華南農業大學

数值分析 课程设计

荣格现象的研究与误差的影响

评分细则

姓 名	学 号	分 工	评 价 指 标			成 绩
			形式规范 （10 分）	内容完整 结果正确 （50 分）	分析合理 有深度 （40 分）	
何唯	201430120308	独立完成				

学 院 名 称　数学与信息学院

专 业 名 称　信息与计算科学

提 交 日 期　2016 年 11 月 30 日

目　录

荣格现象的研究

一、 问题的提出

考虑在一个固定区间上用插值逼近一个函数。显然，Lagrange 插值中使用的节点越多，插值多项式的次数就越高。我们自然关心插值多项式增加时，Ln(x)是否也更加靠近被逼近的函数。荣格(Runge)给出的一个例子是极著名并富有启发性的。设区间[-5,5]上的函数：

$$\boxed{f(x)=\frac{1}{1+x^2}}$$

二、 实验内容

1. 实验初选取函数

考虑区间[-5,5]的一个等距划分，节点为：

$$x_i=-5+\frac{10}{n}i, \qquad i=0,1,2,\cdots,n$$

则拉格朗日插值多项式为：

$$\boxed{L_n(x)=\sum_{i=0}^{n}\frac{1}{1+x_i^{\,2}}a_i(x)}$$

其中的 ai(x),i=0,1,2,⋯,n 是 n 次 Lagrange 插值基函数。

2. 具体实验步骤

选择不断增大的分点数 n=2,3,⋯

1) 画出原函数 f(x)及插值多项式函数 Ln(x)在[-5,5]上的图像；
2) 给出每一次逼近的最大误差；
3) 比较并分析实验结果。
4) 选择其它函数，例如定义在区间[-5,5]上的函数：

$$g(x)=\arctan x$$

重复上述实验看其结果如何。

5) 区间[a,b]上切比雪夫(Chebychev)点的定义为

$$\boxed{x_k=\frac{b+a}{2}+\frac{b-a}{2}\cos\left(\frac{(2k-1)\pi}{2(n+1)}\right),k=1,2,\cdots,n+1}$$

以 x1,x2,⋯,xn+1 为插值节点构造上述各函数的 Lagrange 插值多项式，比较其结果。

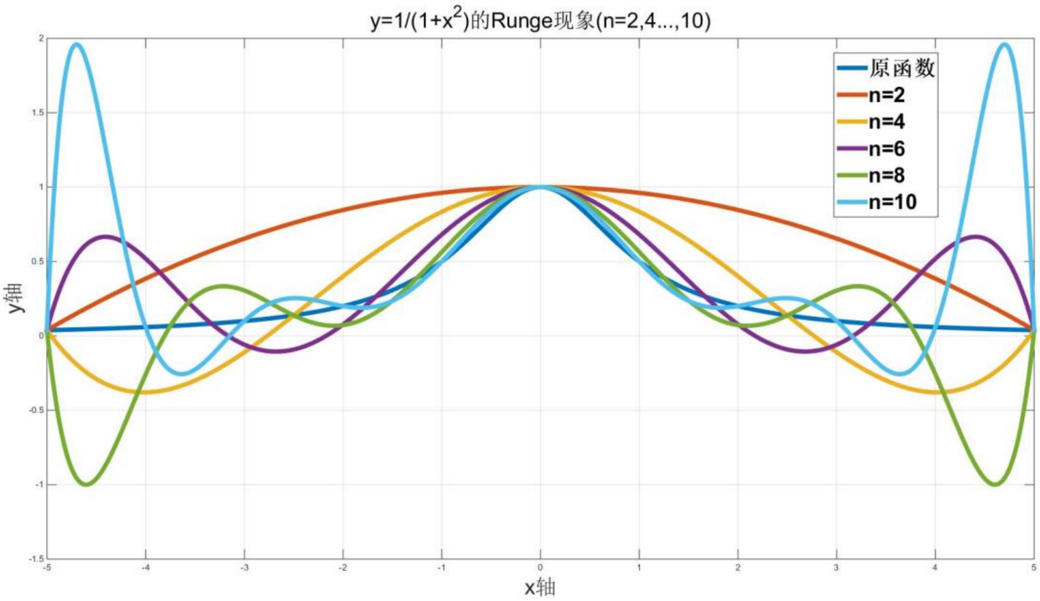
三、　实验结果及分析

1. 使用等距插值节点的荣格现象

为了使实验效果更佳明显，本课程设计将插值区间划分数 n=2,4...,10 与 n=10,12...,20 进行分开画图,并且将每次逼近的最大误差用向量的无穷范数来表示，再用一个数组 R 存储每次逼近的最大误差。

使用无穷范数代表最大误差的原因，其一是准确性，根据无穷范数的定义可知是指向量分量绝对值的最大值，因此我们只需取原函数与插值函数值的差向量就可准确表示最大误差；其二是方便程序的编写，Matlab 中直接有函数 norm 取向量的无穷范数。

具体结果如下图一至图八:

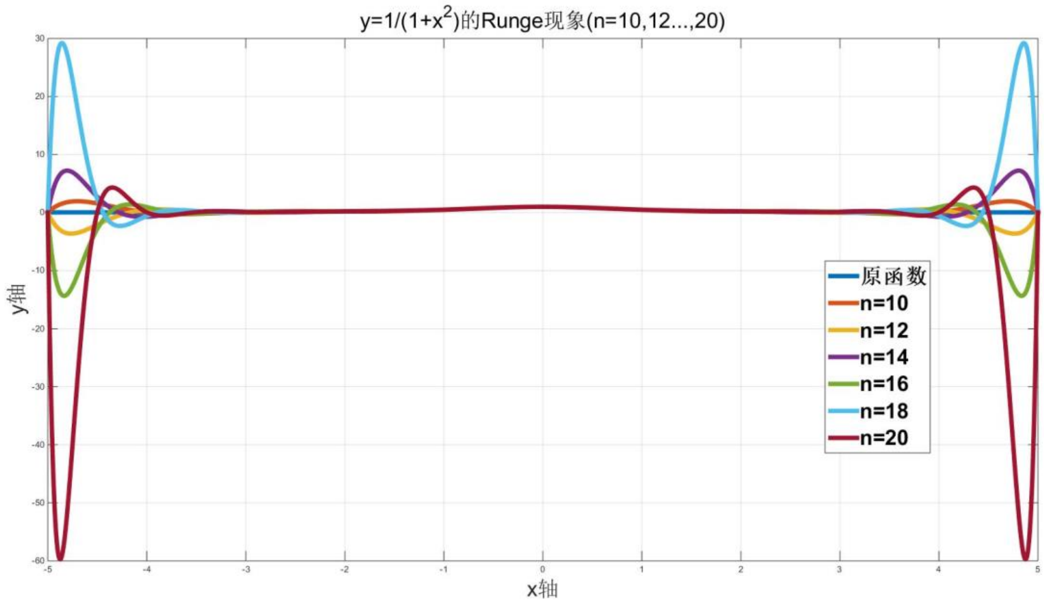


图一 y=1/1+x^2 的 Runge 现象(n=2,4...,10)

```
R =

    2.0000    0.6462
    4.0000    0.4384
    6.0000    0.6169
    8.0000    1.0452
   10.0000    1.9156
```

图二 y=1/1+x^2 的每一次逼近最大误差(n=2,4...,10)

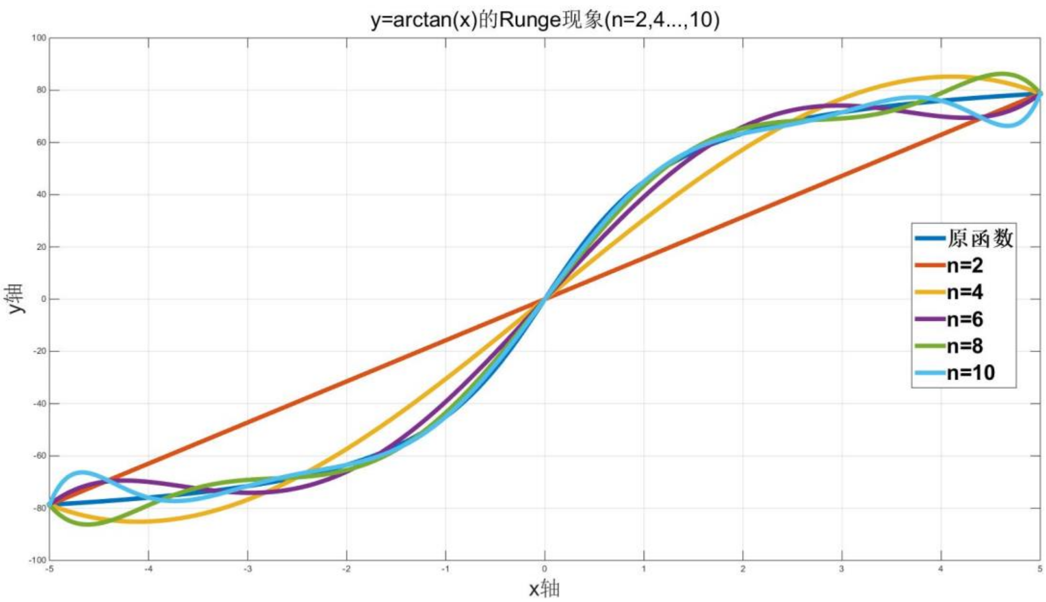


图三 y=1/1+x^2 的 Runge 现象(n=10,12...,20)

R =

10.0000	1.9156
12.0000	3.6630
14.0000	7.1921
16.0000	14.3863
18.0000	29.1856
20.0000	59.7683

图四 y=1/1+x^2 的每一次逼近最大误差(n=10,12...,20)

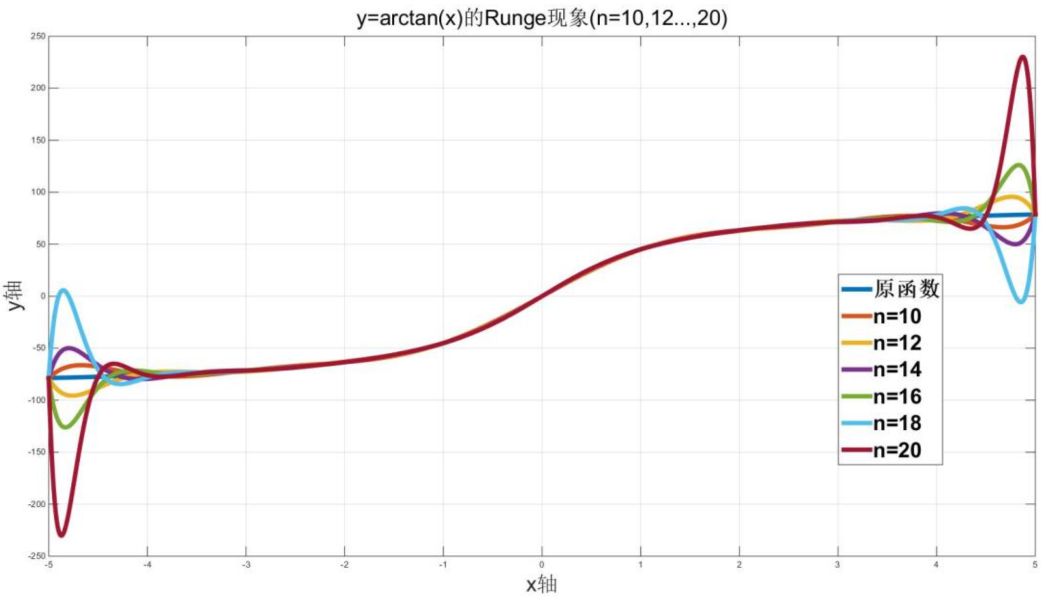


图五 y=arctan(x)的 Runge 现象(n=2,4...,10)

```
R =

    2.0000   32.8181
    4.0000   14.4966
    6.0000    7.4916
    8.0000    8.5468
   10.0000   11.6087
```

图六 y=arctan(x)的每一次逼近最大误差(n=2,4...,10)



图七 y=arctan(x)的 Runge 现象(n=10,12...,20)

```
R =

   10.0000   11.6087
   12.0000   17.4966
   14.0000   28.2532
   16.0000   47.8866
   18.0000   84.1000
   20.0000  151.9617
```

图八 y=arctan(x)的每一次逼近最大误差(n=10,12...,20)

2.等距插值节点荣格现象分析：

荣格现象是指随着插值点个数的增加(n 的增加),插值多项式的误差不但没有减少反而增大的现象。根本原因是在于拉格朗日插值余项大小不仅与插值节点的个数有关，也与函数的高阶导数有关。由上述实验结果也证明其正确性，不同的函数在不同的插值节点个数情况下，荣格现象明显程度不一样。

以函数 y=1/1+x^2 为例，随着 n 的增大，由图二和图四可知，插值多项式与原函数在区间[-5,5]内最大误差先减少(n=2,4)后增大。并且由函数图像,即图一和图三可得，该最大误差主要出现在最靠近两端点的第一个插值区间内，即端点附近插值函数震荡最为明显。说**明随着 n 的增大，最大误差有增大趋势，荣格现象更加明显。**

对比函数 y=arctan(x)，随着 n 的增大，由图六和图八可知，其最大误差也是有增大趋势，但是发生误差增大对应的节点个数不一样，发生在 n=6 的时候。**说明不同函数，具有不同的性质，所以荣格现象明显程度不一样。**

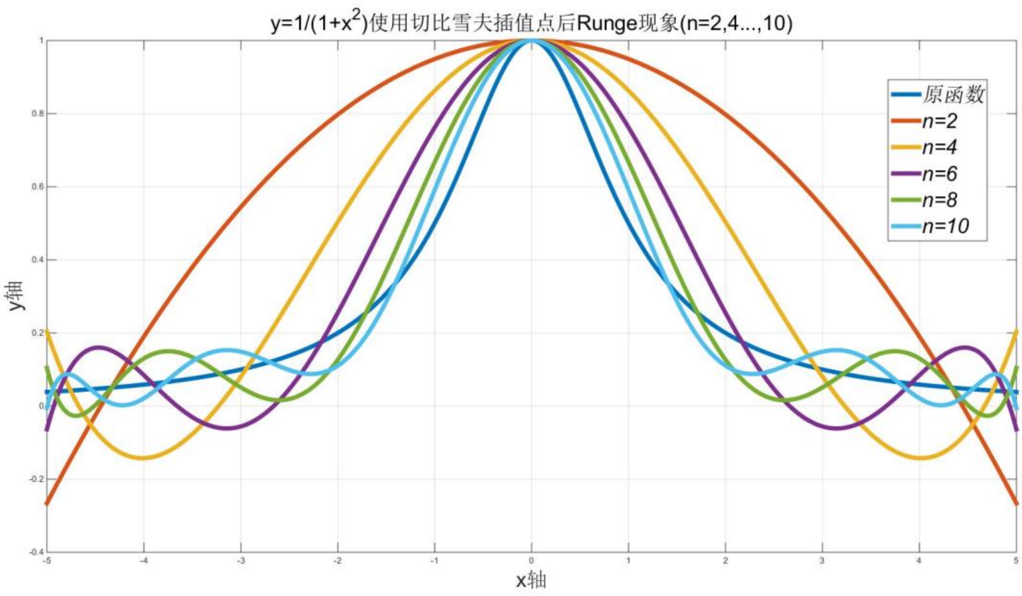
深究其原因，是因为插值余项与函数高阶导数有关，误差由导数一级一级地传播。引用 wiki，可以证明，在多项式的阶数增高时插值误差甚至会趋向无限大：

$$\lim_{n\to\infty}\left(\max_{-1\le x\le 1}|f(x)-P_n(x)|\right)=\infty.$$

这个现象表明高阶多项式通常不适合用于插值。最好使用分段低次插值或者换取插值节点的选择方法。

3.使用切比雪夫插值点后荣格现象

对于解决荣格现象,可以尝试使用切比雪夫插值点作为拉格朗日插值多项式的插值节点。同样地， 为了使实验效果更佳明显，本课程设计将 n=2,4...,10 与 n=10,12...,20 进行分开画图,并且将每次逼近的最大误差用向量的无穷范数来表示，具体结果如下图九至图十四:

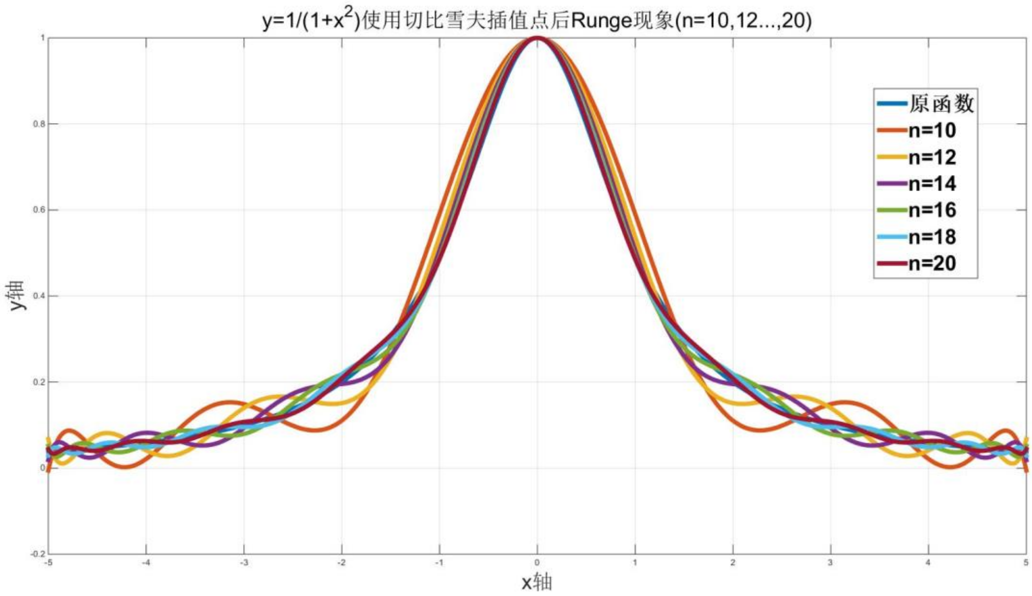


图九 y=1/1+x^2 使用 Chebychev 插值点后 Runge 现象(n=2,4...,10)

```
R =

    2.0000    0.6006
    4.0000    0.4020
    6.0000    0.2642
    8.0000    0.1708
   10.0000    0.1091
```

图十 y=1/1+x^2 的每一次逼近最大误差(n=2,4...,10)

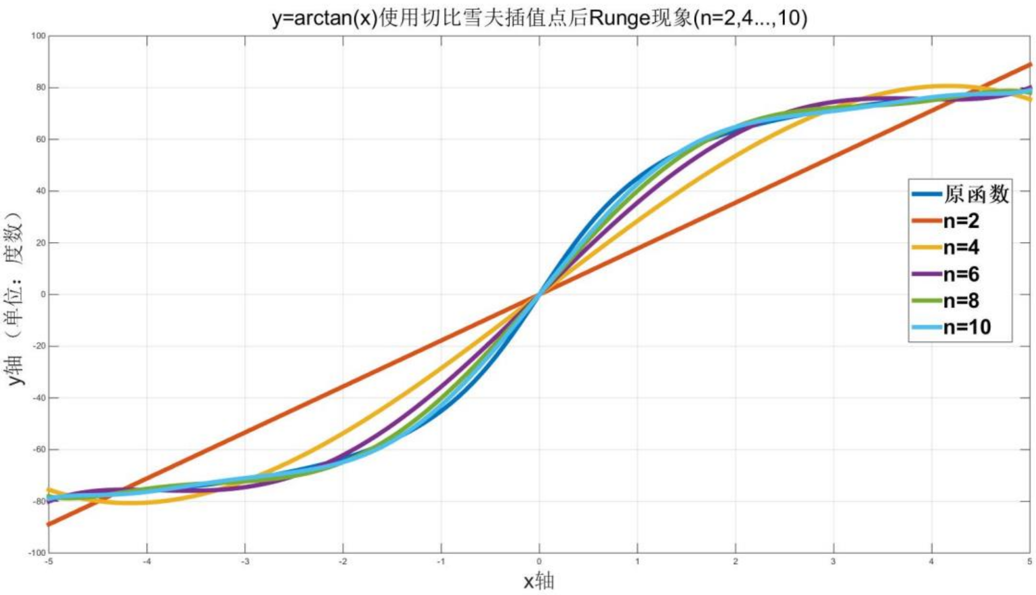


图十一 y=1/1+x^2 使用 Chebychev 插值点后 Runge 现象(n=10,12...,20)

```
R =

   10.0000    0.1091
   12.0000    0.0692
   14.0000    0.0466
   16.0000    0.0326
   18.0000    0.0225
   20.0000    0.0153
```

图十二 y=1/1+x^2 的每一次逼近最大误差(n=10,12...,20)

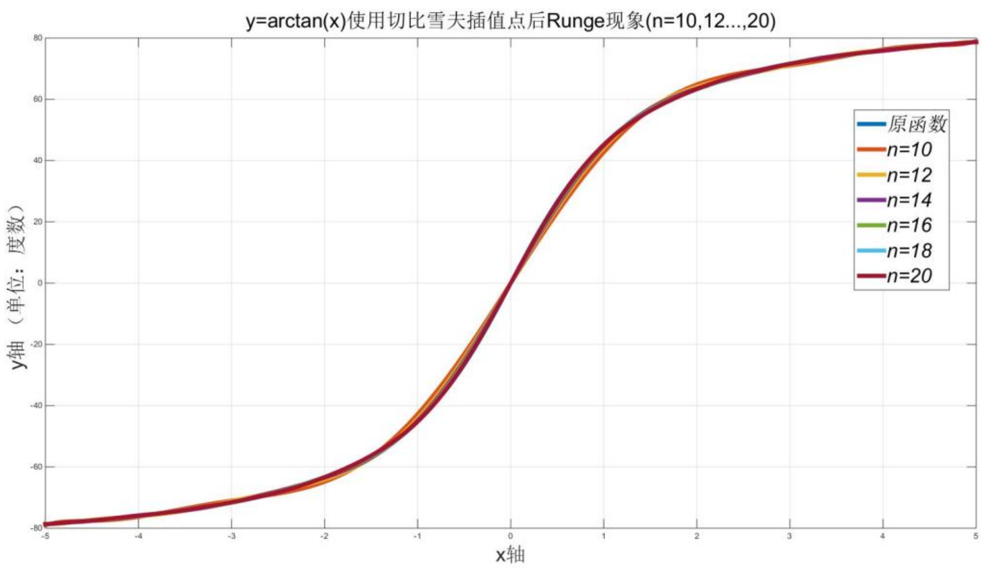


图十三 y=arctan(x)使用 Chebychev 插值点后 Runge 现象(n=2,4...,10)

R =

2.0000	29.6384
4.0000	16.5403
6.0000	9.7394
8.0000	5.8576
10.0000	3.5644

图十四 y=arctan(x)使用 Chebychev 插值点后,每一次逼近最大误差 (n=2,4...,10)



图十五 y=arctan(x)使用 Chebychev 插值点后 Runge 现象(n=10,12...,20)

```
R =

   10.0000    3.5644
   12.0000    2.1867
   14.0000    1.3499
   16.0000    0.8381
   18.0000    0.5228
   20.0000    0.3275
```

图十四 y=arctan(x)使用 Chebychev 插值点后,每一次逼近最大误差
(n=10,12...,20)

4.使用切比雪夫插值点后荣格现象分析

对比使用等距节点以及切比雪夫节点插值函数最大误差，我们可以发现，后者明显地减少了最大误差。(对比图二和图十，图四和图十二)

以函数 $y=1/1+x^2$ 和函数 y=arctan(x)为例，实验中 n=20 时，前者等距节点插值函数与原函数最大误差为 50.7683，而同样 n=20 时，切比雪夫插值函数与原函数最大误差为 0.0153；后者 n=20 时，等距节点插值函数与原函数最大误差为 151.9617,而同样 n=20 时,切比雪夫插值函数与原函数最大误差为 0.3275。仔细观察其他相同的 n 对应最大误差，不难得出，**使用切比雪夫插值点得到的插值函数，能明显降低插值最大误差或者说近似消除了荣格现象。**

并且，对同一个插值函数本身来说，随着 n 的增大，切我比雪夫插值得到的最大误差没有增大，反而减少了。我们也可以由图十五可以看到实际效果，当 n=10,12...,20 时,函数 y=arctan(x)插值函数逼近原函数的效果十分好,特别是 n=20 的时候。

深究其原因，使用切比雪夫节点代替等距节点，有效减小端点附近震荡的原因，我认为是，切比雪夫插值这种方法选取的插值节点，在端点附近是密集的。换句话说，这是因为，**插值函数在端点附近的密集性，能保证其逼近原函数效果之好。**再次引用 wiki，可得在这种情况下，随着多项式阶次的增加最大误差逐渐减小。

从文献中也可知，使用分段多项式样条可以避免这个问题。如果要减小插值误差，那么可以增加构成样条的多项式的数目，而不必是增加多项式的阶次。

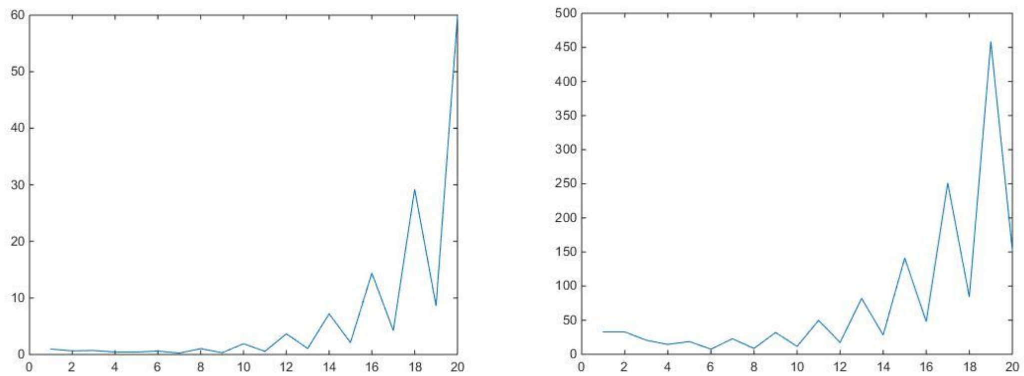
5.补充一点小发现

以上实验结果均基于偶数等分插值区间,而奇数个的情况可由同一个程序可得。经过再次实验，我们可以发现，原函数与插值函数的最大误差还与 n 的奇偶性有关，但是不论 n 为奇数还是偶数，在等距节点的高次插值中，最大误差仍然是有随着 n 的增大而增大。(之前的实验结果是为了函数的清晰而分开讨论 $n<10$ 以及 $n>=20$，后面将合起来讨论)**以函数 $y=1/1+x^2$ 和 y=arctan(x)为例**，如下图十五，$y=1/1+x^2$ 和 y=arctan(x)(n=1,2...,20)插值函数与原函数最大误差变化

图像**均呈锯齿状**。

但是，**对于前者来说(y=1/1+x^2)**，随着 n 的增大，最大误差是保持增大，图十五(左)可以看出,当 n 为偶数的时候明显比 n 为奇数的时候,最大误差较大。**由函数相邻 n 的插值函数图像也可以看出，n 为偶数时插值函数在端点附近震荡现象明显于 n 为奇数,即 Runge 现象明显。对于后者来说(y=arctan(x))，情况则截然相反。即 n 为奇数的时候，Runge 现象较 n 为偶数时明显。**

找其原因，经过对比其他几个函数，可以得出：当被插函数为偶函数时，n 为奇数时，拉格朗日插值函数最高次项的幂次为 n-1，为偶数，与被插函数偶函数性质相同；n 为偶数时，最高幂次为 n-1，为奇数，与被插函数偶函数性质相反。**因此，Runge 现象，还与函数的奇偶性有关。**



图十五 y=1/1+x^2 和 y=arctan(x)每一次逼近最大误差变化图像对比
(n=1,2...,20)

6.解决荣格现象的方法

经过查找相关文献以及网上资料，解决荣格现象的方法可归结为以下几种：

1) 改变插值节点选取方法

比如使用切比雪夫节点代替等距点可以减小震荡，在这种情况下，随着多项式阶次的增加最大误差逐渐减小。

2) 样条逼近

使用样条插值的原因是，分段低次插值虽然计算简单，但是在分段点处常有”尖点”出现，即光滑性比较差。而使用样条插值能解决光滑性这个问题。

3) 放弃个别插值节点

为了达到逼近的终极目的，我们可以放弃个别点的函数值相等，而从整体上逼近。这引出了 Bernstein 的逼近理论。

4) 最小二乘拟合

最小二乘拟合与拉格朗日插值不一样，它允许插值节点函数值与原函数值不一样，只要求总体误差最小。

误差的影响

四、 问题的提出

利用 n 阶泰勒展开多项式 $\sum_{i=0}^{n}\frac{x^i}{i!}$ 计算函数 $f(x)=e^x$ 在给 定点 x 的值。要求绝对误差在最大阶数 MAXN 以内达到给定精度 EPS，即使得 $\left|\sum_{i=0}^{n}\frac{x^i}{i!}-e^x\right|<EPS$，且 $n<MAXN$.

五、 实验内容

1. 自定义函数 Exp_Calculate(x)，使之能实现
   a) 对于给定点 x，若在最大阶数 MAXN=20 的限制内能得到达到精度 EPS=0.00001 要求的函数值，则返回计算所得函数值；若否，则返回-1.（注：可直接调用 C 语言或 MATLAB 的内部函数’exp(x)’ 计算精确值 e^x.）
   b) 将输入的 x 及相应的返回值（保留 4 位小数）显示出来。
2. 使用步骤 1 中定义的函数 Exp_Calculate(x)进行计算：依次输入 x=1, 2, 3, 4, 5, 6, -1, -2, -3, -4, -5, -6，记录下所有输入值和相应的返回值。
3. 对于步骤 2 中返回值为“-1”的情况：
   a) 写出泰勒展开式分析此种情况的特点；
   b) 编写一个测试程序，以检查函数 Exp_Calculate(x)中求和过程 $\sum_{i=0}^{n}\frac{x^i}{i!}$ 的每一步结果，输入各个项（如第 m 项）$\frac{x^m}{m!}$ 的值以及 $\sum_{i=0}^{m}\frac{x^i}{i!}$ 的值，并由此分析造成返回值为-1（计算所得函数值无法达到精度要求）的原因。
4. 根据步骤 3 中分析的原因，找出解决该问题的方法，并编写程序 Exp_Test2(x)（要求同步骤 1 中的 a), b)）进行测试，给出结论。

六、 实验结果及分析

1. 实现自定义函数 Exp_Calculate(x)

详细代码见附录中 Exp_Calculate.m

命令行窗口

```
>> c=Exp_Calculate(4)

x =

     4


c =

   54.5981
```

图十六 显示输入的 x 及相应的返回值 c（保留 4 位小数）

2. 依次输入要求的 x 值并记录下所有输入值和相应的返回值

分别取 x=1，2，3，4，5，6，-1，-2，-3，-4，-5，-6，将得到的实验结果保存如下图十七：

x	返回值	n
1	2.7183	8
2	7.3891	12
3	20.0855	15
4	54.5981	18
5	-1	19
6	-1	19
-1	0.3679	8
-2	0.1353	11
-3	0.0498	14
-4	0.0183	17
-5	-1	19
-6	-1	19

图十七 显示输入要求的 x 结果、相应的返回值 c 以及满足精度的 n
（保留 4 位小数）
(x=1，2，3，4，5，6，-1，-2，-3，-4，-5，-6)

3. 分析上述结果中返回值为“-1”的情况

a) 泰勒展开式进行数值计算产生误差有两个原因，其一是舍弃了余项，产生截断误差；其二，是在于它多项式中每一项都进行了近似计算，有由于除法而产生的舍入误差。考虑题目对应麦克劳林公式(拉格朗日型余项)

$$e^x = 1 + x + \frac{x^2}{2!} + \ldots + \frac{x^n}{n!} + \frac{e^{\theta x}}{(n+1)!} x^{n+1} (0 < \theta < 1)$$

下面是将上述返回值为-1 的特例进行泰勒展开的数值表示：

x=5 时,
e^5=1+5+(5^2)/2!+(5^3)/3!+(5^4)/4!+(5^5)/5!+(5^6)/6!+(5^7)/7!+(5^8)/8!+(5^9)/9!+(5^10)/10!+(5^11)/11!+(5^12)/12!+(5^13)/13!+(5^14)/14!+(5^15)/15!+(5^16)/16!+(5^17)/17! +(5^18)/18!+(5^19)/19!
x=6 时,
e^6=1+6+(6^2)/2!+(6^3)/3!+(6^4)/4!+(6^5)/5!+(6^6)/6!+(6^7)/7!+(6^8)/8!+(6^9)/9!+(6^10)/10!+(6^11)/11!+(6^12)/12!+(6^13)/13!+(6^14)/14!+(6^15)/15!+(6^16)/16!+(6^17)/17!+(6^18)/18!+(6^19)/19!
x=-5 时,
e^(-5)=1+(-5)+((-5)^2)/2!+((-5)^3)/3!+((-5)^4)/4!+((-5)^5)/5!+((-5)^6)/6!+((-5)^7)/7!+((-5)^8)/8!+((-5)^9)/9!+((-5)^10)/10!+((-5)^11)/11!+((-5)^12)/12!+((-5)^13)/13!+((-5)^14)/14!+((-5)^15)/15!+((-5)^16)/16!+((-5)^17)/17!+((-5)^18)/18!+((-5)^19)/19!
x=-6 时,
e^(-6)=1+(-6)+((-6)^2)/2!+((-6)^3)/3!+((-6)^4)/4!+((-6)^5)/5!+((-6)^6)/6!+((-6)^7)/7!+((-6)^8)/8!+((-6)^9)/9!+((-6)^10)/10!+((-6)^11)/11!+((-6)^12)/12!+((-6)^13)/13!+((-6)^14)/14!+((-6)^15)/15!+((-6)^16)/16!+((-6)^17)/17!+((-6)^18)/18!+((-6)^19)/19!

以 x=5 为例，其不能满足精度(返回值为-1)的原因为在 MAXN 中余项过大。令 x=5,先考虑截断误差 $R_{截}=\dfrac{e^{\theta}}{(n+1)!}5^{n+1}$，余项中 $0<e^{\theta}<e$ 所以要使余项 $\dfrac{e^{\theta}}{(n+1)!}5^{n+1}<10^{-5}$,经过计算 n 至少为 21。以上为未考虑舍入误差时，就不可能满足精度。若考虑保留 4 位小数舍入误差，由于前三项无舍入误差，所以对于 x=5，n=MAXN-1 时，还需要加上 $R_{舍}=17\times0.5\times10^{-4}$ ，总误差 $R_{总}=R_{舍}+R_{截}$。

对于其他返回值为-1 的情况，同样也是通过舍入误差和截断误差来分析。其中，$R_{舍}=(\text{n-3})\times0.5\times10^{-4}$，$R_{截}=\dfrac{e^{\theta}}{(n+1)!}x^{n+1}$。根据特定的 x，由 $R_{总}<10^{-5}$，可算出对应 n，若 n>MAXN，则返回值为-1;否则，返回值为所求得麦克劳林公式的值。

b)编写一个测试程序

以 x=3,4,5,6,-5,-6 为例，测试程序的结果如下图十八：

x=3(n=15)	x=4(n=18)	x=5	x=6	x=-5	x=-6
1	1	1	1	1	1
3	4	5	6	-5	-6
4.5	8	12.5	18	12.5	18
4.5	10.6667	20.8333	36	-20.8333	-36
3.375	10.6667	26.0417	54	26.0417	54
2.025	8.5333	26.0417	64.8	-26.0417	-64.8
1.0125	5.6889	21.7014	64.8	21.7014	64.8
0.4339	3.2508	15.501	55.5429	-15.501	-55.5429
0.1627	1.6254	9.6881	41.6571	9.6881	41.6571
0.0542	0.7224	5.3823	27.7714	-5.3823	-27.7714
0.0163	0.289	2.6911	16.6629	2.6911	16.6629
0.0044	0.1051	1.2232	9.0888	-1.2232	-9.0888
0.0011	0.035	0.5097	4.5444	0.5097	4.5444
0.0003	0.0108	0.196	2.0974	-0.196	-2.0974
0.0001	0.0031	0.07	0.8989	0.07	0.8989
0	0.0008	0.0233	0.3596	-0.0233	-0.3596
	0.0002	0.0073	0.1348	0.0073	0.1348
	0	0.0021	0.0476	-0.0021	-0.0476
	0	0.0006	0.0159	0.0006	0.0159
		0.0002	0.005	-0.0002	-0.005

图十八 测试实验函数的麦克劳林公式每一项结果(x=3, 4, 5, 6, -5, -6)

首先,测试结果中，x=3, x=4 均是有返回值的，并且分别在 n=15, n=16 就已经满足精度要求，也就是麦克劳林展开 16 项，17 项就已经满足精度要求。而 x=5, 6, -5, -6 为返回值为-1 的情况。为了方便对比说明，图十七将以上测试结果放在一起。

由测试程序的结果，我们可以发现返回值为-1 对应的 x，麦克劳林公式第 20 项均不为零，也就是说在 n=19 时，仍不能达到精度要求，计算没有停止。在 n=19 时对应的泰勒公式拉格朗日型余项为 $R_{截}=\frac{e^{\theta x}}{20!}x^{20}$。

由 $\left|\sum_{i=0}^{n}\frac{x^i}{i!}-e^x\right|=\left|\frac{e^{\theta x}}{(n+1)!}x^{n+1}\right|(0<\theta<1)$，计算可得，

$x=5$ 时，$\left|\sum_{i=0}^{n}\frac{x^i}{i!}-e^x\right|=\left|\frac{e^{5\theta}}{20!}x^{20}\right|<\left|\frac{e^{5}}{20!}5^{20}\right|$，$R_{截}=5.81765\times10^{-3}>10^{-5}$；

同理，$x=6$ 时，$R_{截}=0.60627>10^{-5}$；

$x=-5$ 时，$R_{截}=3.91990\times10^{-5}>10^{-5}$；

$x=-6$ 时，$R_{截}=1.50280\times10^{-3}>10^{-5}$。

因此就单从截断误差的角度来看就知道不满足精度要求。

由于好奇心所趋，后面继续讨论返回值为-1 情况，在展开到第几项的时候满足精度要求，再次测试结果如下图十八：

x=5(n=21)	x=6(n=24)	x=-5(n=20	x=-6(23)
1	1	1	1
5	6	-5	-6
12.5	18	12.5	18
20.8333	36	-20.8333	-36
26.0417	54	26.0417	54
26.0417	64.8	-26.0417	-64.8
21.7014	64.8	21.7014	64.8
15.501	55.5429	-15.501	-55.5429
9.6881	41.6571	9.6881	41.6571
5.3823	27.7714	-5.3823	-27.7714
2.6911	16.6629	2.6911	16.6629
1.2232	9.0888	-1.2232	-9.0888
0.5097	4.5444	0.5097	4.5444
0.196	2.0974	-0.196	-2.0974
0.07	0.8989	0.07	0.8989
0.0233	0.3596	-0.0233	-0.3596
0.0073	0.1348	0.0073	0.1348
0.0021	0.0476	-0.0021	-0.0476
0.0006	0.0159	0.0006	0.0159
0.0002	0.005	-0.0002	-0.005
0	0.0015	0	0.0015
0	0.0004		-0.0004
	0.0001		0.0001
	0		0
	0		

图十九 测试实验函数的麦克劳林公式每一项结果(x=5,6,-5,-6)以及满足精度展开阶数 n

其实，在保留四位小数的前提下，以 x=5 为例，该公式在展开的第 21 项开始，每一项都被”吃掉”剩余的小数部分。在展开项数增加的时候，舍入误差不断扩大(每一项最大舍入误差为 0.5*10^-4)，而数值结果是会更加精确地，因为截断误差在不断减小，而且减小的速度比舍入误差快。最终，随着展开项数增加，数值精确值是可以不断增加的。本问题体现了误差中截断误差和舍入误差的动态平衡，考虑误差分配原则中的截断误差=舍入误差。

4. 根据步骤 3 中分析的原因，找出解决该问题的方法，并编写程序 Exp_Test2(x)（要求同步骤 1 中的 a)，b)）进行测试，给出结论。

经过修改后的程序 Exp_Test2(x) 见附录。修改的部分主要体现在截断误差的改进上。通过把函数改成 $e^x=(e^{\frac{x}{2}})^2$，从而先通过 $e^{\frac{x}{2}}$ 的泰勒展开 $\mathrm{e}^{\frac{x}{2}}=\sum_{i=0}^{n}\frac{x^i}{2^{\mathrm{i}}\cdot i!}$，再通过其展开结果平方得到结果。$e^{\frac{x}{2}}$ 的截断误差为 $\frac{(\theta x)^{n+1}}{(n+1)!\cdot 2^{n+1}}e^{\frac{\theta x}{2}}$，得到有效地减

小。(n=19 时，$R_{截}=\dfrac{(\theta x)^{20}}{20!\cdot 2^{20}}e^{\frac{\theta x}{2}}(0<\theta<1)$)。计算结果如下：

```
>> c=Exp_Calculate(5)

c =

   148.4132
```

图二十 显示输入的 x(x=5)及相应的返回值 c（保留 4 位小数）

5. 依次输入要求的 x 值并记录下所有输入值和相应的返回值

x=5(n=15)	x=6(n=17)	x=-5(n=12	x=-6(23)
1	1	1	1
2.5	3	-2.5	-3
3.125	4.5	3.125	4.5
2.6042	4.5	-2.6042	-4.5
1.6276	3.375	1.6276	3.375
0.8138	2.025	-0.8138	-2.025
0.3391	1.0125	0.3391	1.0125
0.1211	0.4339	-0.1211	-0.4339
0.0378	0.1627	0.0378	0.1627
0.0105	0.0542	-0.0105	-0.0542
0.0026	0.0163	0.0026	0.0163
0.0006	0.0044	-0.0006	-0.0044
0.0001	0.0011	0.0001	0.0011
0	0.0003		-0.0003
0	0.0001		
0	0		
	0		
	0		

图二十一 显示输入要求的 x 结果、相应的返回值 c 以及满足精度的 n
（保留 4 位小数）(x=5, 6,-5, -6)

由此结果，我们可以发现，相同的 x 的返回值不为-1 了，也就是说之前不能满足精度要求的 x，现在能够满足精度要求了。**这是因为，截断误差的有效缩小，在相同的展开阶数中 $e^{\frac{x}{2}}$ 比 e^{x} 泰勒展开逼近的效果更好，截断误差更小。**

$$\left|\sum_{i=0}^{n}\frac{x^i}{2^i\cdot i!}-e^x\right|=\left|\frac{(\theta x)^{n+1}}{(n+1)!\cdot 2^{n+1}}e^{\frac{\theta x}{2}}\right|(0<\theta<1)$$

以 x=5 为例，n=19 时 $R_{截}$

$$\left|\frac{(5\theta)^{20}}{(20)!\cdot 2^{20}}e^{\frac{5\theta}{2}}\right|<\left|\frac{5^{20}}{(20)!\cdot 2^{20}}e^{\frac{5}{2}}\right|=4.55420\times 10^{-10}<10^{-5}(0<\theta<1)$$

同理，分别计算 x=6 时，

$$R_{截}=2.87862\times 10^{-8}<10^{-5}$$

x=-5 时，

$$R_{截}=3.73831\times 10^{-11}<10^{-5}$$

x=-6 时，

$$R_{截}=1.43318\times 10^{-9}<10^{-5}$$

5. 其他小发现

其实除了从截断误差方面考虑，也可以从舍入误差去考虑(eg.把精确到小数点后 4 位改为更高的位数等)由文献[3]可以得到几种减小舍入误差的方法：

(1)增加除法的计算精度，也就是增加四舍五入到小数点后面位数。

(2)每项分数先不去进行数值运算，而是对分数的和运算进行符号的运算，对最后的结果才进行数值的运算。但是复杂性会随计算项数增加而增加。

(3)寻找不可控有限分数的对偶分数(也就是相加变成可控分数，小数部分在精度范围内)以及无限分数的对偶分数，然后调整运算的次序，挑出一些分数求和，使其和无误差或误差减少。

七、 关于本设计的体会

对于问题一，在荣格现象的研究中，我更加深入理解了拉格朗日插值的优劣以及插值误差余项 Rn 的影响因素分析。对于最大误差，学会了用可视化的方法，用 Matlab 画出图像来表示，在另一方面也加强了自己的编程技能。凡事都是能有多种解决方案的，条条大路通罗马，当插值多项式高次时误差反而增大时，可改变策略采用低次插值或者样条插值等方法，也可以保留原策略去用非等距节点(Chebychev 节点)进行插值同样可以解决 Runge 现象。不得不感慨数学世界的奇妙。

对于附加题，即问题二，在误差的影响研究中，深刻感受到了数值计算中考虑总误差(舍入误差、截断误差)的平衡原则。如同课本上所说的用计算机解决科学计算问题的过程：提出实际问题、建立数学模型、选用数学计算方法、程序设计、上机计算求出数值结果。现实数值计算中，观测误差、模型误差我们无法改变，但是我们可以从截断误差和舍入误差中进行考虑，从而把数值计算的结果更加精确化。误差分析的思想也能很好的运用到数学建模的比赛中，对自己所建立模型求解后，加上误差分析这一步，考虑误差传播的影响，往往能把自己所建的模型更加完善，更具有科学性。

总的来说，这次数值计算的课程设计难度并不高，但是通过自己上网查阅文献，复习数值分析课本、数学分析课本，也发现了不少自己知识的漏洞，为以后层层递进的学习迈上了很重要的一步。

八、 参考文献

[1]石瑞文. 数值计算[M].高等教育出版社,2014.
[2]李庆扬. 数值分析 第 5 版[M].清华大学出版社,2008.
[3]邵任翔,万丽,邓小成. 一种减少泰勒公式计算误差的数值方法[J]. 广州大学学报(自然科学版),2015,05:9-12.

九、 附录

Runge1.m

```
%数值分析课程设计
%Runge 现象的分析
clear;
x=-5:0.01:5;%画图节点步长选择为 0.01
y1=1./(1+x.^(2));%对应原函数值
plot(x,y1,'LineWidth',5);%原函数
hold on
%gtext('原函数');
title('1/(1+x^2)的 Runge 现象(n=2,4...,10)','fontsize',24);
xlabel('x 轴','fontsize',24);
ylabel('y 轴','fontsize',24);
 z=1;%计算次数
for n=2:2:10

x0=-5:10/n:5;%插值节点
y0=1./(1+x0.^(2));%对应函数值
k=length(x0)%计算构造的插值多项式次数

Ln=0;%初始化插值多项式

for i=1:k
    L=1;
    for j=1:k
        if i~=j
            L=L.*((x-x0(j))./(x0(i)-x0(j)));%分别计算 lk(x)
        end
    end
    Ln=Ln+L.*y0(i);%拉格朗日插值多项式
end
result=Ln;
length(y1);

X=Ln-y1;%插值多项式值与原函数值差向量
R(z,1)=n;%插值区间等分份数
R(z,2)=norm(X,inf);%计算每个拉格朗日多项式函数与原函数代入画图节点后
```

```
得到向量差对应无穷范数
plot(x,Ln,'LineWidth',5);
hold on
%gtext(['n = ' num2str(n)]);
z=z+1;
end
legend('原函数','n=2','n=4','n=6','n=8','n=10');
R
grid on
```

Chebychev.m

```
%数值分析课程设计
%Runge 现象的分析
%切比雪夫插值点
clear;
a=-5;
b=5;
x=-5:0.01:5;%画图节点步长选择为 0.01
y1=1./(1+x.^(2));%对应原函数值
plot(x,y1,'LineWidth',5);%原函数
hold on
%gtext('原函数');
title('y=1/(1+x^2)使用切比雪夫插值点后 Runge 现象
(n=2,4...,10)','fontsize',24);
xlabel('x 轴','fontsize',24);
ylabel('y 轴','fontsize',24);
 z=1;%计算次数
for n=2:2:10
   for k=1:n+1%插值节点
       x0(k)=(a+b)/2+((b-a)/2)*cos(((2*k-1)*pi)/(2*(n+1)))
   end
y0=1./(1+x0.^(2));%对应函数值
k=length(x0)%计算构造的插值多项式次数

Ln=0;%初始化插值多项式

for i=1:k
    L=1;
    for j=1:k
        if i~=j
            L=L.*((x-x0(j))./(x0(i)-x0(j)));%分别计算 lk(x)
        end
    end
    Ln=Ln+L.*y0(i);%拉格朗日插值多项式
```

```
end
result=Ln;
length(y1);

X=Ln-y1;%插值多项式值与原函数值差向量
R(z,1)=n;%插值区间等分份数
R(z,2)=norm(X,inf);%计算每个拉格朗日多项式函数与原函数代入画图节点后得到向量差对应无穷范数
plot(x,Ln,'LineWidth',5);
hold on
%gtext(['n = ' num2str(n)]);
z=z+1;
end
legend('原函数','n=2','n=4','n=6','n=8','n=10');
R
grid on
```

Exp_Calculate.m

```
%数值分析课程设计    By 何唯
%误差的影响
function c=Exp_Calculate(x)
eps=0.00001;
MAXN=20;
c=-1;
for n=1:MAXN-1   %n 从 1 取到 19
    sum=0;
    temp=1;
    for i=0:n   %每个 n 对应的泰勒展开式算出来的值
        if i==0
           sum=1;
        else
            temp=temp*i;
            sum=sum+(x^(i)/temp);
        end
       if(abs(sum-exp(x))<eps)%满足精度则退出
           c=sum;
           c=roundn(c,-4);
           return ;
       end
    end
end
c=roundn(c,-4);
End
```

Exp_Calculate2.m

```
%数值分析课程设计    By 何唯
%误差的影响
%测试麦克劳林公式每一项
function c=Exp_Calculate(x)
eps=0.00001;
MAXN=20;
c=-1;
for n=1:MAXN-1    %n 从 1 取到 20
    sum=0;
    temp=1;
    result=zeros(n+1,1);
    for i=0:n    %每个 n 对应的泰勒展开式算出来的值
        if i==0
            sum=1;
            result(1,1)=1;
        else
             temp=temp*i;
             result(i+1,1)=x^(i)/temp;  %利用 result 数组保存麦克劳林公
式中每一项
             sum=sum+(x^(i)/temp);
        end
       if(abs(sum-exp(x))<eps)%满足精度则退出
            c=sum;
            c=roundn(c,-4);
            n
            result
            return ;
       end
    end
end
c=roundn(c,-4);
n
result
end
```

Exp_Test2(x).m

```
%数值分析课程设计    By 何唯
%误差的影响
%改进截断误差后
function [c,result]=Exp_Test2(x)eps=0.00001;
MAXN=20;
c=-1;
format
```

```
for n=1:MAXN-1    %n 从 1 取到 20
    sum=0;
    temp=1;
    result=zeros(n+1,1);
    for i=0:n    %每个 n 对应的泰勒展开式算出来的值
        if i==0
           sum=1;
           result(1,1)=1;
        else
            temp=temp*i;
            result(i+1,1)=(x/2)^(i)/temp;   %利用 result 数组保存麦克劳
林公式中每一项
            sum=sum+((x/2)^(i)/temp);
        end
       if(abs((sum^2)-exp(x))<eps)%满足精度则退出
           c=sum^2;

           n
           result
           c
           return ;
       end
    end
end
n
sum
result
end
```