot f(b) < 0$，则存在唯一的$\xi \in (a, b)$，使$f(\xi) = 0$.

3 算法

在实际情况中，对一个非线性方程，我们通常不需要求出它所有的根，根据实际条件的约束，我们希望根在一个区间内，这就为我们根的上下界提供了条件。

3.1 二分法

设$f(x)$是连续函数, 现有区间$[a, b]$, 且$f(a)f(b) < 0$, 那么取区间中点$c = \frac{a+b}{2}$, 则$[a, c], [b, c]$必有至少一个区间存在$f(x)$的根，即与$f(c)$符号相异的端点与c构成的区间内.

证明. 设$f(a) > 0$，则$f(b) < 0$,当$f(c) \neq 0$时,$f(a)f(c)$与$f(b)f(c)$异号，由定理(1)可得$[a, c], [b, c]$必有至少一个区间存在$f(x)$的根。 □

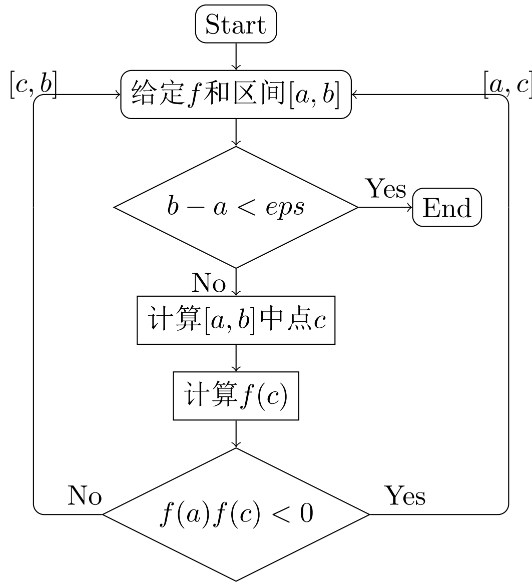

图 1: 二分法流程图

3.1.1 算法流程图

3.1.2 收敛速度

每迭代一次, 误差减半

3.2 牛顿迭代法

对于导函数f'存在的$f(x)$, 给定一个根附近的点x_0, 则由公式

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

所产生的x_1是相较于x_0更好的对根的近似.

可以看出, 每次迭代其实是在x_n处将f看成其泰勒展开的一阶近似, 并求解此线性方程得到新的解x_{n+1}.

因此, 不断重复此过程, 即不断地求

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{4}$$

即可求得根的近似值.

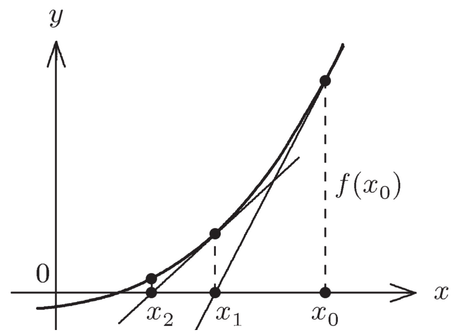

图 2: x_2比x_1更好

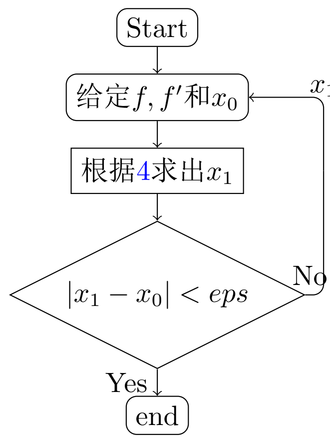

图 3: Newton法流程图

3.2.1 算法流程图

3.2.2 收敛速度

可以证明, Newton方法的收敛速度是二阶的.

4 数值算例

由上一节关于收敛速度的讨论, 可以知道Newton法在理论上比二分法收敛的更快一些. 下面通过一个高次多项式的例子来比较这两个算法实际上的好坏. $f(x) = 8x^5 + x^4 + 9x^2 + 7x + 5$

二分法:设置初始区间为$[-e, 0], eps = 10^{-5}$, 则此算法表现如下:

二分法迭代19次后达到目标精度.

iter	lower	upper	root	err(est)
1	-1.3591409,	0.0000000	-0.6795705	1.3591409
2	-1.3591409,	-0.6795705	-1.0193557	0.6795705
3	-1.0193557,	-0.6795705	-0.8494631	0.3397852
4	-1.0193557,	-0.8494631	-0.9344094	0.1698926
5	-1.0193557,	-0.9344094	-0.9768825	0.0849463
6	-1.0193557,	-0.9768825	-0.9981191	0.0424732
7	-1.0193557,	-0.9981191	-1.0087374	0.0212366
8	-1.0087374,	-0.9981191	-1.0034283	0.0106183
9	-1.0034283,	-0.9981191	-1.0007737	0.0053091
10	-1.0007737,	-0.9981191	-0.9994464	0.0026546
11	-1.0007737,	-0.9994464	-1.0001100	0.0013273
12	-1.0001100,	-0.9994464	-0.9997782	0.0006636
13	-1.0001100,	-0.9997782	-0.9999441	0.0003318
14	-1.0001100,	-0.9999441	-1.0000271	0.0001659
15	-1.0000271,	-0.9999441	-0.9999856	0.0000830
16	-1.0000271,	-0.9999856	-1.0000063	0.0000415
17	-1.0000063,	-0.9999856	-0.9999960	0.0000207
18	-1.0000063,	-0.9999960	-1.0000012	0.0000104
19	-1.0000012,	-0.9999960	-0.9999986	0.0000052

牛顿迭代法:设置初始点$x_0 = 0, eps = 10^{-5}$, 则此算法表现如下:

iter	root	err(est)
1	-0.7142857	-0.7142857
2	-1.8005532	-1.0862675
3	-1.4795259	0.3210273
4	-1.2432980	0.2362280
5	-1.0892808	0.1540172
6	-1.0164491	0.0728317
7	-1.0006718	0.0157774
8	-1.0000012	0.0006706
9	-1.0000000	0.0000012

可以看到, 牛顿法迭代9次后即达到目标精度, 显著优于二分法.

5 分析

可以看出, 不管是理论上还是实际例子中, Newton法在求解函数的根问题上都优于二分法, 不过其有可导的前提, 在应用上没有二分法广泛。(图像请看附录)

参考文献

[1] Mark Galassi. GNU Scientific Library. Online, 2021.

[2] 做你的猫. 非线性方程求根. https://zhuanlan.zhihu.com/p/144536388, 2022. [Online; accessed 1-July-2022].

[3] 管慧莹. 求解非线性方程的两种迭代算法. 硕士, 哈尔滨理工大学, 2020.

[4] 陈纪修. 数学分析（上）. 高等教育出版社, 2019.

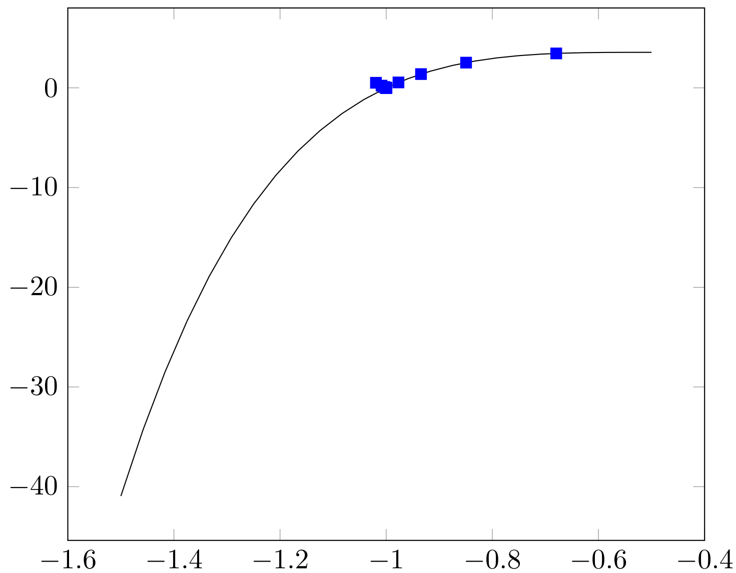

图 4: 二分法

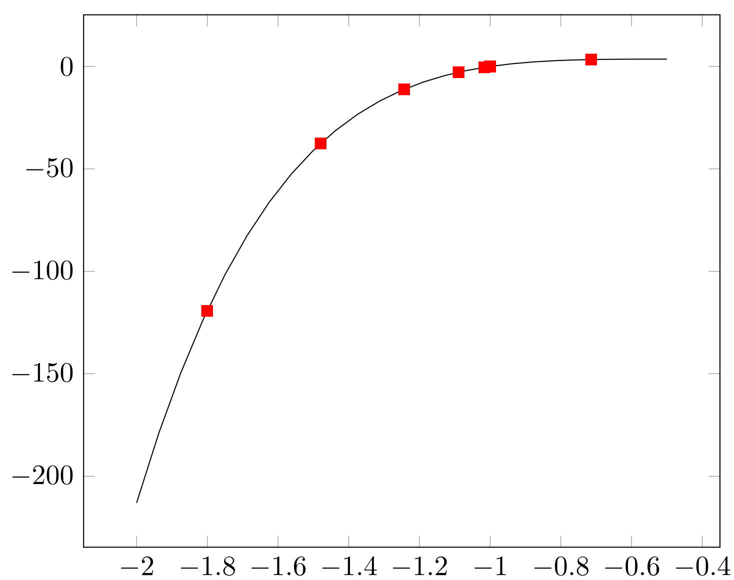

图 5: 牛顿迭代法