

实验报告一

李平治 PB19071501

2022年3月4日

I. 题目及其运行结果

1.1 题目

程序 2 下面给出美国 1920~1970 年的人口表：

年份	1920	1930	1940	1950	1960	1970
人口 (千人)	105711	123203	131669	150697	179323	203212

用表中数据构造一个 5 次 Lagrange 插值多项式, 并用此估计 1910, 1965 和 2002 年的人口. 1910 年的实际人口数约为 91772000, 请判断插值计算得到的 1965 年和 2002 年的人口数据准确性是多少?

程序 3 数据同上表, 用 Newton 插值估计:

- (1) 1965 年的人口数;
- (2) 2012 年的人口数.

1.2 结果

程序2

```
L(1910)= 31872.0
L(1965)= 193081.51171875
L(2002)= 26138.748416004702
L'(1965)= 178340.49609375
L'(2002)= -3964443.3379840525
```

程序3

```
N(1965)= 193081.51171875
N(2012)= 26138.748415999813
```

II. 使用算法

程序2和程序3分别使用了Lagrange插值和Newton插值法，并用Python实现。

Lagrange插值:

$$L_n(x) = \sum_{i=1}^n [f(x_i) \prod_{0 \leq j \leq n, j \neq i} \frac{x - x_j}{x_i - x_j}]$$

Newton插值:

$$N_n(x) = f[x_0] + \sum_{k=1}^n f[x_0, x_1, \dots, x_k](x - x_0)(x - x_1) \dots (x - x_{k-1})$$

III. 结果分析

$L(x)$ 和 $N(x)$ 分别为直接使用表中点插值获得的插值多项式值，无需赘述。

$$\begin{aligned} L(1910) &= 31872.0 \\ L(1965) &= 193081.51171875 \\ L(2002) &= 26138.748416004702 \\ N(1965) &= 193081.51171875 \\ N(2012) &= 26138.748415999813 \end{aligned}$$

$L'(x)$ 为使用1910, 1920, 1930, 1940, 1950, 1960这六组数据插值所得值。

$$L'(1965) = 178340.49609375$$

$$L'(2002) = -3964443.3379840525$$

由此得 $R(x) \approx f(x) - L(x) \approx \frac{x-x_0}{x_0-x_n+1}(L(x) - L'(x))$

代入得 $R(1965) \approx -13512.5$, $R(2002) \approx -6222600$

这表明Lagrange插值拟合效果不佳

附录

程序2和程序3的Python代码分别如下

```
1  import numpy as np
2
3  actual_points = {
4      1920: 105711,
5      1930: 123203,
6      1940: 131669,
7      1950: 150697,
8      1960: 179323,
9      1970: 203212
10 }
11
12 additional_point = {
13     1910: 91772
14 }
15
16 def lagrange_interp(x, inter_points):
17     res = 0
18     for point in inter_points:
19         tmp = 1
20         for _point in inter_points:
21             if _point is not point:
22                 tmp = tmp * (x - _point) / (point - _point)
23         res = res + tmp * inter_points[point]
24     return res
25
26
27 print('L(1910)=', lagrange_interp(1910, actual_points))
28 print('L(1965)=', lagrange_interp(1965, actual_points))
29 print('L(2002)=', lagrange_interp(2002, actual_points))
30
31 actual_points.popitem()
32 print('L\'(1965)=', lagrange_interp(1965, {**actual_points, **additional_point}))
33 print('L\'(2002)=', lagrange_interp(2002, {**actual_points, **additional_point}))
34
```

```
1 import numpy as np
2
3 points = {
4     1920: 105711,
5     1930: 123203,
6     1940: 131669,
7     1950: 150697,
8     1960: 179323,
9     1970: 203212
10 }
11
12
13 def newton_interp(u, inter_points):
14     x, g = [], []
15     n = len(inter_points) - 1
16     for _x in inter_points:
17         x.append(_x)
18         g.append(inter_points[_x])
19     for k in range(1, n + 1):
20         for j in range(n, k - 1, -1):
21             g[j] = (g[j] - g[j - 1]) / (x[j] - x[j - k])
22     t = 1
23     newton = g[0]
24     for k in range(1, n + 1):
25         t = t * (u - x[k - 1])
26         newton = newton + t * g[k]
27     return newton
28
29
30 print('N(1965)=', newton_interp(1965, points))
31 print('N(2012)=', newton_interp(2012, points))
32
```