《数值计算方法》

常微分方程数值解法

讲授人：杨 程
时　间：2021-2022学年
秋季学期
Email: yang_cheng@nun.edu.cn

目录

常微分方程数值解法

1 欧拉方法

2 龙格—库塔方法

3 线性多步法

在工程和科学技术的实际问题中，常需要解常微分方程。但常微分方程组中往往只有少数较简单和典型的常微分方程（例如线性常系数常微分方程等）可求出其解析解。对于变系数常微分方程的解析求解就比较困难，而一般的非线性常微分方程就更不用说了。在大多数情况下，常微分方程只能用近似法求解。

这种近似解法可分为两大类：一类是近似解析法，如级数解法、逐次逼近法等；另一类则是数值解法，它给出方程在一些离散点上的近似解。

在具体求解微分方程时，需要具备某种定解条件，微分方程和定解条件合在一起组成定解问题。

- 两类求定解问题
  - 实际中求解常微分方程的所谓定解问题有两类:初值问题和边值问题
  - 定解指已知因变量和/或其导数在某些点上是已知的(约束条件)。
  - 1. 边值问题

    约束条件为已知，在自变量的任一非初值上，已知函数值和/或其导数值，如

$$\begin{cases} y'' = f(x, y, y') \\ \boxed{y(a) = \alpha,\ y(b) = \beta} \end{cases} \quad \text{求解} y$$

常常可以**将边解问题转化为初值问题**求解。

- 2. 初值问题

约束条件为在自变量的 初值上已知函数值，如：

$$\begin{cases} y' = f(x,y) \\ \boxed{y(x_0) = y_0} \end{cases} \Rightarrow \begin{cases} dy/dx = f(x,y)\ x/x_0 \\ y(x_0) = y_0 \end{cases}$$

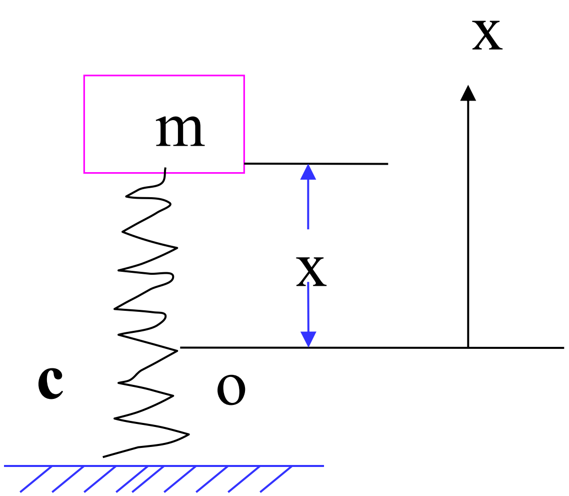
求解$y(x)$，以满足上述两式，即 在$a \le x_0' \le x_1' \cdots \le x_n' \le b$上的$y(x_i)$的近似值 $y_i = (i = 0,1,2,\cdots,n)$。

通常取等距节点，即 $h = x_{i+1} - x_i$，有

$$x_i = x_0 + ih \quad (i = 0,1,2,\cdots,n)$$

初值问题的数值解法特 点：按节点顺序依次推 进，由已知的$y_0, y_1, \cdots, y_i$，求出y_{i+1}，这可以通过递推公式 得到。

例如，弹簧-质量系统的振动问题（左图），作一定的简化后，可用一个二阶常微分方程来描述。



$$\frac{\mathrm{d}^2 x}{dt^2}+\frac{c}{m}x=0$$

式中，x是质量m离平衡位置（0点）的距离；t 是时间；c是弹簧常数。

当弹簧在振动开始时刻$t = t_0$时的初始位置$x(t_0) = x_0$和初速度：

$$\frac{\mathrm{d}x}{dt}\Big|_{t=t_0} = x'(t_0) = x'_0$$

确定时，弹簧的振动规律$x(t)$也就唯一确定。这就是一个常微分方程的**初值问题**，可写成：

$$\begin{cases} \dfrac{d^2x}{dt^2} + \dfrac{c}{m}x = 0 \\ x(t_0) = x_0 \qquad (t > t_0) \\ x'(t_0) = x'_0 \end{cases}$$

我们先从一阶常微分方程的初值问题：

$$\begin{cases} \dfrac{\mathrm{d}y}{dx} = f(x,y), x \in [a,b] \\ y(a) = \alpha \end{cases}$$

由常微分方程的理论知，只要上式中的函数$f(x,y)$在区域 $G=\{a \le x \le b, -\infty < y < \infty\}$ 内连续，且关于y满足**李普希兹(Lipschitz)条件**，即存在常数L(它与x, y无关)使：

$$|f(x,y_1) - f(x,y_2)| \le L|y_1 - y_2|$$

对 $G=\{a \le x \le b, -\infty < y < \infty\}$ 内任意两个y_1和 y_2 都成立，

则方程的解必定存在且唯一。

初值问题（如下式）的数值解法，常采用差分方法，即把一个连续的初值问题离散化为一个差分方程来求解。即将下式离散化后，求找其解$y = y(x)$在一系列离散节点：

$$a = x_0 < x_1 < \cdots < x_i < \cdots x_n = b$$

$$\boxed{\begin{cases} \dfrac{dy}{dx} = f(x, y) & a \le x \le b \\ y(x_0) = y_0 \end{cases}}$$

上的近似值$y_0, y_1, \ldots, y_n$。

两相邻节点间的距离

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

称为步长。当 $h_i = h$（常值）时称为等步长，有

$$x_i = x_0 + ih(i = 1,2,\cdots,n)$$

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

因为初值问题中的初始条件 $y(a)=y_0=\alpha$ 为已知，即可利用已知的 y_0 来求出下一节点处 $y(x_1)$ 的近似值 y_1，再从 y_1 来求出 $y_2,\cdots,$ 如此继续，直到求出 y_n 为止。这种用按节点的排列顺序一步一步地向前推进的方式求解的差分算法称为“步进式”或“递进式”算法。它是初值问题数值解法的各种差分格式的共同特点。因此，只要能写出由前几步已知信息

$$y_0, y_1, \cdots, y_i$$

来计算的递推公式（即差分格式），即可完全表达该种算法。

初值问题的常见解法

单步法：

利用前一个单步的信息(一个点)，在 $y=f(x)$ 上找下一点 y_i，

有欧拉法，龙格—库格法。

预测校正法：

多步法，利用一个以上的前点信息求 $f(x)$ 上的下一个 y_i，

常用迭代法，如改进欧拉法，阿当姆斯法。

- **数值解法含义**
  - 所谓数值解法, 就是设法将常微分方程离散化, 建立差分方程, 给出解在一些离散点上的**近似值**。
  - 微分方程的数值解: 设方程问题的解$y(x)$的存在区间是$[a,b]$, 令$a= x_0< x_1<\ldots< x_n =b$, 其中$h_k=x_{k+1}-x_k$, 如是等距节点$h=(b-a)/n$, h称为步长。
  - 由于$y(x)$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上$y(x_k)$的**近似值**, 用y_k**表示, 即**$y_k\approx y(x_k)$, 这样y_0, y_1, ..., y_n称为微分方程的数值解。

建立数值解法的常用方法

微分方程中含有导数项，建立微分方程数值解法, 首先要将微分方程离散化.（消除导数项）

一般采用以下几种方法:

（1）用差商近似导数

$$\left.\frac{dy}{dx}\right|_{(x_n,y_n)}=\frac{y(x_{n+1})-y(x_n)}{x_{n+1}-x_n}=f(x_n,y(x_n))$$

进一步：令 $y_n\approx y(x_n)$,带入上式右端，结果记为y_{n+1}

$$\underline{y_{n+1}\approx y_n+hf(x_n,y_n)}$$

(2) 用数值积分近似积分

$$\int_{x_n}^{x_{n+1}} \frac{dy}{dx}dx = \int_{x_n}^{x_{n+1}} f(x,y)dx \quad (n=0,1,\cdots)$$

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

进一步：令 $y_n \approx y(x_n)$

$$y_{n+1} - y_n \approx \int_{x_n}^{x_{n+1}} f(x,y(x))dx \approx h f(x_n, y_n)$$

宽

高

实际上是左矩形法

(3) 用Taylor多项式近似并可估计误差

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

进一步：令 $y_n \approx y(x_n)$

$$y_{n+1} \approx y_n + hf(x_n, y_n)$$

$$\left|y(x_{n+1}) - y_{n+1}\right| \le \frac{h^2}{2}\max_{a\le x\le b}\left|y''(x)\right|$$

PART 1

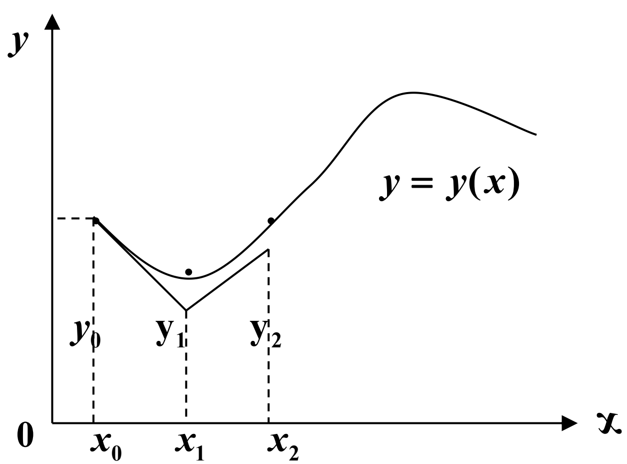
欧拉方法

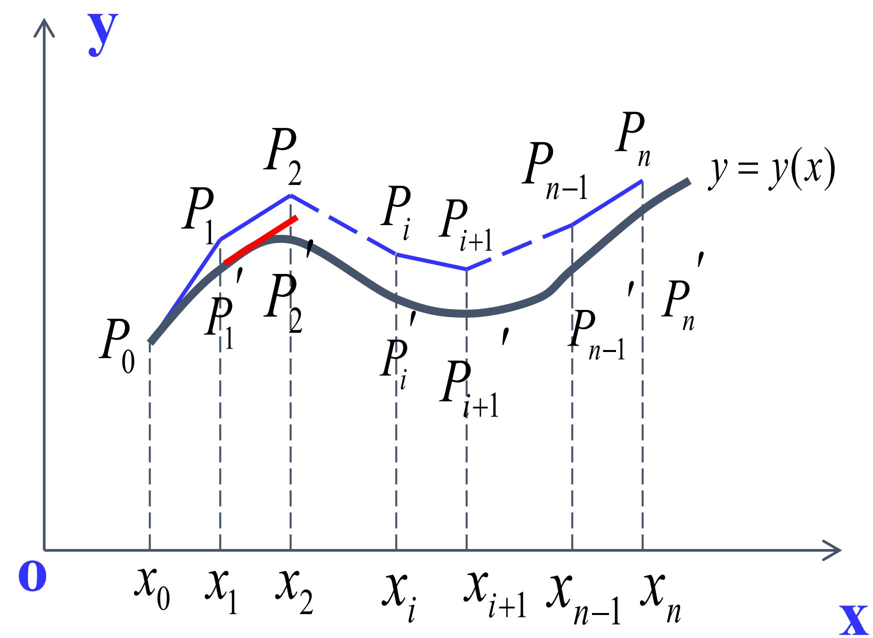
1、欧拉折线法

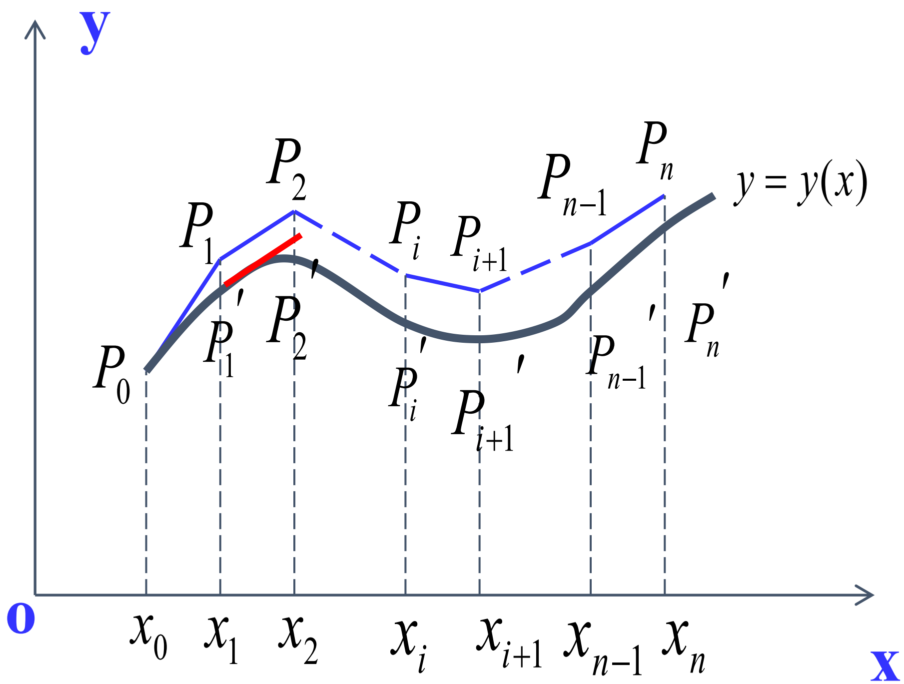
对于一阶常微分方程的初值问题（如下式）的解y=$y(x)$，在几何上是表示过点(x_0,y_0)的一条曲线。

$$\begin{cases} \dfrac{\mathrm{d}y}{\mathrm{d}x} = f(x,y) \\ y(a) = \alpha \end{cases} \quad (1)$$

欧拉折线法的数值解法是如右图所示：







首先，过点(x_0,y_0)作曲线$y=y(x)$的切线，由式（1）知该点切线斜率是$f(x_0,y_0)$，所以切线方程是：

$$y=y_0+f(x_0,y_0)(x-x_0)$$

它与直线$x=x_1$交点的纵坐标为：

$$y_1=y_0+f(x_0,y_0)(x_1-x_0)$$

则取 $y_1\approx y(x_1)$。然后，再过点（x_1,y_1）作斜率如下式的直线：

$$y=y_1+f(x_1,y_1)(x-x_1)$$

它与直线$x=x_2$交点的纵坐标为： $y_2=y_1+f(x_1,y_1)(x_2-x_1)$

依次做下去，如果已求出(x_n,y_n)，那么过这点且斜率是$f(x_n,y_n)$的直线方程为：

$$y = y_n + f(x_n, y_n)(x - x_n)$$

它与直线$x=x_{n+1}$交点的纵坐标为：

$$y_{\mathrm{n+1}} = y_{\mathrm{n}} + f(x_{\mathrm{n}}, y_{\mathrm{n}})(x_{\mathrm{n+1}} - x_{\mathrm{n}})$$

则取 $y_{\mathrm{n+1}} \approx y(x_{\mathrm{n+1}})$ 。

于是便得到处置问题的解$y(x)$在节点 $x_1, x_2, \cdots, x_n, x_{n+1}, \cdots$ 处的近视值 $y_1, y_2, \cdots, y_n, y_{n+1}, \cdots$

欧拉折线法的计算公式为：

$$\begin{cases} y_{n+1} = y_n + hf(x_n, y_n) \\ x_n = x_0 + nh \end{cases} \quad (n = 1,2,\cdots) \qquad （2）$$

这里$h=x_{n+1}-x_n$是常数。这就是**欧拉（Euler）公式**，又称**Euler格式**。

由以上计算过程可知，欧拉折线法的基本思想是：**用一系列直线所组成的折线去近似地代替曲线，并用这些直线交点处的纵坐标y_i作为精确解$y(x_i)$的近似值。**

常微分问题：

$$\begin{cases} \dfrac{\mathrm{d}y}{\mathrm{d}x} = f(x,y) \\ y(a) = \alpha \end{cases}$$

数值解计算公式：

$$\begin{cases} \boxed{y_{n+1} = y_n} + hf(x_n, y_n) \\ x_n = x_0 + nh \end{cases} \quad (n = 1,2,\cdots) \qquad （2）$$

$$\int_{x_n}^{x_{n+1}} \frac{dy}{dx}dx = \int_{x_n}^{x_{n+1}} f(x,y)dx \quad (n = 0,1,\cdots) \xrightarrow{\text{利用数值积分方法求解}}$$

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

所以公式（2）右端是用左矩形公式$hf(x_n,y(x_n))$近似，再以y_n代替$y(x_n)$，y_{n+1}代替$y(x_{n+1})$也得到（显式）欧拉公式。

如果右端积分用右矩形公式$hf(x_{n+1},y(x_{n+1}))$近似，则得到另一个公式：

$$y_{n+1}=y_n+hf(x_{n+1},y_{n+1})$$

称为(隐式)后退的欧拉公式.

后退的欧拉公式与欧拉公式有着本质的区别, 后者是关于y_{n+1}的一个直接计算公式，这类公式称作是**显式的**；前者公式的右端含有未知的y_{n+1}，它实际上是关于y_{n+1}的一个函数方程, 这类方程称作是**隐式的**。

显式与隐式两类方法各有特点，**考虑到数值稳定性等其他因素，人们有时需要选用隐式方法，但使用显式算法远比隐式方便。**

隐式方程通常用迭代法求解，而迭代过程的实质是逐步显式化。

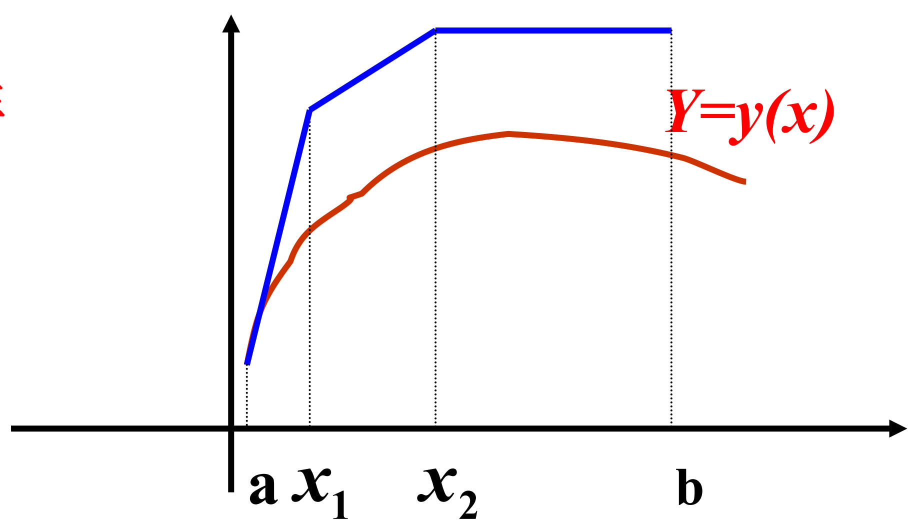
几何意义

由(x_0, y_0)出发，取解曲线$y=y(x)$的切线（存在！），则斜率

$$\left.\frac{dy}{dx}\right|_{(x_0,y_0)} = f(x_0, y_0)$$

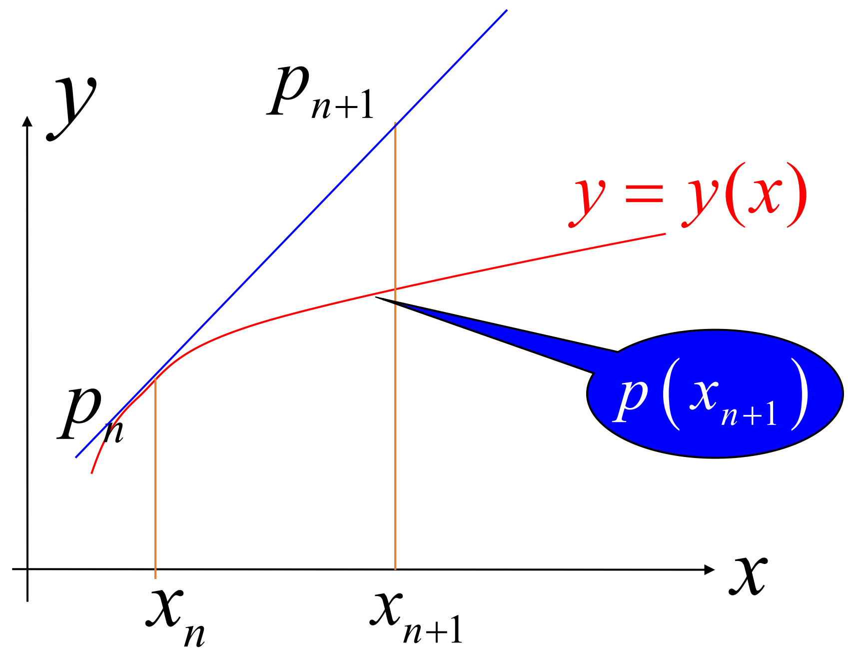
由于$f(x_0, y_0)$及(x_0, y_0)已知，必有切线方程。

注意：这是“折线法”而非“切线法”除第一个点是曲线切线外，其他点不是



还可以通过几何直观来考察欧拉方法的精度. 假设$y_n=y(x_n)$, 即顶点P_n落在积分曲线$y=y(x)$上，那么，

按欧拉方法做出的折线P_nP_{n+1}便是$y=y(x)$过点P_n的切线. 从图形上看, 这样定出的顶点P_{n+1}显著地偏离了原来的积分曲线，可见欧拉方法是相当粗糙的。



2、欧拉公式的截断误差

下面分析欧拉公式的截断误差。假设前述的初值问题在点x_{n+1}的精确解为$y(x_{n+1})$，y_{n+1}为欧拉公式（2）求得的相应的近似解，并**假设第n步求得的y_n是精确的，即$y_n=y(x_n)$**，则称：

$$y(x_{n+1})-y_{n+1}$$

是欧拉公式（2）的截断误差。

如果$y(x)$充分光滑，则由泰勒展开：

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

又由欧拉公式得：

$$y_{n+1} = y_n + hf(x_n, y_n) = y(x_n) + hy'(x_n)$$

由上两式可得：

$$\boldsymbol{y(x_{n+1}) - y_{n+1} = O\left(h^2\right)}$$

上式说明**欧拉公式（2）的截断误差是$O(h^2)$，它的精确度较差，一般很少用来求解近似值**，但是**欧拉折线法却体现了数值方法的基本思想**。

例 求解初值问题(步长$h=0.1$)

$$\begin{cases} y' = y - \dfrac{2x}{y} & (0 < x < 1) \\ y(0) = 1 \end{cases}$$

解 $f(x,y) = y - 2x/y$

Euler格式为:

$$\begin{cases} y_{n+1} = y_n + hf(x_n, y_n) = y_n + h(y_n - 2\dfrac{x_n}{y_n}) \\ y(0) = 1 \end{cases}$$

$$x_i = a + hi, i = 1,2,\cdots,10$$

$$x_1 = 0 + 0.1 * 1 = 0.1, x_2 = 0.2, \cdots, x_{10} = 1$$

$$\begin{cases} y_{n+1} = y_n + hf(x_n, y_n) = y_n + h(y_n - 2\dfrac{x_n}{y_n}) \\ y(0) = 1 \end{cases}$$

$$y_1 = y_0 + h\left(y_0 - 2\frac{x_0}{y_0}\right) = 1 + 0.1 * (1 - 0) = 1.1$$

$$y_2 = y_1 + h\left(y_1 - 2\frac{x_1}{y_1}\right) = 1.1 + 0.1 * \left(1.1 - 2 * \frac{0.1}{1.1}\right) \approx 1.1918$$

$$\vdots$$

• 计算结果为　　解析解：$y=\sqrt{1+2x}$

x_n	y_n	$y(x_n)$	$\lvert y_n-y(x_n)\rvert$
0.1	1.1	1.0954	0.0046
0.2	1.1918	1.1832	0.0086
0.3	1.2774	1.2649	0.0125
0.4	1.3582	1.3416	0.0166
0.5	1.4351	1.4142	0.0209
0.6	1.5090	1.4832	0.0257
0.7	1.5803	1.5492	0.0311
0.8	1.6498	1.6125	0.0373
0.9	1.7178	1.6733	0.0445
1.0	1.7848	1.7321	0.0527

误差较大
精度较低

例：用欧拉法解初值问 题

$$\begin{cases} \dfrac{dy}{dx} = 1 - xy \qquad (0 < x < 1) \\ y(0) = 0 \end{cases}$$

取步长$h = 0.2$。计算过程保留4位小数。

解：因为$f(x, y) = 1 - xy$，$h = 0.2$,

欧拉公式为：

$$y(x_{i+1}) \approx y_{i+1} = y_i + hf(x_i, y_i)$$

$$= y_i + 0.2(1 - x_i y_i) \quad (i = 0,1,2,3,4,5)$$

当i=0，即x_0=0时，

y_1=0+0.2(1-0×0)=0.2000

$$y_{i+1} = y_i + 0.2(1 - x_i y_i)$$

当i=1，即x_1=0.2时，

y_2=0.2+0.2(1-0.2×0.2)=0.3920

当i=2，即x_2=0.4时，

y_3=0.392+0.2(1-0.4×0.392)=0.56064

当i=3，即x_3=0.6时，

y_4=0.56064+0.2(1-0.6×0.56064)=0.6933632

当i=4，即x_4=0.8时，

y_4=0.6933632+0.2(10.8×0.6933632)=0.782425088

列表计算如下：

i	**0**	**1**	**2**	**3**	**4**	**5**
x_i	**0**	**0.2**	**0.4**	**0.6**	**0.8**	**1**
y_i	**0.0000**	**0.200**	**0.3920**	**0.5606**	**0.6934**	**0.7824**

例：利用*Euler*方法求初值问题

$$\begin{cases} y' = \dfrac{1}{1+x^2} - 2y^2, \quad 0 \le x \le 2 \\ y(0) = 0 \end{cases}$$

的数值解，此问题的精 确解是$y(x) = x/(1+x^2)$。

解：此时的*Euler*公式为

$$\begin{cases} y_{i+1} = y_i + h\left(\dfrac{1}{1+x_i^2} - 2y_i^2\right) \\ y_0 = 0 \quad ,i = 0,1,2\cdots \end{cases}$$

分别取步长$h = 0.2,0.1,0.05$，计算结果如下表：

h	x_i	y_i	$y(x_i)$	$y(x_i)-y_i$
h=0.2	0.00	0.00000	0.00000	0.00000
	0.40	0.37631	0.34483	-0.03148
	0.80	0.54228	0.48780	-0.05448
	1.20	0.52709	0.49180	-0.03529
	1.60	0.46632	0.44944	-0.01689
	2.00	0.40682	0.40000	-0.00682
h=0.1	0.00	0.00000	0.00000	0.00000
	0.40	0.36085	0.34483	-0.01603
	0.80	0.51371	0.48780	-0.02590
	1.20	0.50961	0.49180	-0.01781
	1.60	0.45872	0.44944	-0.00928
	2.00	0.40419	0.40000	-0.00419
h=0.05	0.00	0.00000	0.00000	0.00000
	0.40	0.35287	0.34483	-0.00804
	0.80	0.50049	0.48780	-0.01268
	1.20	0.50073	0.49180	-0.00892
	1.60	0.45425	0.44944	-0.00481
	2.00	0.40227	0.40000	-0.00227

3、欧拉方法的改进

为了提高欧拉方法的精确度，将微分方程（1）两边在区间$[x_0,x]$对x取积分，于是得到初值问题的等价积分方程：

$$y(x) = y_0 + \int_{x_0}^{x} f[t, y(t)]\mathrm{d}t$$

因此，求解$y(x)$就转化为计算上式右端的积分。

令x=x_1，得：

$$y(x_1) = y_0 + \int_{x_0}^{x_1} f[t, y(t)]\mathrm{d}t$$

令x=x_2，得：

$$\begin{aligned} y(x_2) &= y_0 + \int_{x_0}^{x_2} f[t, y(t)]\mathrm{d}t \\ &= y_0 + \int_{x_0}^{x_1} f[t, y(t)]\mathrm{d}t + \int_{x_1}^{x_2} f[t, y(t)]\mathrm{d}t \\ &= y(x_1) + \int_{x_1}^{x_2} f[t, y(t)]dt \end{aligned}$$

一般地有：

$$y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} f[t, y(t)]\mathrm{d}t \qquad (3)$$

为了求得$y(x_{n+1})$的近似值，只要用数值积分方法求出积分 $\int_{x_n}^{x_{n+1}} f[t, y(t)]\mathrm{d}t$ 的近似值就可以了。而选用不同的积分方法，将导出不同的计算公式。

例如用矩形公式计算得：

$$\int_{x_n}^{x_{n+1}} f[t, y(t)]\mathrm{d}t \approx hf[x_n, y(x_n)]$$

所以由式（3）有：

$$y(x_{n+1}) \approx y(x_n) + hf[x_n, y(x_n)]$$

若用y_{n+1}，y_n分别近似代替$y(x_{n+1})$，$y(x_n)$，则有计算公式：

$$y_{n+1} = y_n + hf(x_n, y_n)$$

这就是欧拉公式。

为了提高精度，该用梯形公式计算积分，即：

$$\int_{x_n}^{x_{n+1}} f[t,y(t)]\mathrm{d}t \approx \frac{1}{2}h\big[f\big(x_n,y(x_n)\big)+f\big(x_{n+1},y(x_{n+1})\big)\big]$$

所以由式（3）有：

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

若用y_{n+1}，y_n分别近似代替$y(x_{n+1})$，$y(x_n)$，则有计算公式：

$$y_{n+1}=y_n+\frac{1}{2}h[f(x_n,y_n)+f(x_{n+1},y_{n+1})] \qquad （4）$$

$$(n=0,1,2,\cdots)$$

用公式（4）求一阶微分方程初值问题的方法就称为梯形法则，该公式的精确度要比欧拉公式的精确度高。

欧拉公式：$y_{n+1} = y_n + \underline{hf(x_n, y_n)}$

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

- 由上面两个公式可知，这两种公式在计算上有根本的区别。
- 欧拉公式中，y_n是已知值或已算出的值，由y_n可以直接求出y_{n+1}，这种方法通常称为**显式方法**。
- 梯形法则中，y_{n+1}隐含在函数$f(x_{n+1},y_{n+1})$中，必须通过解方程才能求出，因此这种方法称为**隐式方法**。

中点欧拉公式

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

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

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

方 法		
显式欧拉	简单	精度低
隐式欧拉	稳定性最好	精度低，计算量大
梯形公式	精度提高	计算量大
中点公式	精度提高，显式	多一个初值， 可能影响精度

用梯形法则计算时，一般先用欧拉公式求出初始近似值，然后按迭代方法求解。其迭代公式为：

$$\begin{cases} y_{n+1}^{(0)} = y_n + hf(x_n, y_n) \\ y_{n+1}^{(k+1)} = y_n + \frac{h}{2}\left[f(x_n, y_n) + f\left(x_{n+1}, y_{n+1}^{(k)}\right)\right] \end{cases} \quad (5)$$

$$(n = 0,1,2,\cdots)$$

利用（5）求$y(x_{n+1})$的近似值y_{n+1}的方法称为改进的欧拉方法。

若函数$f(x,y)$连续，而且序列 $y_{n+1}^{(0)}, y_{n+1}^{(1)}, \cdots$ 收敛，那么它的极限必满足梯形法则（4），这时，序列的极限就是y_{n+1}。但是这样做**计算量太大**，因此实际计算时，**若步长h比较小，迭代一次的结果就取作y_{n+1}**，于是得计算公式：

$$\begin{cases} y_{n+1}^{(0)} = y_n + hf(x_n, y_n) \\ y_{n+1} = y_n + \frac{h}{2}\left[f(x_n, y_n) + f\left(x_{n+1}, y_{n+1}^{(0)}\right)\right] \end{cases} \quad (6)$$

上式通常称为**预报-校正公式**。其中，第一式称为预报公式，第二式称为校正公式。为了便于计算，将（6）式改为：

$$\begin{cases} y_{n+1}^{(0)} = y_n + hf(x_n, y_n) \\ y_{n+1} = y_n + \dfrac{h}{2}\left[f(x_n, y_n) + f\left(x_{n+1}, y_{n+1}^{(0)}\right)\right] \end{cases} \qquad （6）$$

令 $k_1 = hf(x_n, y_n)$ 则

$$y_{n+1}^{(0)} = y_n + k_1$$

$$y_{n+1} = y_n + \frac{1}{2}\left(k_1 + hf(x_{n+1}, y_n + k_1)\right)$$

再令 $k_2 = hf(x_n + h, y_n + k_1)$ 则

$$y_{n+1} = y_n + \frac{1}{2}(k_1 + k_2)$$

$$\begin{cases} y_{n+1} = y_n + \frac{1}{2}(k_1 + k_2) \\ k_1 = hf(x_n, y_n) \\ k_2 = hf(x_n + h, y_n + k_1) \end{cases} \quad (7)$$

下面分析公式(6)（即式(7)）的截断误差。**假设$y_n=y(x_n)$**，由于

$$k_1 = hf(x_n, y_n) = hf\big(x_n, y(x_n)\big) = hy'(x_n)$$

$$k_2 = hf(x_n + h, y_n + k_1) = hf(x_n + h, y(x_n) + k_1)$$

$$= h\left[f\big(x_n, y(x_n)\big) + h\frac{\partial}{\partial x}f\big(x_n, y(x_n)\big) + k_1\frac{\partial}{\partial y}f\big(x_n, y(x_n)\big) + \cdots\right]$$

$$= hy'(x_n) + h^2y''(x_n) + O(h^3)$$

将上述的 k_1 ，k_2 代入公式（7）的第一式，整理得：

$$y_{n+1} = y_n + hy'(x_n) + \frac{h^2}{2} y''(x_n) + O(h^3)$$

因此，预报-校正公式的截断误差为：

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

由此可见，它比欧拉折线法的截断误差提高了一阶。

由于数值积分的梯形公式的截断误差为 $O(h^3)$，因此，梯形法则的截断误差也是 $O(h^3)$ 。

预报-校正公式的截断误差与梯形法则是同阶的，然而预报-校正公式是显示的，便于计算。

为了便于比较各种方法的数值解与准确解之间的误差，从而看出各种方法的优劣，我们以常见的电路作为例子。

例　一阶RL暂态电路如下图所示。其中 $E_m = 311\mathrm{V}$ ，$\omega = 314\mathrm{rad}$

$R = 10\Omega$, $L = 500\mathrm{mH}$ 。求开关S合上后电流 $i(t)$ 在区间 $[0,0.01]$ 上的数值解。$h = 0.001$

解：根据电路理论得初值问题为：。

$$\begin{cases} \dfrac{\mathrm{d}i}{\mathrm{d}t} = \dfrac{E_m}{L}\sin\omega t - \dfrac{R}{L}i \\ i(0) = 0 \end{cases}$$

即 $$\begin{cases} \dfrac{\mathrm{d}i}{\mathrm{d}t} = 622\sin 314t - 20i \\ i(0) = 0 \end{cases}$$

（1）用欧拉折线法计算。计算公式如下：

$$\begin{cases} i_{n+1} = i_n + hf(t_n, i_n) \\ = i_n + 0.001 \times (622\sin 314t_n - 20i_n) \\ = 0.98i_n + 0.622\sin 314t_n \\ i_0 = 0 \end{cases}$$

（2）用预报-校正公式计算。计算公式如下：

$$\begin{cases} i_{n+1}^{(0)} = i_n + hf(t_n, i_n) = 0.98i_n + 0.622\sin 314t_n \\ i_{n+1} = i_n + \dfrac{h}{2}\left[f(t_n, i_n) + f\left(t_{n+1}, i_{n+1}^{(0)}\right)\right] \\ \qquad = 0.99i_n + 0.311(\sin 314t_n + \sin 314t_{n+1}) - 0.01i_{n+1}^{0} \\ i_0 = 0 \end{cases}$$

此初值问题的准确解是：

$$i(t) = \frac{E_m \omega L}{R^2 + (\omega L)^2}\left(e^{-\frac{R}{L}t} + \frac{R}{\omega L}\sin\omega t - \cos\omega t\right)$$

各公式计算结果如下表：

t_n	欧拉折线法 i_n	预报—校正公式 i_n	准确解 $i(t_n)$
0	0	0	0
0.001	0.192 11	0.096 05	0.096 20
0.002	0.553 71	0.371 01	0.372 88
0.003	1.045 67	0.794 25	0.799 22
0.004	1.616 20	1.320 73	1.329 83
0.005	2.205 87	1.895 38	1.909 23
0.006	2.753 49	2.458 5	2.477 20
0.007	3.202 04	2.951 58	2.974 74
0.008	3.504 24	3.323 04	3.349 80
0.009	3.627 21	3.533 23	3.562 35
0.01	3.555 66	3.558 36	3.588 35

由计算结果可以看出，预报—校正公式比欧拉折线法精确。

例 求解初值问题(步长$h=0.1$)

$$\begin{cases} y' = y - \dfrac{2x}{y} & (0 < x < 1) \\ y(0) = 1 \end{cases}$$

解 $f(x,y) = y - 2x/y$

改进Euler格式 $\begin{cases} y_p = y_n + hf(x_n, y_n) \\ y_c = y_n + hf(x_{n+1}, y_p) \\ y_{n+1} = \dfrac{1}{2}(y_p + y_c) \end{cases}$

	*Euler*法		改进*Euler*法		准确解
x_n	Y_n	$\lvert y_n-y(x_n)\rvert$	Y_n	$\lvert y_n-y(x_n)\rvert$	$y(x_n)$
0.1	1.1	0.0046	1.0959	0.0005	1.0954
0.2	1.1918	0.0086	1.1841	0.0009	1.1832
0.3	1.2774	0.0125	1.2662	0.0013	1.2649
0.4	1.3582	0.0166	1.3434	0.0018	1.3416
0.5	1.4351	0.0209	1.4164	0.0022	1.4142
0.6	1.5090	0.0257	1.4860	0.0028	1.4832
0.7	1.5803	0.0311	1.5525	0.0033	1.5492
0.8	1.6498	0.0373	1.6165	0.0040	1.6125
0.9	1.7178	0.0445	1.6782	0.0049	1.6733
1.0	1.7848	0.0527	1.7379	0.0058	1.7321

4、收敛性和稳定性

收敛性与相容性

数值解法的基本思想是，通过某种离散化手段将微分方程转化为差分方程，如单步法

$$y_{n+1}=y_n+h\phi(x_n,y_n,h)$$

它在点x_n处的解为y_n，而初值问题在点x_n处的精确解为$y(x_n)$，记$e_n=y(x_n)-y_n$称为**整体截断误差**. 收敛性就是讨论当 $x=x_n$ 固定且 $h=\dfrac{x_n-x_0}{n}\to 0$ 时$e_n\to 0$的问题.

绝对稳定性与绝对稳定域

前面关于收敛性的讨论有个前提，必须假定数值方法本身的计算是准确的. 实际情形并不是这样，差分方程的求解还会有计算误差. 譬如由于数字舍入而引起的小扰动. 这类小扰动在传播过程中会不会恶性增长，以至于“淹没”了差分方程的“真解”呢？这就是差分方程的稳定性问题. 在实际计算时，我们希望某一步产生的扰动值，在后面的计算中能够被控制，甚至是逐步衰减的.

例 用欧拉公式求解初值问题

$$\begin{cases} y' = -100y, \\ y(0) = 1. \end{cases}$$

其准确解 $y(x) = e^{-100x}$是一个按指数曲线衰减很快的函数.

$$\begin{cases} y_{n+1} = y_n + hf(x_n, y_n) \\ x_n = x_0 + nh \end{cases}$$

解 用欧拉法解方程$y'=-100y$ 得

$$y_{n+1} = y_n + h(-100y_n)$$

$$y_{n+1} = (1-100h)y_n$$

若取步长h=0.025，则欧拉公式的具体形式为

$$y_{n+1} = -1.5y_n.$$

计算结果见表，明显计算过程不稳定，但取h=0.005，y_{n+1}=0.5y_n，则计算过程稳定.

对后退的欧拉公式，取h=0.025时，则计算公式为y_{n+1}=−(1/3.5)y_n. 计算结果见表，这时计算过程是稳定的.

节点x_n	欧拉方法y_n	后退欧拉方法y_n
0.025	−1.5	0.2857
0.050	2.25	0.0816
0.075	−3.375	0.0233
0.100	5.0625	0.0067

例题表明稳定性**不但与方法有关，也与步长*h*有关**，当然与方程中的$f(x, y)$有关。

定义：

用某方法固定步长h作计算，由初值y_0严格得出y_i，观假设初值有微小误差δ_0（实际初值为$y_0+\delta_0$），则引起第i步计算值有误差δ_i（实际为y_i+S_i）。若

$$|\delta_i| \le |\delta_0| \quad (i=1,2,\cdots)$$

则称该方法关于步长h绝对稳定——$|\delta_i|$不随i无限扩大。

稳定性比较复杂。

为了只考察数值方法本身，通常只检验数值方法用于解**试验方程**的稳定性，**试验方程**为：

$$y' = \lambda y$$

常数，可以是复数

例：对$Euler$法

$$y_{i+1} = y_i + h(\lambda y_i) = (1+\lambda h) y_i = \cdots = (1+\lambda h)^{i+1} y_0$$

设初值有小扰动δ_0，则

$$y_{i+1} + \delta_{i+1} = (1+\lambda h)^{i+1} (y_0 + \delta_0)$$

与上式相减得

$$\delta_{i+1} = (1+\lambda h)^{i+1} \delta_0$$

显然，$Euler$方法绝对稳定$\Leftrightarrow |1+\lambda h| \le 1$，即

$h \le -\frac{1}{\lambda}$时$Euler$方法绝对稳定。

PART 2

龙格-库塔方法

欧拉方法和预报-校正方法的优点是算法简单，但是由于它们的精度比较低，不满足工程计算的要求，所以一般都不使用。

然而，分析它们算法的共同点，可从中得到启示，从而构造出精度较高的龙格-库塔（Runge-Kutta）方法。

将欧拉公式改写成：$\begin{cases} y_{n+1} = y_n + k \\ k = hf(x_n, y_n) \end{cases}$

其截断误差为$O(h^2)$，显然这是二阶的方法。

前面我们分析过，预报-校正公式比欧拉方法精度高一阶，为什么？

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

上面2个公式。左边是欧拉方法，右边是预报-校正公式。对比两式发现：

预报-校正公式计算y_{n+1}时，多计算了一次$f(x,y)$的值（即k_2），而且在计算k_2时又利用了已算出的k_1的信息，最后用$f(x,y)$在这两点(x_n,y_n)，(x_{n+1},y_{n+1})上的值的线性组合$(k_1+k_2)/2$作为的补偿，所以提高了精度。

因此，**如果设法多计算几点$f(x,y)$的值，并且用它们的线性组合作为y_n的补偿，精度还会提高**。基于这一设想，尝试构造如下形式的计算公式：

$$\begin{cases} y_{n+1} = y_n + \lambda_1 k_1 + \lambda_2 k_2 + \cdots + \lambda_m k_m \\ k_1 = hf(x_n, y_n) \\ k_2 = hf(x_n + \alpha_2 h, y_n + \beta_{21} k_1) \\ k_3 = hf(x_n + \alpha_3 h, y_n + \beta_{31} k_1 + \beta_{32} k_2) \\ \vdots \\ k_m = hf\left(x_n + \alpha_m h, y_n + \beta_{m1} k_1 + \cdots + \beta_{m,m-1} k_{m-1}\right) \end{cases} \quad (8)$$

其中，$\lambda_i, \alpha_i, \beta_{ij}$ 都是待定常数。公式（8）称为m阶**显式**龙格-库塔公式。

下面以几种龙格-库塔公式为例进行说明。

（1）二阶龙格-库塔公式

$$\begin{cases} y_{n+1} = y_n + \lambda_1 k_1 + \lambda_2 k_2 \\ k_1 = hf(x_n, y_n) \\ k_2 = hf(x_n + \alpha_2 h, y_n + \beta_{21} k_1) \end{cases} \qquad (9)$$

将$\boldsymbol{k_2}$在点$(\boldsymbol{x_n}, \boldsymbol{y_n})$作二元函数泰勒展开，得：

$$k_2 = hf(x_n, y_n) + \alpha_2 h^2 \frac{\partial f}{\partial x} + \beta_{21} k_1 h \frac{\partial f}{\partial y} + O(h^3)$$

其中，$\frac{\partial f}{\partial x}$，$\frac{\partial f}{\partial y}$ 都是在$\boldsymbol{x_n}, \boldsymbol{y_n}$处取值。将$\boldsymbol{k_2}$代入公式（9）的第一个式子可得：

$$
\begin{aligned}
&y_{n+1} \\
&= y_n + (\lambda_1 + \lambda_2) h y'(x_n) + \alpha_2 \lambda_2 h^2 \frac{\partial f}{\partial x} \qquad (10)\\
&+ \lambda_2 \beta_{21} k_1 h \frac{\partial f}{\partial y} + O(h^3)
\end{aligned}
$$

另一方面，将$y(x_{n+1})$在点x_n点作泰勒展开：

$$y(x_{n+1}) = y(x_n) + h y'(x_n) + \frac{h^2}{2!} y''(x_n) + O(h^3)$$

上式中 $y'(x_n)$ ，$y''(x_n)$ ，可用函数$\boldsymbol{f(x,y)}$表示为：

$$
\boxed{
\begin{aligned}
&y'(x_n) = f(x_n, y_n) \\
&y''(x_n) = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y}\frac{\mathrm{d}y}{\mathrm{d}x} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y} f(x_n, y_n)
\end{aligned}
}
$$

其中，$\dfrac{\partial f}{\partial x}$，$\dfrac{\partial f}{\partial y}$ 都是在x_n, y_n处取值。于是，

$$y(x_{n+1}) = y(x_n) + hy'(x_n) + \frac{h^2}{2!}\frac{\partial f}{\partial x} + \frac{h^2}{2!}f(x_n, y_n)\frac{\partial f}{\partial y} + O(h^3)$$

（11）

比较公式（10）和（11），在$y_n=y(x_n)$的假定下，要使得截断误差$y(x_{n+1}) - y_{n+1} = O(h^3)$，即数值求解公式（9）为二阶的，必须且只需满足公式（10）和公式（11）相同，即：

$$y_{n+1} = y_n + (\lambda_1 + \lambda_2)hy'(x_n) + \alpha_2\lambda_2 h^2\frac{\partial f}{\partial x} + \lambda_2\beta_{21}k_1 h\frac{\partial f}{\partial y} + O(h^3) \qquad (10)$$

$$k_1 = f(x_n, y_n)$$

$$\begin{cases} \boldsymbol{\lambda_1 + \lambda_2 = 1} \\ \boldsymbol{\alpha_2\lambda_2 = \dfrac{1}{2}} \\ \boldsymbol{\lambda_2\beta_{21} = \dfrac{1}{2}} \end{cases}$$

但是，上述方程组有4个未知数，却只有3个方程，因而不定方程组，根据线性代数的知识可知，它的解释无穷多。所以，**满足上述方程组的二阶龙格-库塔都具有二阶精度**。

如取 $\lambda_1 = \dfrac{1}{2}$，则 $\lambda_2 = \dfrac{1}{2}, \alpha_2 = \beta_{21} = 1$ ，此时是预报-校正公式：

$$\begin{cases} y_{n+1} = y_n + \lambda_1 k_1 + \lambda_2 k_2 \\ k_1 = hf(x_n, y_n) \\ k_2 = hf(x_n + \alpha_2 h, y_n + \beta_{21} k_1) \end{cases}$$

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

如取 $\lambda_1 = 0$ ，则 $\lambda_2 = 1, \alpha_2 = \beta_{21} = \dfrac{1}{2}$ ，又得：

$$\begin{cases} y_{n+1} = y_n + \lambda_1 k_1 + \lambda_2 k_2 \\ k_1 = hf(x_n, y_n) \\ k_2 = hf(x_n + \alpha_2 h, y_n + \beta_{21} k_1) \end{cases}$$

$$\Rightarrow \quad \begin{cases} y_{n+1} = y_n + k_2 \\ k_1 = hf(x_n, y_n) \\ k_2 = hf\left(x_n + \dfrac{1}{2} h, y_n + \dfrac{1}{2} k_1\right) \end{cases}$$

（2）四阶龙格-库塔公式

类似二阶龙格-库塔公式的推导方法，可得到实际中常用的四阶龙格-库塔公式：

$$\begin{cases} y_{n+1} = y_n + \dfrac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ k_1 = hf(x_n, y_n) \\ k_2 = hf\left(x_n + \dfrac{h}{2}, y_n + \dfrac{1}{2}k_1\right) \\ k_3 = hf\left(x_n + \dfrac{h}{2}, y_n + \dfrac{1}{2}k_2\right) \\ k_4 = hf(x_n + h, y_n + k_3) \end{cases}$$

其截断误差 $y(x_{n+1}) - y_{n+1} = O(h^5)$

四阶龙格-库塔法的**精度较高**，能满足一般工程设计的要求，另一方面，这种方法所需**编制的程序也较简单**。

但是，利用这种方法每计算一个y_{n+1}的值，需要反复计算四次$\boldsymbol{f(x,y)}$的值，**计算量较大**，而且**截断误差不容易估计**，这是它的不足之处。

这说明，**用到的节点越多，精度越高**。换句话说，**步长越小，截断误差越小**。但是在同一个区间[a,b]内，**计算的节点越多，计算量将增大，且还可能导致舍入误差严重累积**。

（3）变步长四阶龙格-库塔公式

因此，在满足精度的要求下，应考虑如何自动选择适当的步长。下面介绍变步长的四阶龙格-库塔方法。

假定从x_n出发，以h为步长，经过一步计算得$y(x_{n+1})$的近似值为 $y_{n+1}^{(h)}$，由于截断误差为$O(h^5)$，故有：

$$y(x_{n+1}) - y_{n+1}^{(h)} \approx ch^5 \qquad (12)$$

当h不大时，c可近似地看作常数。然后将步长折半，即取$h/2$为步长，从x_n出发经两步计算求得$y(x_{n+1})$的近似值 $y_{n+1}^{(h/2)}$。

由于每一步的截断误差是$c\left(\frac{h}{2}\right)^5$，因此有

$$y(x_{n+1}) - y_{n+1}^{(h/2)} \approx 2c\left(\frac{h}{2}\right)^5 \qquad (13)$$

将公式（12）和（13）相除得：

$$\frac{y(x_{n+1})-y_{n+1}^{(h/2)}}{y(x_{n+1})-y_{n+1}^{(h)}}\approx\frac{1}{16}$$

$$\Longrightarrow\quad 16\left(y(x_{n+1})-y_{n+1}^{(h/2)}\right)\approx y(x_{n+1})-y_{n+1}^{(h)}$$

整理得：$\boldsymbol{y(x_{n+1})-y_{n+1}^{(h/2)}\approx\frac{1}{15}\left[y_{n+1}^{(h/2)}-y_{n+1}^{(h)}\right]}$

上式说明，以 $y_{n+1}^{(h/2)}$ 作为$y(x_{n+1})$ 的近似值，其误差可用步长分半前后两次计算结果之差来表示。因此，若事先给定要求的精度ε 后，只需考察 $|y_{n+1}^{(h/2)}-y_{n+1}^{(h)}|<\varepsilon$ 是否成立。

如果成立，则将 $y_{n+1}^{(h/2)}$ 作为 $y(x_{n+1})$ 的近似值，否则，将步长再次折半进行计算，直到满足精度要求为止。

这种方法称为变步长的四阶龙格-库塔方法。

龙格-库塔法是“自开始的”，即可以直接利用给定的初始值 y_0，一步一步地算出 $y_1, y_2, \cdots, y_{n+1}$ 的值。这种方法称为线性单步法。

PART 3

线性多步法

单步法在计算y_{n+1}时，只用到前一步的信息y_n，为了提高精度，每一步都要增加计算非结点处的函数值，如龙格-库塔方法，计算量大。

事实上，在计算y_{n+1}之前，已经求得了一系列的近似值，如$y_n, y_{n-1}, \cdots, y_{n-r}$ 等，如果能充分利用这些信息来计算y_{n+1}，数值计算的结果可望具有较高的精度，这就是线性多步法的基本思想。

多步法中最常用的是**线性多步法**。构造多步法的途径一般有两种：**基于数值积分方法**和**基于泰勒展开方法**。前者可以直接由一般问题两端积分后，利用数值求积公式得到；而**泰勒展开方法的应用更广，它不仅可以得到求数值解的公式，而且容易估计阶段误差。**

前面介绍欧拉方法的改进时曾指处，初值问题

$$\begin{cases} \dfrac{dy}{dx} = f(x,y) & a \le x \le b \\ y(x_0) = y_0 \end{cases}$$

与积分方程：

$$y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} f[x, y(x)]\mathrm{d}x \quad (n = 0,1,2,\cdots) \quad （14）$$

是等价的，并且用梯形公式近似代替右端积分，可导出梯形法则：

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

$$(n = 0,1,2,\cdots)$$

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

梯形法则实际上是用节点 x_n 和 x_{n+1} 的线性插值函数代替被积函数 $f(x,y)$ 而得到的。

由此得到启发，**若用一个插值多项式 $P(x)$ 去近似代替被积函数 $f(x,y)$ ，然后用**

$$y_{n+1}=y_n+\int_{x_n}^{x_{n+1}}P(x)\mathrm{d}x$$

代替公式（14），若选取不同的插值点作插值多项式，相应地就会导出不同的数值解法。这就是构造阿达姆斯方法的基本思想。

1、一般线性多步法

➢由于在计算y_{n+1}时, 已经知道y_n, y_{n-1}, … , 及$f(x_n,y_n)$, $f(x_{n-1},y_{n-1})$, … , 利用这些值构造出精度高、计算量小的差分公式就是线性多步法。

➢即用**若干**节点处的 y 及 y' 值的**线性组合**来近似$y(x_{i+1})$。

$$f_j = f(x_j, y_j)$$

$$y_{i+1} = \alpha_0 y_i + \alpha_1 y_{i-1} + \ldots + \alpha_k y_{i-k} + h(\beta_{-1} f_{i+1} + \beta_0 f_i + \beta_1 f_{i-1} + \ldots + \beta_k f_{i-k})$$

当 $\beta_{-1} \neq 0$ 时，为**隐式公式**; $\beta_{-1}=0$ 则为**显式公式**。

2、阿达姆斯外推公式

假定已算出 $y_{n-3}, y_{n-2}, y_{n-1}, y_n$ 的值，记$f_k = f(x_k, y_k)$，以这四个数据点(x_{n-3}, f_{n-3}), (x_{n-2}, f_{n-2}), (x_{n-1}, f_{n-1})，和 (x_n, f_n) 作插值多项式：

$$
\begin{aligned}
P(x) \\
&= \frac{(x - x_{n-1})(x - x_{n-2})(x - x_{n-3})}{(x_n - x_{n-1})(x_n - x_{n-2})(x_n - x_{n-3})} f_n \\
&+ \frac{(x - x_n)(x - x_{n-2})(x - x_{n-3})}{(x_{n-1} - x_n)(x_{n-1} - x_{n-2})(x_{n-1} - x_{n-3})} f_{n-1} \\
&+ \frac{(x - x_n)(x - x_{n-1})(x - x_{n-3})}{(x_{n-2} - x_n)(x_{n-2} - x_{n-1})(x_{n-2} - x_{n-3})} f_{n-2} \\
&+ \frac{(x - x_n)(x - x_{n-1})(x - x_{n-2})}{(x_{n-3} - x_n)(x_{n-3} - x_{n-1})(x_{n-3} - x_{n-2})} f_{n-3}
\end{aligned}
$$

用 $P(x) \approx f[x, y(x)]$，得 $y(x_{n+1})$ 的近似值 y_{n+1}，即

$$y_{n+1} = y(x_n) + \int_{x_n}^{x_{n+1}} p(x)\mathrm{d}x$$

为了便于计算上式右端的积分，作变量代换 $x = x_n + th$，

由于

$$x_n - x_{n-1} = x_{n-1} - x_{n-2} = x_{n-2} - x_{n-3} = h$$

所以

$$\frac{(x - x_{n-1})(x - x_{n-2})(x - x_{n-3})}{(x_n - x_{n-1})(x_n - x_{n-2})(x_n - x_{n-3})} f_n$$

$$= \frac{(x_n + th - x_{n-1})(x_n + th - x_{n-2})(x_n + th - x_{n-3})}{h \cdot 2h \cdot 3h} f_n$$

$$= \frac{(h + th)(2h + th)(3h + th)}{h \cdot 2h \cdot 3h} f_n$$

$$= \frac{(t + 1)(t + 2)(t + 3)}{6} f_n$$

$$\frac{(x-x_n)(x-x_{n-2})(x-x_{n-3})}{(x_{n-1}-x_n)(x_{n-1}-x_{n-2})(x_{n-1}-x_{n-3})}f_{n-1} \qquad \boldsymbol{x = x_n + th}$$

$$= \frac{(x_n+th-x_n)(x_n+th-x_{n-2})(x_n+th-x_{n-3})}{-h\cdot(-2h)\cdot(-3h)}f_{n-1}$$

$$= \frac{th(2h+th)(3h+th)}{-h\cdot 2h\cdot 3h}f_{n-1}$$

$$=-\frac{t(t+2)(t+3)}{6}f_{n-1}$$

$$\frac{(x-x_n)(x-x_{n-1})(x-x_{n-3})}{(x_{n-2}-x_n)(x_{n-2}-x_{n-1})(x_{n-2}-x_{n-3})}f_{n-2} = \frac{th(h+th)(3h+th)}{-2h\cdot(-h)\cdot h}f_{n-2}$$

$$= \frac{t(t+1)(t+3)}{2}f_{n-2}$$

$$\frac{(x-x_n)(x-x_{n-1})(x-x_{n-2})}{(x_{n-3}-x_n)(x_{n-3}-x_{n-1})(x_{n-3}-x_{n-2})}f_{n-3} = \frac{th(h+th)(2h+th)}{-3h\cdot(-2h)\cdot(-h)}f_{n-3}$$

$$=-\frac{t(t+1)(t+2)}{6}f_{n-3}$$

$$P(x)=\frac{(t+1)(t+2)(t+3)}{6}f_n-\frac{t(t+2)(t+3)}{6}f_{n-1}$$
$$+\frac{t(t+1)(t+3)}{2}f_{n-2}-\frac{t(t+1)(t+2)}{6}f_{n-3}$$

由 $\boldsymbol{x=x_n+th}$ 可得： $\mathrm{d}x=h\mathrm{d}t\quad[x_n,x_{n+1}]\to[0,1]$

则 $y_{n+1}=y_n+\int_{x_n}^{x_{n+1}}P(x)\mathrm{d}x$ 转化为：

$$y_{n+1}=y(x_n)+\frac{h}{6}f_n\int_0^1(t+1)(t+2)(t+3)\mathrm{d}t$$
$$-\frac{h}{2}f_{n-1}\int_0^1 t(t+2)(t+3)\mathrm{d}t$$
$$+\frac{h}{2}f_{n-2}\int_0^1 t(t+1)(t+3)\mathrm{d}t$$
$$-\frac{h}{6}f_{n-3}\int_0^1 t(t+1)(t+2)\mathrm{d}t$$

$$y_{n+1} = y(x_n) + \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3})$$

上式中用 y_n 代替 $y(x_n)$ ，即得：

$$y_{n+1} = y_n + \frac{h}{24}[55f(x_n, y_n) - 59f(x_{n-1}, y_{n-1})$$

$$+37f(x_{n-2}, y_{n-2}) - 9f(x_{n-3}, y_{n-3})] \quad （15）$$

公式（15）称为**阿达姆斯外推公式**，它是一个**显式公式**。用该公式求数值解 y_{n+1} 时，必须用到前四步的数值，因此通常称这种数值方法为线性多步法。

阿达姆斯外推公式的截断误差

假定 $y_n=y(x_n), y_{n-1}=y(x_{n-1}), y_{n-2}=y(x_{n-2}), y_{n-3}=y(x_{n-3})$

由公式（15）及泰勒展开公式得：

$$\left.\begin{aligned} y_{n+1} &= y_n + \frac{h}{24}[55f(x_n, y_n) - 59f(x_{n-1}, y_{n-1}) \\ &\quad +37f(x_{n-2}, y_{n-2}) - 9f(x_{n-3}, y_{n-3})] \\ &\text{一阶常微分问题一般形式：}\ \frac{\mathrm{d}y}{\mathrm{d}x} = f(x, y) \end{aligned}\right\}$$

$$y_{n+1} = y_n + \frac{h}{24}\left(55y'_n - 59y'_{n-1} + 37y'_{n-2} - 9y'_{n-3}\right) \quad (16)$$

$$y_{n-1} = y_n - hy'_n + \frac{h^2}{2}y''_n - \frac{h^3}{6}y'''_n + \frac{h^4}{24}y_n^{(4)} + \cdots$$

$$y'_{n-1} = y'_n - hy''_n + \frac{h^2}{2}y'''_n - \frac{h^3}{6}y_n^{(4)} + \frac{h^4}{24}y_n^{(5)} + \cdots$$

同理：

$$y'_{n-2} = y'_n - 2hy''_n + \frac{4h^2}{2}y'''_n - \frac{8h^3}{6}y_n^{(4)} + \frac{16h^4}{24}y_n^{(5)} + \cdots$$

$$y'_{n-3} = y'_n - 3hy''_n + \frac{9h^2}{2}y'''_n - \frac{27h^3}{6}y_n^{(4)} + \frac{81h^4}{24}y_n^{(5)} + \cdots$$

$$y_{n+1} = y_n + \frac{h}{24}(55y'_n - 59y'_{n-1} + 37y'_{n-2} - 9y'_{n-3})$$

$$= y_n + hy'_n + \boxed{\frac{1}{2}h^2y''_n} + \boxed{\frac{1}{6}h^3y'''_n} + \boxed{\frac{1}{24}h^4y_n^{(4)}}$$

$$\boxed{-\frac{49}{144}h^5y_n^{(5)}} + \cdots \quad （17）$$

$$y_{n+1} = y_n + hy'_n + \frac{1}{2}h^2y''_n + \frac{1}{6}h^3y'''_n + \frac{1}{24}h^4y_n^{(4)} - \frac{49}{144}h^5y_n^{(5)} + \cdots \qquad (17)$$

又由泰勒展开得：

$$y_{n+1} = y_n + hy'_n + \frac{1}{2}h^2y''_n + \frac{1}{6}h^3y'''_n + \frac{1}{24}h^4y_n^{(4)} + \frac{1}{120}h^5y_n^{(5)} + \cdots \qquad (18)$$

由公式（17）和（18）得截断误差：

$$y(x_{n+1}) - y_{n+1} = \frac{251}{720}h^5y_n^{(5)} + \cdots = O(h^5)$$

这说明阿达姆斯外推公式是四阶的方法。

2、阿达姆斯内推公式

若以 $x_{n-2}, x_{n-1}, x_n, x_{n+1}$ 作为插值节点，过数据点(x_{n-2}, f_{n-2}) ,(x_{n-1}, f_{n-1}) ,(x_n, f_n) ,(x_{n+1}, f_{n+1}) 作插值多项式：

$$
\begin{aligned}
&P(x)\\
&=\frac{(x-x_n)(x-x_{n-1})(x-x_{n-2})}{(x_{n+1}-x_n)(x_{n+1}-x_{n-1})(x_{n+1}-x_{n-2})}f_{n+1}\\
&+\frac{(x-x_{n+1})(x-x_{n-1})(x-x_{n-2})}{(x_n-x_{n+1})(x_n-x_{n-1})(x_n-x_{n-2})}f_n\\
&+\frac{(x-x_{n+1})(x-x_n)(x-x_{n-2})}{(x_{n-1}-x_{n+1})(x_{n-1}-x_n)(x_{n-1}-x_{n-2})}f_{n-1}\\
&+\frac{(x-x_{n+1})(x-x_n)(x-x_{n-1})}{(x_{n-2}-x_{n+1})(x_{n-2}-x_n)(x_{n-2}-x_{n-1})}f_{n-2}
\end{aligned}
$$

用 $P(x) \approx f[x, y(x)]$，得 $y(x_{n+1})$ 的近似值 y_{n+1}，即

$$y_{n+1} = y(x_n) + \int_{x_n}^{x_{n+1}} p(x)\mathrm{d}x$$

作变量代换 $x = x_n + th$，由于

$$x_{n+1} - x_n = x_n - x_{n-1} = x_{n-1} - x_{n-2} = h$$

$$\begin{aligned} y_{n+1} = y(x_n) &+ \frac{h}{6} f_{n+1} \int_0^1 t(t+1)(t+2)\mathrm{d}t \\ &- \frac{h}{2} f_n \int_0^1 (t-1)(t+1)(t+2)\mathrm{d}t \\ &+ \frac{h}{2} f_{n-1} \int_0^1 (t-1)t(t+2)\mathrm{d}t \\ &- \frac{h}{6} f_{n-2} \int_0^1 (t-1)t(t+1)\mathrm{d}t \end{aligned}$$

$$y_{n+1} = y(x_n) + \frac{h}{24}(9f_{n+1} + 19f_n - 5f_{n-1} + f_{n-2})$$

上式中用 y_n 代替 $y(x_n)$ ，即得阿达姆斯内插公式：

$$y_{n+1} = y_n + \frac{h}{24}[9f(x_{n+1}, y_{n+1}) + 19f(x_n, y_n)$$

$$+5f(x_{n-1}, y_{n-1}) + f(x_{n-2}, y_{n-2})]$$

由于上式右端**隐含y_{n+1}**，所以内插公式是隐式的。同样，假定 $y_n = y(x_n), y_{n-1} = y(x_{n-1}), y_{n-2} = y(x_{n-2})$ ，可得阿达姆斯内插公式的截断误差：

$$y(x_{n+1}) - y_{n+1} = -\frac{19}{720}h^5 y_n^{(5)} + \cdots = O(h^5)$$

阿达姆斯内插公式的优点是计算比较稳定，但计算y_{n+1}需要用迭代法，因此，就像处理改进的欧拉公式一样，**由外推公式提供初值，然后由内插公式迭代一次所得的结果作为y_{n+1}**，由此得到**阿达姆斯预报-校正公式**：

$$\begin{cases} y_{n+1}^{(0)} = y_n + \dfrac{h}{24}[55f(x_n, y_n) - 59f(x_{n-1}, y_{n-1}) \\ \qquad +37f(x_{n-2}, y_{n-2}) - 9f(x_{n-3}, y_{n-3})] \\ y_{n+1} = y_n + \dfrac{h}{24}[9f\left(x_{n+1}, y_{n+1}^{(0)}\right) + 19f(x_n, y_n) \\ \qquad -5f(x_{n-1}, y_{n-1}) + f(x_{n-2}, y_{n-2})] \end{cases}$$

知识结构图

- 常微分方程初值问题数值解法
  - 单步法
    - 重要概念（截断误差、方法精度、收敛性、相容性、绝对稳定性等）
    - 主要方法
      - 欧拉方法
      - 梯形方法
      - 龙格-库塔法（包括改进的欧拉法）
  - 线性多步法
    - 构造方法（数值积分法、泰勒展开法）
    - 主要方法
      - 阿达姆斯显式与隐式方法
      - 米尔尼方法与辛普森方法
      - 汉明方法
      - 预测-校正方法
  - 方程组与高阶方程

20XXPOWERPOINT

感谢观看 THANGKS!

单击此处添加您的副标题文字