第9章 微分方程求解

$$\frac{dy}{dx} = 3y$$

$$\frac{dy}{y} = 3dx$$

9.1 引言

科学和工程中建立数学模型时经常用到微分方程。如Logistic人口增长模型、种群动力学模型、SIR流行病模型以及SEIR流行病模型。由于它们通常没有已知的解析解，因而需要求其数值近似解。

《改进SIR 模型在社交网络信息传播中的应用》

信息在社交网络中的传播方式与传染病在人群中的传播方式存在一定的相似性。将经典的传染病SIR模型进行改进，构建基于改进SIR的社交网络信息传播模型，并以某社交网络实际数据为样本，对模型进行仿真研究，对研究信息在社交网络中的传播机制及传播规律具有重要的实际意义。

9.1 引言

考察如下种群动力学模型，称为Lotka-Volterra (洛特卡-沃尔泰拉)方程

$$\begin{cases} x' = f(t,x,y) = x - xy - x^2/10, \\ y' = g(t,x,y) = xy - y - y^2/20. \end{cases}$$

初始条件是 $x(0)=2$ 和 $y(0)=1, 0\leqslant t\leqslant 30$。尽管数值解只是一个数值列表，但画出连接近似点 $\{(x_k, y_k)\}$ 得到的多边形路径和轨线还是有助于了解问题，如图 9.1 所示。本章给出求解常微分方程、微分方程组和边界值问题的标准方法。

x表示被捕食者
y表示捕食者

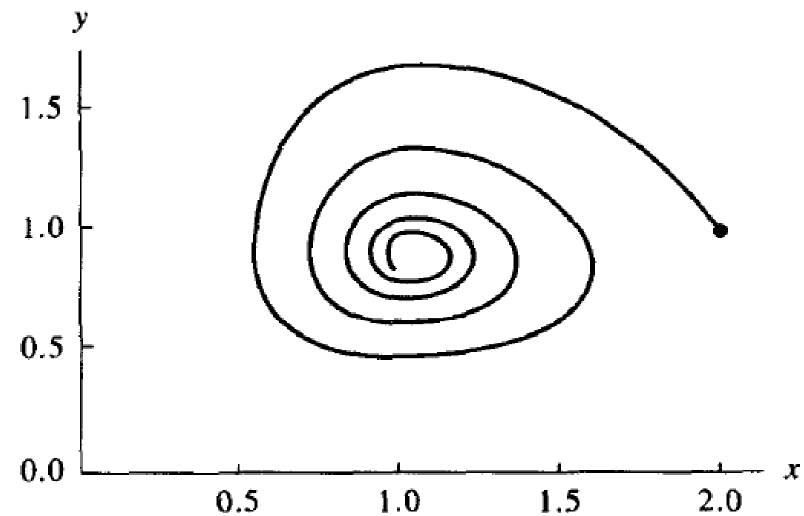


图 9.1 非线性系统微分方程 $x' = f(t,x,y)$ 和 $y' = g(t,x,y)$ 的轨线

考察方程

$$\frac{dy}{dt} = 1 - e^{-t} \tag{1}$$

是一个微分方程，因为它包含“未知函数”$y = y(t)$的导数 dy/dt。由于只有独立变量 t 出现在式(1)的右端项中，因此 $1 - e^{-t}$的不定积分是方程的一个解。可由积分公式求解 $y(t)$：

$$y(t) = t + e^{-t} + C \tag{2}$$

其中 C 为积分常数。式(2)中的所有函数都是方程(1)的解，因为它们都能满足 $y'(t) = 1 - e^{-t}$，构成了图 9.2 中的曲线族。

积分方法可用于求解式(2)中函数的显式公式。图 9.2 显示，在这样的解中有 1 个自由度，即积分常量 C。通过改变 C 的值可以向上或向下“移动解曲线”，从而可以找到过任意需要点的曲线。然而世界的奥秘极少表现为显式的公式，通常只能考查一个变量的变化如何影响另一个变量。将这种方法翻译为数学模型，得到的就是包含未知函数的变化率及自变量和/或应变量的方程。

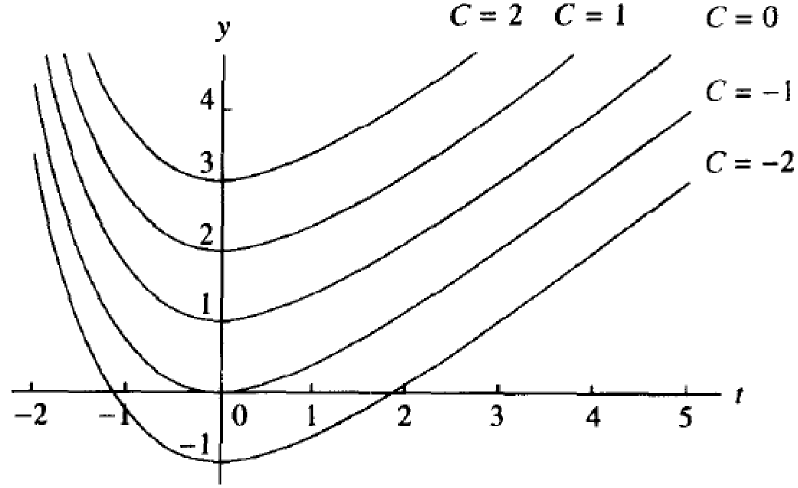


图 9.2　解曲线 $y(t) = t + e^{-t} + C$

9.2 Eular方法

一、初值问题及其数值解的概念

一阶常微分方程初值问题：
$$\begin{cases} \dfrac{dy}{dx} = f(x,y) \\ y(x_0) = y_0,\ x \geq x_0 \end{cases} \qquad (*)$$

注意解曲线必须经过初始点(x_0, y_0).

常用的一些解析解法：

分离变量法、变量代换、
常数变易法、Laplace变换等

定理9.1（解的存在唯一性）

对于初值问题(*)，如果 $f(x,y)$ 在下列区域内连续：

$$G=\{a\le x\le b;|y|<\infty\}$$

且关于y满足Lipschitz条件，即存在常数$L>0$，使

$$|f(x,y_1)-f(x,y_2)|\le L|y_1-y_2|;\forall x,y\in G$$

则初值问题(*)存在唯一解，且解是连续可微的。

这里的Lipschitz条件可以加强为：

设 $f(t,y)$ 定义在区域 R 上，如果存在一个常数 $L>0$，使得

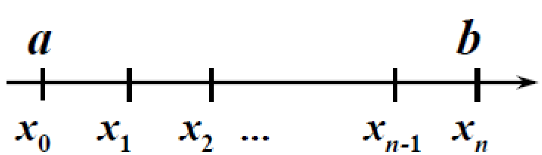
$$|f_y(t,y)|\le L,\qquad (t,y)\in R$$

应相信，不是所有的初值问题都有显示解，而且通常不可能找到解y(x)的公式。因此，在科学与工程中，需有计算近似解的方法。如果要求解具有多位有效数字，则需要更多的计算量和复杂的算法。

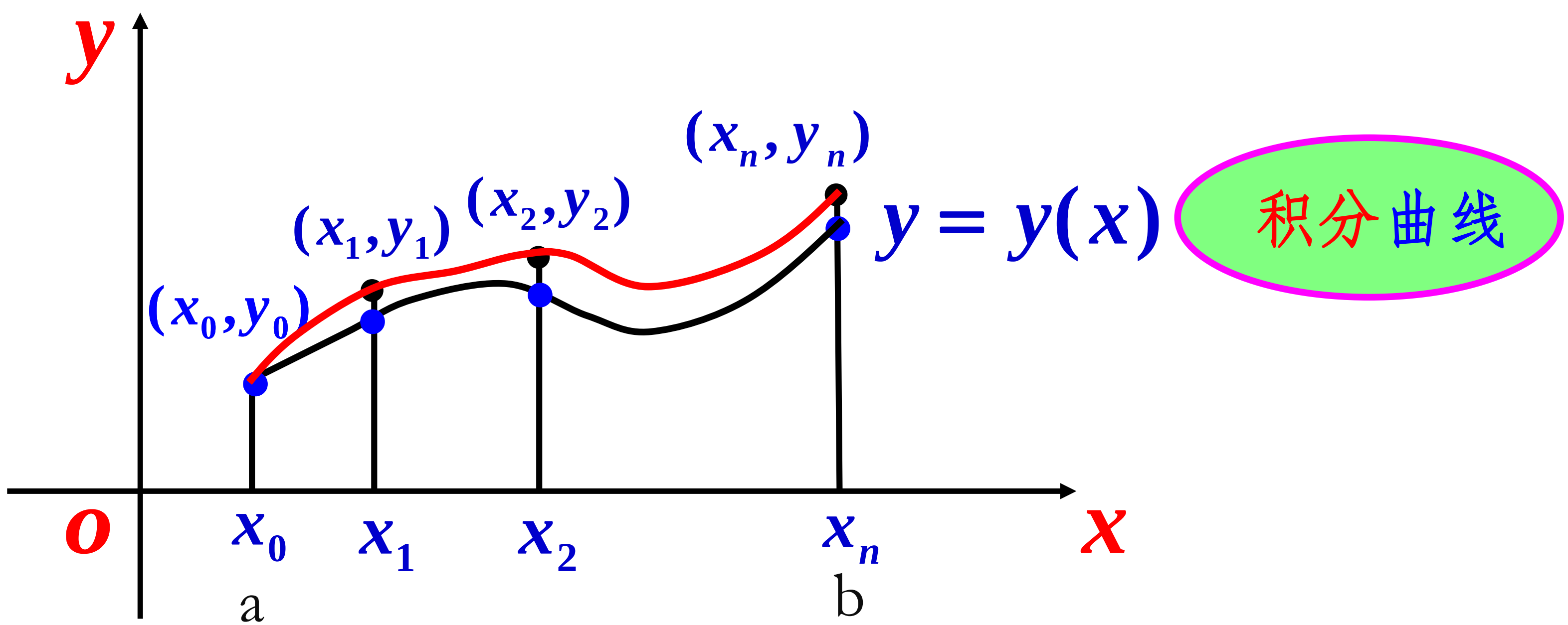
下面我们给出数值解的定义：

设微分方程问题的解$y(x)$的存在区间是$[a,b]$，初始点$x_0=a$，将$[a,b]$进行划分得一系列节点 $x_0, x_1, \ldots,$ x_n，其中$a=x_0<x_1<\ldots<x_n=b$。

$y(x)$的解析表达式不容易得到或根本无法得到，我们用数值方法求得$y(x)$在每个节点x_k的近似值$y(x_k)$，即$y\approx y(x_k)$，这样$y_0, y_1, \ldots, y_n$称为微分方程的数值解。如图所示：



➢初值问题(*)的解析解及其数值解的几何意义：



初值问题(*)的解表示过点(x_0,y_0)的一条曲线

初值问题(*)的数值解表示一组离散点列(x_i,y_i)

可用插值或拟合方法求该组数据(x_i,y_i)的近似曲线

9.2 Eular方法

考察初值问题:
$$\begin{cases} \dfrac{dy}{dx} = f(x,y) \\ y(x_0) = y_0; x \geq x_0 \end{cases}$$

将区间[a, b]划分为M个等距子区间，并选择网格点

$$x_k = a + kh,\ k = 0, 1, 2, \cdots, M \text{ with } h = \frac{b-a}{M}$$

值h称为步长。

三类Eular方法的推导公式:

(1) 向前差商近似微分法

向前差商 $\dfrac{y(x_{k+1}) - y(x_k)}{h}$ 近似 $y'(x_k)$, 得 $\dfrac{y(x_{k+1}) - y(x_k)}{h} \approx f(x_k, y(x_k))$

将近似号改为等号，用 $y_k, y_{k+1}, f_k = f(x_k, y_k)$ 近似 $y(x_k), y(x_{k+1})$, $f(x_k, y(x_k))$, 并结合初始条件即得

欧拉格式
$$\begin{cases} y_0 = y(x_0) \\ y_{k+1} = y_k + hf(x_k, y_k), \quad k = 0, 1, \cdots, n-1 \end{cases} \tag{9.1}$$

(2) Taylor展开法

由于 $y'(x) = \dfrac{dy}{dx} = f(x, y)$

因此 $y'(x_k) = f(x_k, y(x_k))$

将$y(x_{k+1})$在 $x = x_k$ 点进行Taylor展开

$$y(x_{k+1}) = y(x_k) + h\underbrace{f(x_k, y(x_k))}_{y'(x_k)} + \boxed{\frac{y''(\xi_k)}{2!}h^2}, \xi_k \in [x_k, x_{k+1}]$$

忽略h_k^2这一高阶项，分别用 y_k, y_{k+1}，近似$y(x_k), y(x_{k+1})$，得

$$y_{k+1} = y_k + hf(x_k, y_k)$$

结合初值条件$y(x_0) = y_0$即得（9.1）.

(3) 数值积分法

将初值问题两端从 x_k 到 x_{k+1} 积分,得

$$y(x_{k+1})-y(x_k)=\int_{x_k}^{x_{k+1}} f(x,y(x))dx$$

假设数值积分采用左矩形公式，即

$$\int_{x_k}^{x_{k+1}} f(x,y(x))dx \approx hf(x_k,y(x_k))$$

用 y_k, y_{k+1} 近似 $y(x_k)$, $y(x_{k+1})$ 得

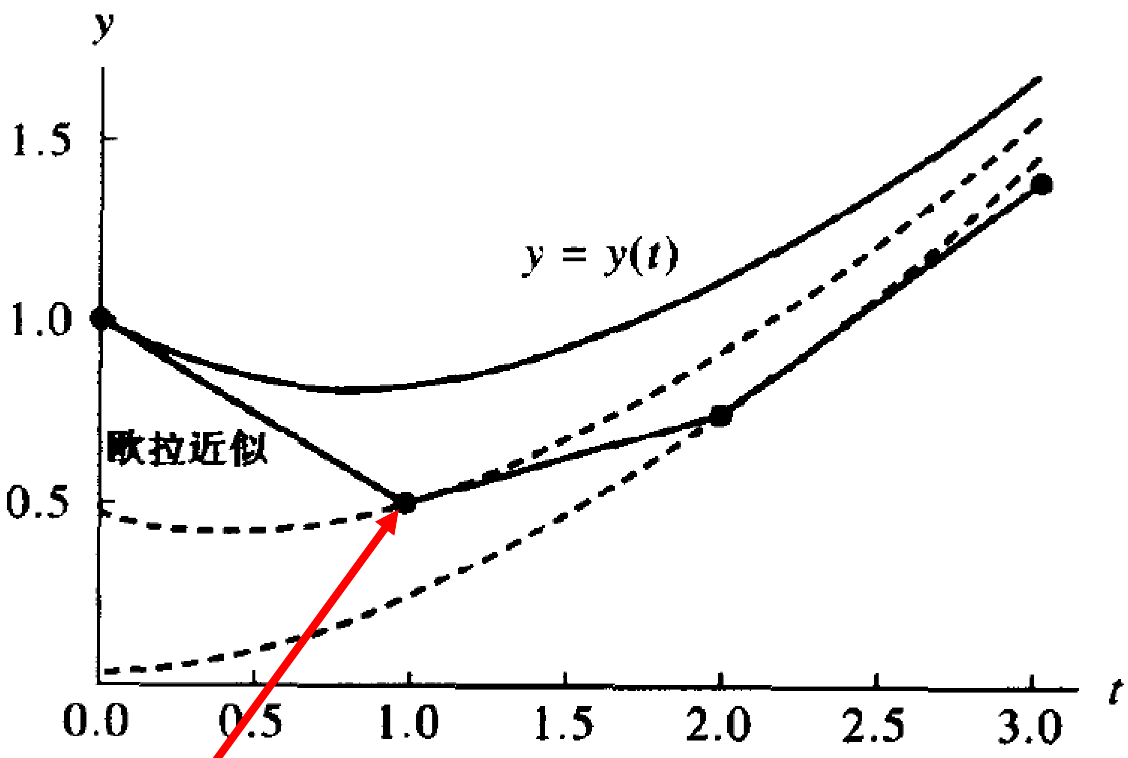
$$y_{k+1}-y_k=hf(x_k,y_k)$$

由初始条件亦得 (9.1).

$$\begin{cases} y_0=y(x_0), \\ y_{k+1}=y_k+hf(x_k,y_k), \end{cases} \quad k=0,1,\cdots,n-1$$

9.2 Eular方法

$$\begin{cases} \dfrac{dy}{dx} = f(x, y) \\ y(x_0) = y_0; x \geq x_0 \end{cases}$$



Euler方法
又称折线法

注意（t1, y1）并不在所要的解曲线上，但是这里计算的是近似值，因此必须用（t1, y1），并通过计算t1点的斜率，来得到下一个竖直位移，找到（t2, y2）.

Euler方法的几何意义：用一条折线近似代替积分曲线

Euler方法的缺陷：误差比较大

Euler格式的步长与误差

定义9.1： 设$y=y(x)$是初值问题的唯一解，$\{x_k,y_k\}_{k=0}^{M}$是离散近似解集。迭代格式：

$$\boxed{y_{k+1}=y_k+hf(x_k,y_k),}\quad k=0,1,\cdots,n-1$$

单步的，显示的

全局离散误差e_k定义为：

$$e_k=y(x_k)-y_k,\quad k=0,1,2,\ldots,M$$

它是唯一解与离散方法得到的解之间的差.

局部离散误差R_k定义为

$$R_{k+1}=y(x_{k+1})-hf(x_k,y_k)-y_k,\quad k=0,1,\ldots,M-1$$

它是从x_k到x_{k+1}这一步计算的误差.

Euler格式的步长与误差

定义9.2: 若某算法的局部截断误差为$O(h^{p+1})$，则称该算法有***p* 阶精度** (整体截断误差)。

☞ 欧拉法的局部截断误差：

$$R_{i+1} = y(x_{i+1}) - y_{i+1} = [\cancel{y(x_i)} + \cancel{hy'(x_i)} + \frac{h^2}{2} y''(x_i) + O(h^3)] - [\cancel{y_i} + \cancel{hf(x_i, y_i)}]$$

$$= \frac{h^2}{2} y''(x_i) + O(h^3) = O(h^2)$$

欧拉法具有 **1** 阶精度。

回顾

判断非线性方程迭代求解收敛速度的概念

设迭代 $x_{k+1}=g(x_k)$ 收敛到 $g(x)$ 的不动点 x^*。记绝对误差 $e_k = x_k - x^*$，若

$$\lim_{k\to\infty}\left|\frac{e_{k+1}}{e_k{}^p}\right| = C \neq 0 \quad (C\text{为常数})$$

则称该迭代为以收敛阶p收敛到x^*。数C称为渐近误差常数。

判断数值积分求积公式收敛速度的概念

求积公式有m次代数精度的充要条件为相应公式式对$f(x)=1, x, x^2, \ldots, x^m$精确成立，但对 $f(x)=x^{m+1}$ 不精确成立.

判断微分方程数值解收敛速度的概念

若某算法的局部断误差为$O(h^{p+1})$，则称该算法有p 阶精度 (整体截断误差)。

例9.1 考察方程 $\begin{cases} \dfrac{dy}{dx} = \dfrac{2y}{x} + 2 \\ y(1) = -1 \end{cases}$

解： $h=0.2$, $x_i=1+ih$

$$y_1 = y_0 + hf(x_0, y_0) = -1+0.2\times\left(\frac{2\times(-1)}{1.0}+2\right) = -1$$

$$y2 = y1 + hf(x1, y1) = -1+0.2\times\left(\frac{2\times(-1)}{1.2}+2\right) = -0.9333$$

$$y3 = y2 + hf(x2, y2) = -0.9333+0.2\times\left(\frac{2\times(--0.9333)}{1.4}+2\right) = -0.8$$

$$y4 = y3 + hf(x3, y3) = 0.8+0.2\times\left(\frac{2\times(-0.8)}{1.6}+2\right) = -0.6$$

$$y5 = y4 + hf(x4, y4) = 0.6+0.2\times\left(\frac{2\times(-0.6)}{1.8}+2\right) = -0.3333$$

$$y6 = y5 + hf(x5, y5) = 0.3333+0.2\times\left(\frac{2\times(-0.3333)}{2}+2\right) = 0$$

精确解为： $y = x^2 - 2x$

全部离散误差

x_k	$y(x_k)$	y_k	e_k
1.2	-0.96	-1	0.04
1.4	-0.84	-0.9333	0.0933
1.6	-0.64	-0.8	0.16
1.8	-0.36	-0.6	0.24
2.0	0	-0.3333	0.3333
2.2	0.44	0	0.44

可以看出误差随着计算在积累。

Matlab实现

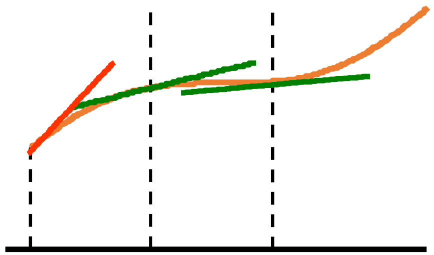
```
function E=Euler(f,a,b,ya,M)

%Input    - y'=f is the function
%         - a and b are the left and right endpoints
%         - ya is the initial condition y(a)
%         - M is the number of steps
%Output - E=[T' Y'] where T is the vector of
 % abscissas and Y is the vector of ordinates

h=(b-a)/M;
T=zeros(1,M+1);
Y=zeros(1,M+1);
T=a:h:b;
Y(1)=ya;

for j=1:M
   Y(j+1)=Y(j)+h*f(T(j),Y(j));
end

E=[T' Y'];
```

命令行窗口

```
>> f=@(t,y)  (t*exp(t)-1)*sin(y); a=1; b=3; ya=0.1; M=20;
 E=Euler(f,a,b,ya,M)

E =

    1.0000    0.1000
    1.1000    0.1172
    1.2000    0.1441
    1.3000    0.1869
    1.4000    0.2570
    1.5000    0.3759
    1.6000    0.5860
    1.7000    0.9689
    1.8000    1.6536
    1.9000    2.6391
    2.0000    3.2027
    2.1000    3.1185
    2.2000    3.1558
    2.3000    3.1290
    2.4000    3.1566
    2.5000    3.1184
    2.6000    3.1866
    2.7000    3.0335
    2.8000    3.4561
    2.9000    2.0628
    3.0000    6.6200

fx >>
```

9.3 显式欧拉格式的改进

1). 隐式欧拉格式



向后差商近似导数 ⟶ $y'(x_1) \approx \dfrac{y(x_1) - y(x_0)}{h}$

⟶ $y(x_1) \approx y_0 + h\, f(x_1, y(x_1))$

$$y_{i+1} = y_i + h\, f(x_{i+1}, y_{i+1}) \quad (i = 0, \ldots, n-1)$$

由于未知数 y_{i+1} 同时出现在等式的两边，不能直接得到，故称为隐式欧拉格式。

一般先用显式计算一个初值, 再迭代求解. 对于简单格式, 把隐式化为显式.

隐式欧拉法的局部截断误差

$$R_{i+1} = y(x_{i+1}) - y_{i+1} = O(h^2)$$

即隐式欧拉格式具有 1 阶精度。

2). 梯形格式 — 显式、隐式两种算法的平均 又称为休恩方法

在区间$[x_n, x_{n+1}]$上对初值问题两端积分:

$$y(x_{n+1}) - y(x_n) = \int_{x_n}^{x_{n+1}} f(x, y)dx$$

$$\approx \frac{h}{2}[f(x_n, y(x_n)) + f(x_{n+1}, y(x_{n+1}))]$$

然后用y_n代替$y(x_n)$，得

$$y_{n+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}, y_{n+1})]$$

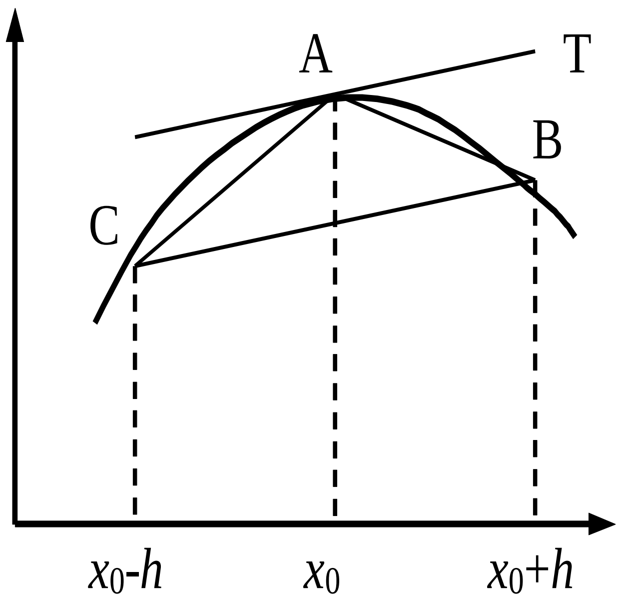
用 $y_n + hf(x_n, y_n)$ 替代，休恩方法

称上述公式为改进的Euler 公式（梯形公式）。

$$\boxed{\text{梯形公式} \int_a^b f(x)dx \approx \frac{b-a}{2}[f(a) + f(b)]}$$

注：它有局部截断误差 $R_{i+1}=y(x_{i+1})-y_{i+1}=O(h^3)$，即梯形格式具有2阶精度，比欧拉方法有了进步。但注意到该格式是隐式格式，计算时不得不用到迭代法，其迭代收敛性与欧拉格式相似。

3). 两步欧拉格式



中心差商近似导数 → $y'(x_1)\approx\dfrac{y(x_2)-y(x_0)}{2h}$

→ $y(x_2)\approx y(x_0)+2h\,f(x_1,y(x_1))$

$$y_{i+1}=y_{i-1}+2h\,f(x_i,y_i)\quad i=1,\ldots,n-1$$

假设 $y_{i-1}=y(x_{i-1})$，$y_i=y(x_i)$，则可以导出

$R_{i+1}=y(x_{i+1})-y_{i+1}=O(h^3)$

即两步Euler格式具有 2 阶精度。

需要2个初值 y_0 和 y_1 来启动递推过程，这样的算法称为两步法，而前面的三种算法都是单步法

例9.2：分别利用Euler公式、改进的Euler公式求解下列初值问题的数值解（取步长为$h=0.1$）

$$\begin{cases} \dfrac{dy}{dx} = y - \dfrac{2x}{y} & x \in (0,1) \\ y(0) = 1 \end{cases}$$

解： Euler公式：

$$y_{n+1} = y_n + hf(x_n, y_n) = 1.1y_n - \frac{0.2x_n}{y_n}$$

x_n	0.1	0.2	0.3	0.4	0.5
y_n	1.1000	1.1918	1.2774	1.3582	1.4351

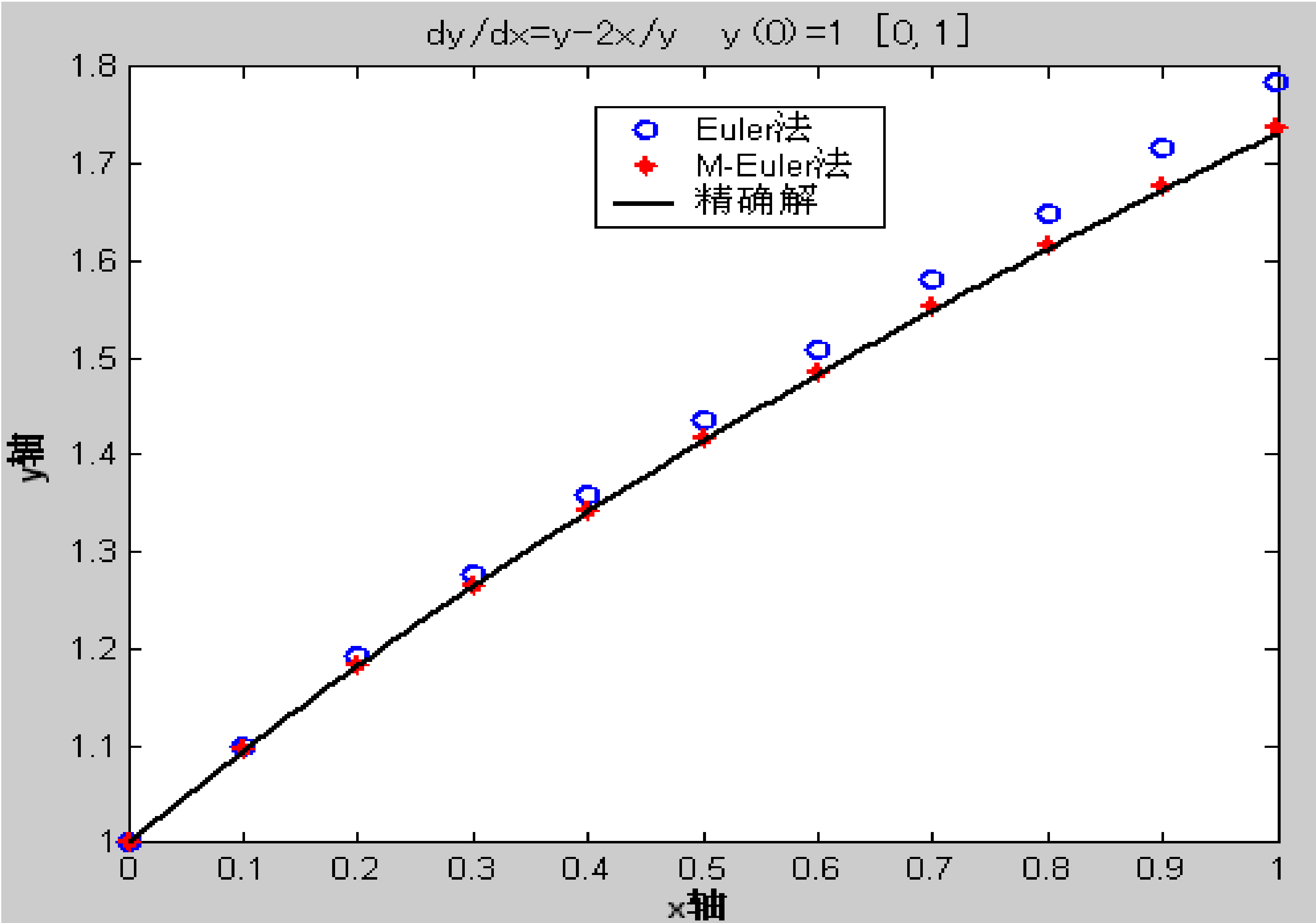
x_n	0.6	0.7	0.8	0.9	1.0
y_n	1.5090	1.5803	1.6498	1.7178	1.7848

改进的Euler公式：

$$\begin{cases} y_{n+1}^{(0)} = 1.1 y_n - \dfrac{0.2 x_n}{y_n} \\ y_{n+1} = y_n + \dfrac{0.1}{2}\left[\left(y_n - \dfrac{2x_n}{y_n}\right) + \left(y_{n+1}^{(0)} - \dfrac{2x_{n+1}}{y_{n+1}^{(0)}}\right)\right] \end{cases}$$

x_n	0.1	0.2	0.3	0.4	0.5
y_n	1.0959	1.1841	1.2662	1.3434	1.4164

x_n	0.6	0.7	0.8	0.9	1.0
y_n	1.4860	1.5525	1.6165	1.6782	1.7379



精确解 $y=\sqrt{-e^{x}+2x+2}$

程序 9.2(休恩方法) 通过计算

Matlab 实现

$$y_{k+1} = y_k + \frac{h}{2}(f(t_k, y_k) + f(t_{k+1}, y_k + hf(t_k, y_k)))$$

求$[a,b]$上的初值问题 $y' = (t-y), y(a) = y_0$ 的近似解,其中 $k = 0,1,\cdots,M-1$。

```
function H=heun(f,a,b,ya,M)

%Input   - y'= f is the function
%        - a and b are the left and right endpoints
%        - ya is the initial condition y(a)
%        - M is the number of steps
%Output - H=[T' Y'] where T is the vector of
%abscissas and Y is the vector of ordinates
h=(b-a)/M;
T=zeros(1,M+1);
Y=zeros(1,M+1);
T=a:h:b;
Y(1)=ya;
for j=1:M
   k1=f(T(j),Y(j));
   k2=f(T(j+1),Y(j)+h*k1);
   Y(j+1)=Y(j)+(h/2)*(k1+k2);
end

H=[T' Y'];
```

命令行窗口

```
>>  f=@(t,y)  (t*exp(t)-1)*sin(y);  a=1;  b=3;  ya=0.1;  M=20;
H=heun(f,a,b,ya,M)

H =

    1.0000    0.1000
    1.1000    0.1220
    1.2000    0.1584
    1.3000    0.2204
    1.4000    0.3313
    1.5000    0.5404
    1.6000    0.9442
    1.7000    1.6391
    1.8000    2.3620
    1.9000    2.7589
    2.0000    2.9400
    2.1000    3.0180
    2.2000    3.0465
    2.3000    3.0440
    2.4000    3.0033
    2.5000    2.8680
    2.6000    2.4178
    2.7000    1.5868
    2.8000    1.9624
    2.9000    3.6392
    3.0000    5.1344

fx >>
```

作业9.2

程序与实现

对于下面的第 1 题至第 5 题,用休恩方法求解微分方程。

(a) 令 $h=0.1$,程序 9.2 执行 20 步,然后令 $h=0.05$,程序 9.2 执行 40 步。

(b) 比较(a)中的两个近似解与精确解 $y(2)$。

(c) 当 h 减半时,(a)中的最终全局误差是否和预期相符?

(d) 将两个近似解和精确解画在同一坐标系中。提示:程序 9.2 输出的矩阵 `H` 是近似解的 x 和 y 坐标,命令 `plot(H(:,1),H(:,2))` 将画出与图 9.8 类似的图。

1. $y'=t^2-y,\quad y(0)=1,\quad y(t)=-e^{-t}+t^2-2t+2$

2. $y'=3y+3t,\quad y(0)=1,\quad y(t)=\frac{4}{3}e^{3t}-t-\frac{1}{3}$

方　法	优点	缺点
显式欧拉格式	简单	精度低
隐式欧拉格式	稳定性最好	精度低, 计算量大
梯形格式	精度提高	计算量大
两步欧拉格式	精度提高, 显式	多一个初值, 可能影响精度

能找到一个具有上面所有优点的格式吗?

9.4 龙格-库塔（Runge-Kutta）法

前面所讨论的欧拉法与改进的欧拉法都是一步法，即计算y_{n+1}时，只用到前一步的值，但精度却不一样，欧拉法是1阶的，而改进的欧拉法却是2阶的.

龙格-库塔(Runge-Kutta)法(简称R-K方法)是一类高精度的一步法，这类方法与泰勒展开级数法有着密切的关系.

9.4 龙格-库塔（Runge-Kutta）法

一、泰勒级数法

设有初值问题 $\begin{cases} \dfrac{dy}{dx} = f(x,y) \\ y(x_0) = y_0 \end{cases}$

由泰勒展开式

$$y(x_{n+1}) = y(x_n + h) = y(x_n) + hy'(x_n) + \frac{h^2}{2!}y''(x_n) + \cdots + \frac{h^k}{k!}y^{(k)}(x_n) + O(h^{k+1})$$

若令 $$y_{n+1} = y(x_n) + hy'(x_n) + \frac{h^2}{2!}y''(x_n) + \cdots + \frac{h^k}{k!}y^{(k)}(x_n) \quad (9.2)$$

则 $y(x_{n+1}) - y_{n+1} = O(h^{k+1})$

即公式(9.2)为k阶方法.

9.4 龙格-库塔（Runge-Kutta）法

R-*K*方法不是通过求导数的方法构造近似公式，而是通过计算不同点上的函数值，并对这些函数值作线性组合，构造近似公式，再把近似公式与解的泰勒展开式进行比较，使前面的若干项相同，从而使近似公式达到一定的阶数.

分析欧拉法

$$\begin{cases} y_{n+1} = y_n + k_1 \\ k_1 = hf(x_n, y_n) \end{cases}$$

每步计算f的值一次，其截断误差为$O(h^2)$.

分析改进的欧拉法

$$\begin{cases} y_{n+1} = y_n + \frac{1}{2}k_1 + \frac{1}{2}k_2 \\ k_1 = hf(x_n, y_n) \\ k_2 = hf(x_n + h, y_n + k_1) \end{cases}$$

每步计算f的值两次，其截断误差为$O(h^3)$.

上述两组公式在形式上共同点:

都是用$f(x, y)$在某些点上值的线性组合得出$y(x_{i+1})$的近似值y_{i+1},

且增加计算的次数$f(x, y)$的次数,可提高截断误差的阶。

如欧拉法：每步计算一次$f(x, y)$的值,为一阶方法。

改进欧拉法需计算两次$f(x, y)$的值，为二阶方法。

一种折中办法考虑用函数 $f(x, y)$在若干点上的函数值的线性组合来构造近似公式，构造时要求近似公式在(x_i, y_i)处的 *Taylor*展开式与解$y(x)$在x_i处的 *Taylor*展开式的前面几项重合，从而使近似公式达到所需要的阶数。

既避免求高阶导数,又提高了计算方法精度的阶数。或者说,在$[x_i, x_{i+1}]$这一步内多计算几个点的斜率值，然后将其进行加权平均作为平均斜率，则可构造出更高精度的计算格式，这就是龙格—库塔（*Runge-Kutta*）法的基本思想。

R-K方法不是直接使用Taylor级数,而是利用它的思想

一般龙格一库塔方法的形式为

$$\begin{cases} y_{i+1} = y_i + c_1 K_1 + c_2 K_2 + \cdots + c_p K_p \\ K_1 = hf(x_i, y_i) \\ K_2 = hf(x_i + a_2 h, y_i + b_{21} K_1) \\ \bullet \ldots \bullet\bullet \ldots \bullet\bullet \ldots \bullet\bullet \ldots \bullet\bullet \ldots \bullet\bullet \ldots \\ K_p = hf(x_i + a_p h, y_i + b_{p1} K_1 + \cdots + b_{p,p-1} K_{p-1}) \end{cases}$$

称为P阶龙格一库塔方法。

其中a_i, b_{ij}, c_i为待定参数，要求上式y_{i+1}在点(x_i, y_i)处作$Taylor$展开，通过相同项的系数确定参数。

课堂作业

9. **流行病模型**。流行病的数学模型描述如下：设有 L 个成员的构成的群落，其中有 P 个感染个体，Q 为未感染个体。令 $y(t)$表示时刻 t 感染个体的数量。对于温和的疾病，如普通感冒，每个个体保持存活，流行病从感染者传播到未感染者。由于两组间有 PQ 种可能的接触，$y(t)$的变化率正比于 PQ。故该问题可以描述为初值问题：

$$y' = ky(L - y) \qquad y(0) = y_0$$

(a) 用 $L = 25000$，$k = 0.00003$，$h = 0.2$ 和初值条件 $y(0) = 250$，并用程序 9.1 计算$[0,60]$上的欧拉近似解。

(b) 画出(a)中的近似解。

(c) 通过求(a)中欧拉方法的纵坐标平均值来估计平均感染个体的数目。

(d) 通过用曲线拟合(a)中的数据，并用定理 1.10(积分均值定理)，估计平均感染个体的数目。