第 42 卷第 3 期 数学的实践与认识 Vol.42, No.3
2012年 2 月 MATHEMATICS IN PRACTICE AND THEORY Feb., 2012

基于前向差分的一阶数值微分公式验证与实践

张雨浓, 侯占伟, 郭东生

(中山大学 信息科学与技术学院, 广东 中山 510006)

摘 要: 根据多项式插值理论, 可以通过构造相应的插值多项式来逼近未知的目标函数, 再进一步求一阶导数, 从而得到该目标函数的一阶数值微分公式. 对于此数值微分公式, 探讨基于前向差分的未知目标函数的多点一阶微分近似公式; 即, 等间距情况下的二至十六个数据点的前向差分公式. 计算机数值实验进一步验证与表明, 该用于未知目标函数一阶数值微分的多点公式可以取得较高的计算精度.

关键词: 未知目标函数; 插值多项式; 一阶数值微分公式; 前向差分公式; 计算精度

1 引言

数值微分方法是根据离散的采样数据点及其对应的函数值, 来逼近未知目标函数在某点处的导数值的方法 [1]. 数值微分的方法广泛应用于科学研究和工程实践等领域 [2-5], 比如可用于求解常微分方程组 (ODE) 和偏微分方程组 (PDE) 等工程系统中常用的问题. 对于未知的映射关系 (即, 仅知道其离散的采样数据点及对应的函数值), 计算数值微分的方法通常有: 多项式插值法 (如 Lagrange 插值法、Newton 插值法、Chebyshev 插值法、样条插值法等)、有限差分法、正则化方法、待定系数法等 [6-10]. 而对于有着广泛应用背景和实用性的有限差分类型的方法, 人们对其研究兴趣与日俱增 [7-8]. 值得指出的是, 有限差分公式包括 1) 利用未来数据的前向差分、2) 利用历史数据的后向差分和 3) 历史和未来数据均用的中间差分三种形式 [6]. 考虑到一阶前向差分公式在有关研究工作 (比如神经网络离散化及其数字电路硬件实现) 中的重要性 [11-13], 本文着重探讨并实验二至十六个数据点的一阶导数的等间距前向差分公式.

总的来说, 基于文献 [8] 所给出的未知目标函数的一阶数值微分公式, 代入相应的数据点值, 我们即可得到原未知目标函数的一阶等间距前向差分公式. 目前有文献提供了二、三及五个数据点的前向差分公式 [6], 然而实际应用中往往要求达到更高的估计精度而需要用到更多数据点的前向差分公式. 鉴于其它多点差分公式尚未明确且完备地给出, 为了后续研究的需要, 本文进一步明确和完备地给出二至十六个数据点的等间距前向差分公式及其数值实践与验证, 以方便理论研究者和工程应用者在此基础上的各种尝试与探讨.

2 一阶前向差分公式

基于多项式插值理论, 利用采样数据点以及相应的函数值, 我们可以通过构造插值多项

收稿日期: 2010-07-22
资助项目: 国家自然科学基金 (60935001, 60775050); 中山大学实验室开放基金 (KF200937)

式 $P(x)$ 来近似求得一个未知目标函数 $f(x)$ 的逼近表达式, 即 $f(x)\approx P(x)$; 进而, 对 $P(x)$ 求一阶导数便可近似得到未知目标函数 $f(x)$ 的一阶数值微分公式, 即

$$f'(x)\approx P'(x)^{[6,11]}$$

另外, 在实际应用中, 数字离散系统往往是等间距地采集数据 [14], 也即

$$x_{i+1}-x_i=h\ (i=0,1,2,\cdots,n)$$

其中 $h>0$ 为采样间隔常数, 为此本文将主要探讨等间距前向差分公式 (即, 通项公式及二至十六点的特定差分公式).

定义 [6,8] 设未知目标函数 $f(x)$ 在离散采样数据点 x_i 处的函数值为 $f(x_i),i=0,1,2,\cdots,n$. 若存在 n 次多项式 $P_n(x)=a_0+a_1x+\cdots+a_nx^n$ 满足条件 $P_n(x_i)=f(x_i)$, 则称 $P_n(x)$ 为未知目标函数 $f(x)$ 的 n 次插值多项式.

引理 [6,8] 如果 $x_0,x_1,\cdots,x_n$ 是采样区间 $[a,b]$ 上 $(n+1)$ 个互不相同的数据点, 且未知目标函数 $f(x)$ 的 $(n+1)$ 阶导数连续 (记 $f\in C^{n+1}[a,b]$), 则对任一 $x\in[a,b]$, 存在 $\xi\in(a,b)$, 使得

$$f(x)=P_n(x)+R_{n+1}(x)$$

其中 $P_n(x)$ 为未知目标函数 $f(x)$ 的插值多项式, $R_{n+1}(x)$ 为余项且

$$R_{n+1}(x):=f^{(n+1)}(\xi)(x-x_0)(x-x_1)\cdots(x-x_n)/(n+1)!$$

根据上述定义和引理, 我们可以构造一个 n 次插值多项式 $P_n(x)$ 来逼近未知目标函数 $f(x)$; 进而对该多项式 $P_n(x)$ 求一阶导数, 便可近似求得未知目标函数 $f(x)$ 的一阶数值微分公式, 也即,

$$f'(x)=P_n'(x)+R_{n+1}'(x)$$

其中 $R_{n+1}'(x)$ 为相应余项的一阶导数项. 特别值得指出的是, 当各数据点为等间距分布时, 可有如下 $f(x)$ 的一阶数值微分公式 [7–8][其中 $C_n^i:=n!/((n-i)!i!)$ 且 $\xi\in(a,b)$]:

$$f'(x_i)=\frac{1}{h}\sum_{\substack{j=0\\ j\neq i}}^{n}\frac{(-1)^{i-j}\frac{i!(n-i)!}{j!(n-j)!}f(x_j)+f(x_i)}{(i-j)}+\frac{(-1)^{n-i}h^nf^{(n+1)}(\xi)}{(n+1)C_n^i}\tag{1}$$

值得指出的是, 对于 $f(x)$ 的一阶微分公式 (1), 当 $i=0$ 时, 忽略余项 (即公式 (1) 最右端项), 可以近似得到未知目标函数 $f(x)$ 一阶导数的等间距前向差分估计公式:

$$f'(x_0)\approx\frac{1}{h}\sum_{j=1}^{n}\frac{(-1)^{-j}C_n^jf(x_j)+f(x_0)}{-j}\tag{2}$$

在公式 (2) 中, 分别令 $n=1,2,\cdots,15$, 便可推导出二至十六个数据点的等间距前向差分公式. 其中, 表 1 列出了二至八个数据点的等间距前向差分公式, 而九至十六个数据点的等间距前向差分公式在附录中列出, 以方便其他研究者和应用者的各种尝试与探讨.

3 计算机验证与实践

为验证上述等间距前向差分公式的计算精度与有效性, 我们分别选取目标函数 $f(x)=x^2[\mathrm{e}^{-x}\sin(x)+x]$ 和 $f(x)=\mathrm{e}^x$ 来进行计算机数值实验, 求目标函数 $f(x)$ 在 $x=0.5$ 处的导

数值. 在计算机实验中, 考虑不同的步长值 h(选取了 12 个不同的步长值), 使用二至八个数据点的前向差分公式进行计算, 实验结果的计算误差情况如表 2 和表 3 所示.

表 1　二至八个数据点的等间距前向差分公式

数据点数	等间距前向差分公式
二	$f'(x_0)\approx(f(x_1)-f(x_0))/h$
三	$f'(x_0)\approx(-f(x_2)+4f(x_1)-3f(x_0))/2h$
四	$f'(x_0)\approx(2f(x_3)-9f(x_2)+18f(x_1)-11f(x_0))/6h$
五	$f'(x_0)\approx(-3f(x_4)+16f(x_3)-36f(x_2)+48f(x_1)-25f(x_0))/12h$
六	$f'(x_0)\approx(12f(x_5)-75f(x_4)+200f(x_3)-300f(x_2)+300f(x_1)$ $-137f(x_0))/60h$
七	$f'(x_0)\approx(-10f(x_6)+72f(x_5)-225f(x_4)+400f(x_3)-450f(x_2)$ $+360f(x_1)-147f(x_0))/60h$
八	$f'(x_0)\approx(60f(x_7)-490f(x_6)+1764f(x_5)-3675f(x_4)+4900f(x_3)$ $-4410f(x_2)+2940f(x_1)-1089f(x_0))/420h$

表 2　目标函数 $f(x)=x^2[\mathrm{e}^{-x}\sin(x)+x]$ 在使用二至八个数据点的前向差分公式时的一阶导数计算误差

h	二	三	四	五	六	七	八
0.5	1.12	3.32×10^{-1}	6.55×10^{-2}	2.32×10^{-2}	5.61×10^{-2}	4.90×10^{-2}	2.77×10^{-2}
0.1	1.97×10^{-1}	1.42×10^{-2}	8.97×10^{-4}	4.05×10^{-4}	6.41×10^{-5}	3.21×10^{-6}	9.57×10^{-7}
0.05	9.69×10^{-2}	3.71×10^{-3}	1.55×10^{-4}	2.94×10^{-5}	2.08×10^{-6}	1.87×10^{-8}	1.23×10^{-8}
0.01	1.91×10^{-2}	1.54×10^{-4}	1.54×10^{-6}	5.24×10^{-8}	6.64×10^{-10}	1.07×10^{-12}	2.57×10^{-13}
0.005	9.52×10^{-3}	3.87×10^{-5}	1.98×10^{-7}	3.32×10^{-9}	2.07×10^{-11}	4.66×10^{-14}	1.89×10^{-14}
0.001	1.90×10^{-3}	1.55×10^{-6}	1.61×10^{-9}	5.39×10^{-12}	1.60×10^{-14}	1.78×10^{-13}	2.93×10^{-13}
5×10^{-4}	9.50×10^{-4}	3.89×10^{-7}	2.02×10^{-10}	6.34×10^{-13}	1.27×10^{-12}	3.69×10^{-14}	2.77×10^{-12}
1×10^{-4}	1.90×10^{-4}	1.56×10^{-8}	9.53×10^{-13}	1.41×10^{-12}	2.27×10^{-13}	1.30×10^{-12}	8.52×10^{-12}
5×10^{-5}	9.50×10^{-5}	3.89×10^{-9}	3.87×10^{-12}	6.23×10^{-12}	8.86×10^{-12}	6.09×10^{-12}	6.30×10^{-12}
1×10^{-5}	1.90×10^{-5}	1.46×10^{-10}	1.30×10^{-11}	1.37×10^{-11}	5.12×10^{-11}	4.08×10^{-11}	1.27×10^{-10}
5×10^{-6}	9.50×10^{-6}	4.08×10^{-11}	9.18×10^{-12}	4.22×10^{-11}	1.30×10^{-10}	2.03×10^{-11}	3.42×10^{-11}
1×10^{-6}	1.90×10^{-6}	2.41×10^{-10}	7.86×10^{-11}	1.71×10^{-10}	7.72×10^{-10}	3.42×10^{-10}	7.27×10^{-10}

对表 2 各种不同情况下的计算误差的比较, 可以发现：步长 h 的大小和不同数据点数的前向差分公式对计算出的一阶导数精度都有影响. 比如, 当步长值 h 大于等于 0.005 时, 前向差分公式应选取更多数据点为宜, 这样截断误差就会减小, 计算出的一阶导数精度也更高；而当步长值 h 小于 2×10^{-5} 时, 因为舍入误差的缘故, 选取太多的数据点反而会降低差分公式的计算精度 (此时也有差分公式系数较大的原因). 总而言之, 通过优化步长值和使用合适点数的前向差分公式, 我们可以得到对目标函数一阶导数的高精度估计 (如计算误差介于 10^{-8} 至 10^{-13}). 从表 3 的计算误差也进一步可以看出, 差分公式的计算精度会随着步长值 h 以及选取的数据点数目的变化而变化, 合适选取步长值和数据点数目的前向差分公式可以使估计

值达到较高的精度.

表 3 目标函数 $f(x)=\mathrm{e}^x$ 在使用二至八个数据点的前向差分公式时的一阶导数计算误差

h	二	三	四	五	六	七	八
0.5	4.90×10^{-1}	2.03×10^{-1}	9.66×10^{-2}	4.94×10^{-2}	2.64×10^{-2}	1.46×10^{-2}	8.21×10^{-3}
0.1	8.53×10^{-2}	5.93×10^{-3}	4.65×10^{-4}	3.90×10^{-5}	3.41×10^{-6}	3.07×10^{-7}	2.82×10^{-8}
0.05	4.19×10^{-2}	1.43×10^{-3}	5.47×10^{-5}	2.24×10^{-6}	9.56×10^{-8}	4.20×10^{-9}	1.88×10^{-10}
0.01	8.27×10^{-3}	5.54×10^{-5}	4.17×10^{-7}	3.35×10^{-9}	2.82×10^{-11}	3.07×10^{-13}	2.04×10^{-14}
0.005	4.13×10^{-3}	1.38×10^{-5}	5.18×10^{-8}	2.08×10^{-10}	8.09×10^{-13}	3.68×10^{-13}	2.65×10^{-13}
0.001	8.25×10^{-4}	5.50×10^{-7}	4.12×10^{-10}	1.87×10^{-13}	4.24×10^{-13}	1.03×10^{-12}	7.98×10^{-13}
5×10^{-4}	4.12×10^{-4}	1.37×10^{-7}	5.22×10^{-11}	3.46×10^{-12}	9.09×10^{-13}	7.24×10^{-12}	3.07×10^{-12}
1×10^{-4}	8.24×10^{-5}	5.50×10^{-9}	6.46×10^{-13}	5.46×10^{-12}	1.38×10^{-11}	1.05×10^{-11}	1.91×10^{-11}
5×10^{-5}	4.12×10^{-5}	1.38×10^{-9}	2.87×10^{-12}	1.38×10^{-11}	4.91×10^{-12}	1.06×10^{-11}	1.39×10^{-10}
1×10^{-5}	8.24×10^{-6}	4.73×10^{-11}	9.17×10^{-11}	8.42×10^{-12}	8.04×10^{-11}	9.17×10^{-11}	2.83×10^{-10}
5×10^{-6}	4.12×10^{-6}	3.04×10^{-11}	1.41×10^{-10}	8.60×10^{-11}	7.74×10^{-10}	6.37×10^{-11}	9.41×10^{-10}
1×10^{-6}	8.24×10^{-7}	3.03×10^{-10}	8.08×10^{-10}	8.63×10^{-10}	2.36×10^{-9}	4.19×10^{-10}	7.05×10^{-9}

4 结论

作为科学研究及工程实践的常用工具, 一阶数值微分已广泛应用于各个科研领域: 插值法是数值微分方法中较为方便和有效的一种, 利用插值法可较容易地求得多点有限差分公式. 当前二、三及五个数据点的差分公式较为常见, 但其它数目的多点差分公式及其更细致深入的实践验证尚未在文献中明确完备地列出. 为此, 本文探讨并给出了未知目标函数一阶导数的等间距前向差分公式 (通项公式及二至十六点特定差分公式). 计算机数值实验进一步验证与表明, 该多点等间距前向差分公式可以取得较好的一阶导数估计精度.

参考文献

[1] 袁慰平, 孙志忠, 吴宏伟, 闻震初. 计算方法与实习 (第三版)[M]. 南京: 东南大学出版社, 2003.

[2] Zhang Y, Jiang D, Wang J. A recurrent neural network for solving Sylvester equation with time-varying coefficients[J]. IEEE Transactions on Neural Networks, 2002, 13(5): 1053-1063.

[3] Zhang Y, Ge S S. Design and analysis of a general recurrent neural network model for time-varying matrix inversion[J]. IEEE Transactions on Neural Networks, 2005, 16(6): 1477-1490.

[4] Zhang Y, Chen K. Comparison on Zhang neural network and gradient neural network for time-varying linear matrix equation AXB=C solving[C]//Proceedings of International Conference on Industrial Technology, Chengdu, China, April 2008.

[5] Ma W, Zhang Y, Wang J. MATLAB simulink modeling and simulation of Zhang neural networks for online time-varying Sylvester equation solving[C]. Proceedings of International Joint Conference on Neural Networks, Hong Kong, China, June 2008, pp. 286-290.

[6] Mathews J H, Fink K D. Numerical Methods Using MATLAB[M]. 北京: 电子工业出版社, 2005.

[7] Li J P. General explicit difference formulas for numerical differentiation[J]. Journal of Computational

and Applied Mathematics, 2005, 183(1): 29-52.
[8] 张雨浓, 郭东生, 徐思洪, 李海林. 未知目标函数之一阶数值微分公式验证与实践 [J]. 甘肃科学学报, 2009, 21(1): 13-18.
[9] 郑华盛, 喻德生. 求解数值微分公式及其余项的一种新方法 [J]. 科技通报, 2004, 20(2): 147-150.
[10] 孙亮. 具有三阶精度的数值微分紧致格式及其应用 [J]. 数学的实践与认识, 2003, 33(3): 64-66.
[11] Zhang Y, Ma W, Yi C, The link between Newton iteration for matrix inversion and Zhang neural network (ZNN)[C]. Proceedings of IEEE Conference on Industrial Technology, 2008, pp. 1-6.
[12] Zhang Y, Cai B, Liang M, Ma W. On the variable step-size of discrete-time Zhang neural network and Newton iteration for constant matrix inversion[C]. Proceedings of the 2nd International Symposium on Intelligent Information Technology Application, 2008, vol. 1, pp. 34-38.
[13] Zhang Y, Ma W, Cai B. From Zhang neural network to Newton iteration for matrix inversion[J]. IEEE Transactions on Circuits and Systems-I: Regular Papers, 2009, 56(7): 1405-1415.
[14] 李中华, 张雨浓. 自动控制原理与设计 [M]. 北京: 人民邮电出版社, 2007.

Verification and Practice on Forward-Difference Based First-Order Numerical-Differentiation Formulas

ZHANG Yu-nong, HOU Zhan-wei, GUO Dong-sheng

(School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510006, China)

Abstract: Based on the polynomial-interpolation theory, interpolating polynomials could be constructed to approximate an unknown target-function. Then, the first-order numerical-differentiation formulas could be obtained by differentiating the constructed interpolating polynomials. This paper investigates the approximate first-order numerical-differentiation formulas of the unknown target-function in terms of multiple sampling-nodes by using the forward-difference method. Specifically, the equally-spaced forward-difference formulas involving two to sixteen sampling-nodes are presented. Computer experimental results verify and show that relatively high computational precision could be achieved by using the presented formulas to estimate the numerical values of the first-order derivatives of unknown target-functions.

Keywords: unknown target function; interpolating polynomials; first-order numerical differentiation formulas; forward-difference formulas; computational accuracy

附录 九至十六个数据点的等间距前向差分公式

前文已列出了二至八个数据点的等间距前向差分公式, 考虑到方便其他研究者和应用者的各种尝试与探讨, 此处再列出九至十六个数据点的等间距前向差分公式, 如表 4 所示.

表 4 九至十六个数据点的等间距前向差分公式

数据点数	等间距前向差分公式
九	$f'(x_0) \approx (-105f(x_8)+960f(x_7)-3920f(x_6)+9408f(x_5)-14700f(x_4)+15680f(x_3)$ $-11760f(x_2)+6720f(x_1)-2283f(x_0))/840h$
十	$f'(x_0) \approx (280f(x_9)-2835f(x_8)+12960f(x_7)-35280f(x_6)+63504f(x_5)$ $-79380f(x_4)+70560f(x_3)-45360f(x_2)+22680f(x_1)-7129f(x_0))/2520h$
十一	$f'(x_0) \approx (-252f(x_{10})+2800f(x_9)-14175f(x_8)+43200f(x_7)-88200f(x_6)$ $+127008f(x_5)-132300f(x_4)+100800f(x_3)-56700f(x_2)+25200f(x_1)$ $-7381f(x_0))/2520h$
十二	$f'(x_0) \approx (2520f(x_{11})-30492f(x_{10})+169400f(x_9)-571725f(x_8)+1306800f(x_7)$ $-2134440f(x_6)+2561328f(x_5)-2286900f(x_4)+1524600f(x_3)-762300f(x_2)$ $+304920f(x_1)-83711f(x_0))/27720h$
十三	$f'(x_0) \approx (-2310f(x_{12})+30240f(x_{11})-182952f(x_{10})+677600f(x_9)-1715175f(x_8)$ $+3136320f(x_7)-4268880f(x_6)+4390848f(x_5)-3430350f(x_4)$ $+2032800f(x_3)-914760f(x_2)+332640f(x_1)-86021f(x_0))/27720h$
十四	$f'(x_0) \approx (27720f(x_{13})-390390f(x_{12})+2555280f(x_{11})-10306296f(x_{10})$ $+28628600f(x_9)-57972915f(x_8)+88339680f(x_7)-103062960f(x_6)$ $+92756664f(x_5)-64414350f(x_4)+34354320f(x_3)-14054040f(x_2)$ $+4684680f(x_1)-1145993f(x_0))/360360h$
十五	$f'(x_0) \approx (-25740f(x_{14})+388080f(x_{13})-2732730f(x_{12})+11924640f(x_{11})$ $-36072036f(x_{10})+80160080f(x_9)-135270135f(x_8)+176679360f(x_7)$ $-180360180f(x_6)+144288144f(x_5)-90180090f(x_4)+43723680f(x_3)$ $-16396380f(x_2)+5045040f(x_1)-1171733f(x_0))/360360h$
十六	$f'(x_0) \approx (24024f(x_{15})-386100f(x_{14})+2910600f(x_{13})-13663650f(x_{12})$ $+44717400f(x_{11})-108216108f(x_{10})+200400200f(x_9)-289864575f(x_8)$ $+331273800f(x_7)-300600300f(x_6)+216432216f(x_5)-122972850f(x_4)$ $+54654600f(x_3)-18918900f(x_2)+5405400f(x_1)-1195757f(x_0))/360360h$