SI 211: Numerical Analysis

Homework 1

朱佳会_hw1_2018233141

Problem 1.

Solution:

(a) When x = π, f(π) = 0, so the numerical approximation error is 10^{-13}. Here is the program in Julia.

```
julia> function f(x)
           return (sin(1e+4*x))/x;
           end
f (generic function with 1 method)

julia> f(pi)
-1.5459749480948014e-13
```

(b) The numerical approximation error is 10^{-2}.

According to the form of a factorable function $f(x)=\frac{\sin(10^4 x)}{x}$, when x = 10^{-10}:

a_0 = x,

a_1 = $10^4 * a_0$,

a_2 = $\sin(a_1)$,

a_3 = a_2 / a_0,

f(x) = a_3.

In the worst case, the numerical errors associated with evaluating the atom operators $\Phi_1, \ldots, \Phi_N$ may add up and lead to a potentially large evaluation error Δa_N:

$\Delta a_0 \approx$ eps

$$\Delta a_1 \approx \left|\frac{\partial_{\phi_1}}{\partial_{a_0}}(a_0)\right| * \Delta a_0 + eps = 10^4 * eps + eps$$

$$\Delta a_2 \approx \left|\frac{\partial_{\phi_2}}{\partial_{a_0}}(a_0, a_1)\right| * \Delta a_0 + \left|\frac{\partial_{\phi_2}}{\partial_{a_1}}(a_0, a_1)\right| * \Delta a_1 + eps = \cos(10^{-6}) * (10^4 * eps + eps) + eps$$

$$\Delta a_3 \approx \left|\frac{\partial_{\phi_3}}{\partial_{a_0}}(a_0, a_1, a_2)\right| * \Delta a_0 + \left|\frac{\partial_{\phi_3}}{\partial_{a_1}}(a_0, a_1, a_2)\right| * \Delta a_1 + \left|\frac{\partial_{\phi_3}}{\partial_{a_2}}(a_0, a_1, a_2)\right| * \Delta a_2 + eps$$

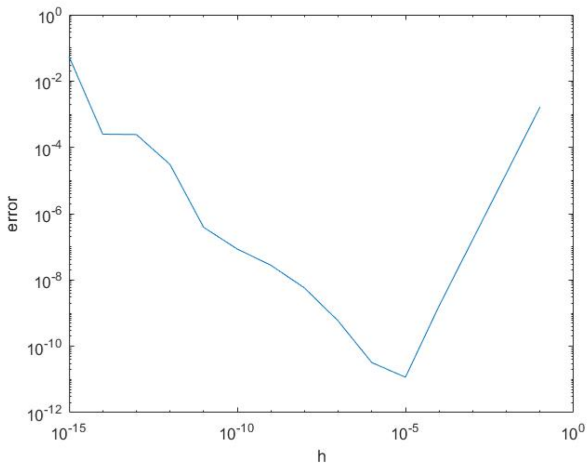
$$= -\frac{\sin(10^{-6})}{10^8} * eps + 10^{10} * \cos(10^{-6}) * (10^4 * eps + eps) + eps$$

$$\approx 10^{-2}$$

Problem 2.

Solution:

Here is the program in Julia by using the above routine diff. and the plot of the numerical differentiation error.



From the plot, when h ≈ 10^{-5}, the numerical differentiation error is the minimum.

```
julia> function f(x)
           return exp(x)
           end
f (generic function with 1 method)

julia> function diffCentral(f,x,h)
           return (f(x+h)-f(x-h))/(2*h)
           end
diffCentral (generic function with 1 method)

julia> x = 0.0
0.0

julia> function error(h)
           return abs(diffCentral(f,x,h)-1.0);
           end
error (generic function with 1 method)

julia> error(1e-1)
0.0016675001984409743

julia> error(1e-2)
1.6666749992122476e-5

julia> error(1e-3)
1.6666668134490692e-7
```

```
julia> error(1e-4)
1.6668897373506297e-9

julia> error(1e-5)
1.2102319146833906e-11

julia> error(1e-6)
2.6755486715046572e-11

julia> error(1e-7)
5.26355847796367e-10

julia> error(1e-8)
6.07747097092215e-9

julia> error(1e-9)
2.7229219767832546e-8

julia> error(1e-10)
8.274037099909037e-8

julia> error(1e-11)
8.274037099909037e-8

julia> error(1e-12)
3.3389431109753787e-5

julia> error(1e-13)
0.00024416632504653535

julia> error(1e-14)
0.0007992778373591136

julia> error(1e-15)
0.05471187339389871
```

Problem 3.

Solution:

(a)The corresponding algorithm for evaluating the derivative of f (x) using the forward mode of algorithmic differentiation (AD) is:

$b_0 = 1,$

$b_1 = -b_0*\sin(a_0),$

$b_2 = b_1*a_1 + a_1*b_1,$

$b_3 = b_1*\cos(a_0),$

$b_4 = b_2 * a_3 + a_2 * b_3$,
$f'(x) = b_4$

(b)The order of magnitude of the numerical error is zero. Theoretically, there is no error when calculating the function because of x = 0.
Here is the program in Julia. From the result, the numerical error is zero.

```
julia> import Base.*

julia> import Base.sin

julia> import Base.cos

julia> mutable struct ADV
           a
           b
           end

julia> function *(A::ADV,B::ADV)
           return ADV(A.a*B.a,A.b*B.a+A.a*B.b);
           end
* (generic function with 344 methods)

julia> function sin(A::ADV)
           return ADV(sin(A.a),cos(A.a)*A.b);
           end
sin (generic function with 13 methods)

julia> function cos(A::ADV)
           return ADV(cos(A.a),-sin(A.a)*A.b);
           end
cos (generic function with 13 methods)

julia> function f(x)
           return (sin(cos(x)))*(cos(x*x));
           end
f (generic function with 1 method)

julia> x = ADV(0.0,1.0);

julia> f(x)
ADV(0.8414709848078965, -0.0)
```