Computing Method - Programming 1

Zhixin Zhang, 3210106357

1 问题

函数 $f(x)=\frac{1}{1+25x^2}, x\in[-1,1]$，利用下列条件做插值逼近，并与函数 $f(x)$ 的图像进行比较．$(n=10)$

(a) 用等距节点 $x_i=-1+\frac{2}{n}i, i=0,1,2,\cdots,n$，试建立 n 次 Largrange 插值多项式和 Newtown 插值多项式，绘出插值多项式的图像；

(b) 用节点 $x_i=\cos(\frac{2i+1}{42}\pi), i=0,1,\cdots,20$，绘出 20 次 Lagrange 插值多项式的图像；

(c) 用等距节点 $x_i=-1+\frac{2}{10}i, i=0,1,2,\cdots,10$，绘出它的分段线性插值函数的图像；

(d) 用等距节点 $x_i=-1+\frac{2}{10}i, i=0,1,2,\cdots,10$，绘出它的分段三次 Hermite 插值函数的图像．

2 公式与算法

(a) Lagrange 插值公式：$\varphi_n(x)=\sum_{j=0}^{n}y_j\frac{\prod_{i\neq j,i=0}^{n}(x-x_i)}{\prod_{i\neq j,i=0}^{n}(x_j-x_i)}$.

Newton 插值公式：$\varphi_n(x)=f(x_0)+\sum_{i=1}^{n}f[x_1,\ldots,x_i]\prod_{j=0}^{i-1}(x-x_j)$.

(b) 使用不均匀的节点进行 Lagrange 插值，公式同 (a).

(c) 分段线性插值，直接对于相邻两个节点进行线性拟合．

(d) 分段三次 Hermite 插值：

$$
\begin{array}{c|llll}
x_0 & y_0 & & & \\
x_0 & y_0 & y_0' & & \\
x_1 & y_1 & f[x_0,x_1] & \frac{y_0'-f[x_0,x_1]}{x_0-x_1} & \\
x_1 & y_1 & y_1' & \frac{y_1'-f[x_0,x_1]}{x_1-x_0} & \frac{y_1'+y_0'-2f[x_0,x_1]}{(x_1-x_0)^2}
\end{array}
$$

所以，

$$f(x)=\frac{y_1'+y_0'-2f[x_0,x_1]}{(x_1-x_0)^2}(x-x_0)^2(x-x_1)+\frac{y_0'-f[x_0,x_1]}{x_0-x_1}(x-x_0)^2+y_0'(x-x_0)+y_0$$

3 程序

3.1 调用接口

```
f = @(x) 1./(1 + 25*x.^2);
xa = ((1:11)-6)./5;
ya = f(xa);
y1 = -50 * xa ./ (1 + 25 * xa.^2).^2;
xb = cos((2*(0:20) + 1)*pi/42);
yb = f(xb);
ans11 = @(x) lagrange(xa, ya, x);
Difftable = diffTable(xa, ya);
ans12 = @(x) newton(xa, Difftable, x);
ans2 = @(x) lagrange(xb, yb, x);
ans3 = @(x) linear(xa, ya, x);
ans4 = @(x) hermite(xa, ya, y1, x);
```

3.2 绘图

```
x_ = ((1:1001)-501)./500;
ploto = f(x_);
plot11 = ans11(x_);
plot12 = ans12(x_);
plot2 = ans2(x_);
plot3 = ans3(x_);
plot4 = ans4(x_);
figure;
subplot(2,2,1);
plot(x_, ploto, 'b--', 'DisplayName', 'Original');
hold on;
plot(x_, plot11, 'k-', 'DisplayName', 'Lagrange-1');
plot(x_, plot12, 'r-.', 'DisplayName', 'Newton');
legend('show');
title('(a) Lagrange vs Newton');
xlabel('x');
ylabel('y');
subplot(2,2,2);
plot(x_, ploto, 'b--', 'DisplayName', 'Original');
hold on;
plot(x_, plot2, 'k-', 'DisplayName', 'Lagrange-2');
legend('show');
title('(b) Lagrange');
xlabel('x');
ylabel('y');
subplot(2,2,3);
plot(x_, ploto, 'b--', 'DisplayName', 'Original');
hold on;
plot(x_, plot3, 'k-', 'DisplayName', 'Linear');
legend('show');
title('(c) Linear');
xlabel('x');
ylabel('y');
subplot(2,2,4);
plot(x_, ploto, 'b--', 'DisplayName', 'Original');
hold on;
plot(x_, plot4, 'k-', 'DisplayName', 'Hermite');
legend('show');
title('(d) Hermite');
xlabel('x');
ylabel('y');
```

3.3 求解部分

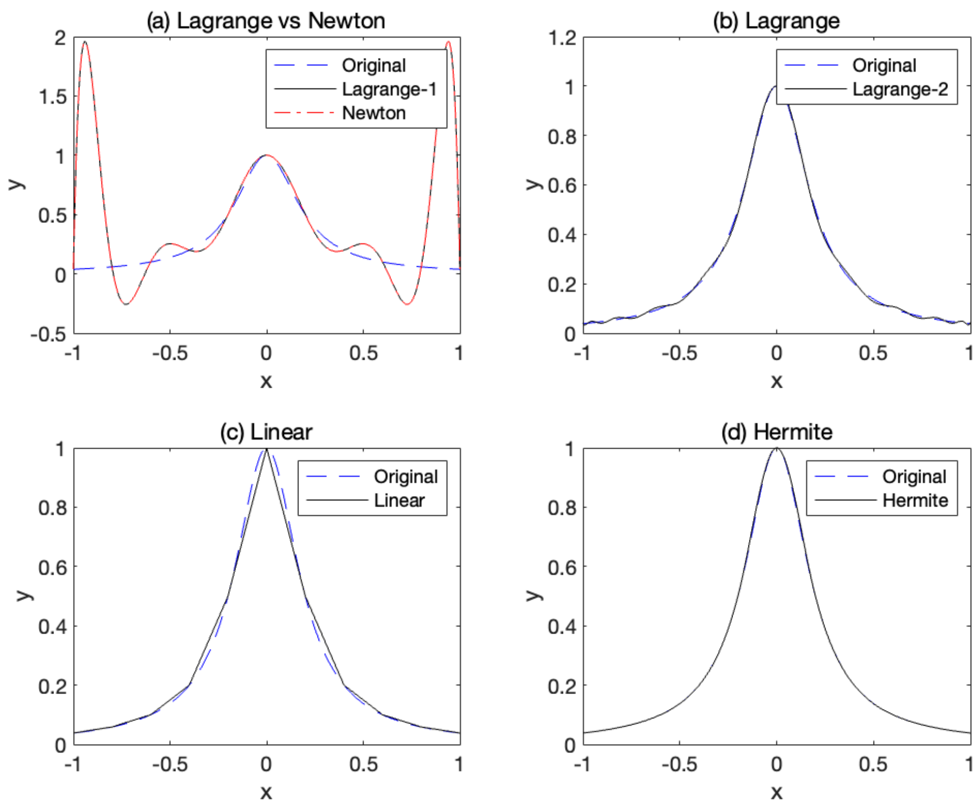
```
function P = lagrange(X, Y, x)
    n = length(X);
    L = ones(n, length(x));
    for i = 1:n
        for j = 1:n
            if i ~= j
                L(i,:) = L(i,:).*(x-X(j))./(X(i)-X(j));
```

```
            end
        end
    end
    P = Y*L;
end
function P = newton(X, Table, x)
    n = length(X);
    P(length(x)) = 0;
    for ii = 1:length(x)
        num = 1;
        P(ii) = 0;
        for i = 1:n
            P(ii) = P(ii) + num * Table(i);
            num = num * (x(ii) - X(i));
        end
    end
end
function table = diffTable(X, Y)
    n = length(X);
    table(n) = 0;
    for i = 1:n
        table(i) = Y(i);
    end
    for j = 2:n
        for i = n+j-(j:n)
            table(i) = (table(i) - table(i-1)) / (X(i)-X(i-j+1));
        end
    end
end
function P = linear(X, Y, x)
    P(length(x)) = 0;
    for i = 1:length(x)
        for j = 1:length(X)-1
            if x(i) >= X(j) && x(i) <= X(j+1)
                P(i) = Y(j) + (Y(j+1) - Y(j)) / (X(j+1) - X(j)) * (x(i) - X(j));
                break;
            end
        end
    end
end
function P = hermite(X, Y, Y1, x)
    P(length(x)) = 0;
    for i = 1:length(x)
        for j = 1:length(X)-1
            if x(i) >= X(j) && x(i) <= X(j+1)
                P(i) = hermite_(X(j), X(j+1), Y(j), Y(j+1), Y1(j), Y1(j+1), x(i));
            end
        end
    end
end
function P = hermite_(x0, x1, y0, y1, y0p, y1p, x)
    P = 0;
    P = P + y0;
    P = P + (x - x0) * y0p;
```

```
    fx0x1 = (y1 - y0) / (x1 - x0);
    P = P + (x - x0) * (x - x0) * (y0p - fx0x1) / (x0 - x1);
    P = P + (x - x0) * (x - x0) * (x - x1) * (y1p + y0p - 2 * fx0x1) / (x1 - x0) / (x1 - x0);
end
```

4 数据与结果

输入函数为 $f(x) = \frac{1}{1+25x^2}, x \in [-1, 1]$，四问的求解结果如下：



5 结论

(a) 对于相同的输入，Lagrange 和 Newton 的求解结果完全相同，这符合插值的唯一性．插值结果在两个端点处出现了明显的振荡，产生 Runge 现象．

(b) 使用 $\cos(\frac{2i+1}{42}\pi), i = 0, 1, \cdots, 20$ 作为插值节点，插值点的数量相比之前变多了，并且在两个端点处的插值点更加密集，最后结果在 $[-0.5, 0.5]$ 上的拟合结果明显更好，且在两侧的振荡也减小了许多．

(c) 线性插值结果为原函数的若干条割线相连接形成的分段函数．

(d) 两点三次 Hermite 插值结果的拟合效果最好．