

数値計算 Class-2 演習

21T2166D 渡辺大樹

2023 年 4 月 20 日

1 演習内容

Class-2 の演習ではコンピューターで任意の方程式を解くためのアルゴリズム、二分法とニュートン法を C 言語で実装し、実際にいくつかの複雑な方程式について解いていった。

1.1 二分法

二分法は、解がある閉区間 $[a, b]$ (a, b は異符号) にある時、その区間の中間値 $c = \frac{a+b}{2}$ を求め、 a ないし b と異符号かどうかを評価し異符号となった方の値の間にできた閉区間でさらに同じ操作を繰り返すことで解を求めるアルゴリズムである。このアルゴリズムは中間値の定理を利用している。以下ソースコード??が C 言語で二分法を実装し、演習で利用したコードである。

ソースコード 1 bisection.c

```
1 #include <stdio.h>
2 #define _USE_MATH_DEFINES
3 #include <math.h>
4
5 double f(double x, int i){
6     //i の値に応じて適切な f(x) を呼び出す関数
7     switch (i){
8         case 0:
9             case 1:
10                 return pow(x, 4.0) - 3*x + 1;
11             case 2:
12                 return cos(x) - x;
13             case 3:
14                 return pow(M_E, x) - 1/x;
15             default:
16                 break;
17     }
```

```

18 }
19 #define MAX_ITERATIONS 100
20 double bisection(double a, double b, double eps, int i) {
21     double c;
22     int it = 0;
23     do {
24         c = (a+b)/2.0; //a,bを二分したx座標の値,cを用意
25         if (f(a, i)*f(c, i) < 0.0) { //符号が異なるとき
26             b = c;
27         }
28         else { //符号が一致するとき
29             a = c;
30         }
31         it++;
32     } while(fabs(b-a) >= eps && it <= MAX_ITERATIONS); //a,
        bの差が2e-30より小さくなるか計算回数が100回を超えるときループを抜け出す
33     c = (a+b)/2.0;
34     printf("it = %d \n", it);
35
36     return c;
37 }
38 int main() {
39     double date[4][2] = {{1.0, -1.0},{2.0, 1.0},{1.0, 0.0},{1.0,
        0.0}};
40     //各関数における解を含む閉区間の配列
41     double epsilon = 2e-30;
42     for(int i = 0; i < 4; i++){
43         double root = bisection(date[i][0],date[i][1],epsilon,i);
44         printf("a~b 閉区間の解は x=%f , f(x) = %g\n", root, f(root,i));
45     }
46
47     return 0;
48 }

```
