Numerical Computing Methods
数值计算方法

LI, Xiao Peng
李晓鹏

(Email: x.p.li@szu.edu.cn)

Shenzhen University

27 Nov. 2023

常微分方程数值解法

1. 引言

2. Runge-Kutta法

1 引言

2 Runge-Kutta法

引言

微分方程定义

- 含有未知函数的导数 (微分) 的方程, 称为微分方程
- 未知函数为一元函数的微分方程, 称为常微分方程
- 未知函数为多元函数的微分方程, 称为偏微分方程

给定微分方程及其初始条件, 称为初值问题:

$$\begin{cases} y' = f(x, y), \quad a \leq x \leq b \\ y(a) = y_0 \end{cases}$$

给定微分方程及其边界条件, 称为边值问题:

$$\begin{cases} y'' = f(x, y, y'), \quad a \leq x \leq b \\ y(a) = y_0 \\ y(b) = y_N \end{cases}$$

本章主要讨论常微分方程的**初值问题**。

引言

问题: 实际中需要求解常微分方程的初值问题

$$\begin{cases} y' = f(x, y), & a \le x \le b \\ y(a) = y_0 \end{cases} \tag{5.1}$$

上述方程等价于积分方程

$$y(x) = y_0 + \int_a^x f(t, y(t))\mathrm{d}t$$

引言

当 $f(x,y)$ 在矩形区域 $a \leq x \leq b, c \leq y \leq d$ 中连续, 并且关于变量 y 满足 Lipschitz 条件, 即对任意 $x \in [a,b], y \in [c,d]$ 都有

$$\left|f\left(x, y_1\right)-f\left(x, y_2\right)\right| \leq L\left|y_1-y_2\right|$$

其中 L 为有限常数, 则初值问题的解存在且唯一。

所谓常微分方程的数值解：就是求 $y\left(x_n\right)$ 在区间 $[a,b]$ 中一系列离散点（或节点）

$$a<x_1<x_2<\cdots<x_N \leq b$$

上 $y\left(x_n\right)$ 的近似值 $y_n \ (n=1,2,\cdots,N)$, 这些近似值就是 (5.1)的数值解。

求解计算这些近似值的方法称为该方程数值方法。

由数值方法计算所得的数值解统称为计算解。

引言

在计算时，从初始条件 $y(a) = y_0$ 出发，先求 y_1，再从 y_0, y_1 求出 y_2，以此递推求出所有

$$y_n, \quad (n = 1, 2, \cdots, N),$$

这种顺着节点的排列顺序一步步地向前推进的求解方法，称为步进法。

如果在计算 y_n 时只利用了 y_{n-1}，这种方法称为单步法。

如果在计算 y_n 时利用了 $y_{n-1}, y_{n-2}, \cdots, y_{n-p}$，这种方法称为多步法。

用数值方法求解常微分方程时，往往采用等分区间取节点的方法对数学问题进行数值化，即将求解区间 $[a, b]$ 分成 N 等份：

$$\frac{b-a}{N} = h, x_n - x_{n-1} = h, \quad (n = 1, 2, \cdots, N)$$

Euler方法

将 $y(x_{n+1})$ 在点 x_n 处进行 Taylor 展开

$$y\left(x_{n+1}\right)=y\left(x_n\right)+hy'\left(x_n\right)+\frac{h^2}{2!}y''\left(\xi_n\right),\quad \xi_n\in\left[x_n,x_{n+1}\right]$$

略去 h^2 项：

$$y\left(x_{n+1}\right)\approx y\left(x_n\right)+hf(x_n,y(x_n))$$

然后用 y_n 代替 $y(x_n)$，即得

$$\begin{aligned}
y_{n+1}&=y_n+hf(x_n,y_n),\quad (n=0,1,2\cdots,N-1)\\
e_{n+1}(h)&=\frac{h^2}{2!}y''\left(\xi_n\right)\approx\frac{h^2}{2!}y''\left(x_n\right)
\end{aligned}$$

称上述公式为前进 Euler 公式。

Euler方法

将 $y(x_n)$ 在点 x_{n+1} 处进行 Taylor 展开

$$y\left(x_n\right)=y\left(x_{n+1}\right)-hy'\left(x_{n+1}\right)+\frac{h^2}{2!}y''\left(\xi_n\right), \quad \xi_n\in\left[x_n,x_{n+1}\right]$$

略去 h^2 项：

$$y\left(x_{n+1}\right)\approx y\left(x_n\right)+hf(x_{n+1},y(x_{n+1}))$$

然后用 y_n 代替 $y(x_n)$，即得

$$y_{n+1}=y_n+hf(x_{n+1},y_{n+1}), \quad (n=0,1,2\cdots,N-1)$$

$$e_{n+1}(h)=-\frac{h^2}{2!}y''\left(\xi_n\right)\approx-\frac{h^2}{2!}y''\left(x_{n+1}\right)$$

称上述公式为后退 Euler 公式。

注意：后退 Euler 公式是非显式的。

例题

例：取 $h=0.1$, 使用前进 Euler 公式、后退Euler 公式求解初值问题

$$\begin{cases} y' = y - \frac{2x}{y}, \quad x \in [0,1] \\ y(0) = 1 \end{cases}$$

例题

例：取 $h=0.1$, 使用前进 Euler 公式、后退Euler 公式求解初值问题

$$\begin{cases} y' = y - \frac{2x}{y}, \quad x \in [0,1] \\ y(0) = 1 \end{cases}$$

解: 已知 $x_0 = a = 0, y_0 = 1, N = 1/h = 10,$

$$f(x,y) = y - 2x/y$$

利用 前进 Euler 法求解：

$$y_1 = y_0 + hf(x_0, y_0) = y_0 + h\left(y_0 - \frac{2x_0}{y_0}\right) = 1.1000$$

$$y_2 = y_1 + hf(x_1, y_1) = y_1 + h\left(y_1 - \frac{2x_1}{y_1}\right) \approx 1.1918$$

$$\vdots$$

$$y_{10} = y_9 + hf(x_9, y_9) = y_9 + h\left(y_9 - \frac{2x_9}{y_9}\right) \approx 1.7848$$

例题

例：取 $h=0.1$, 求解初值问题

$$\begin{cases} y' = y - \frac{2x}{y}, & x \in [0,1] \\ y(0) = 1 \end{cases}$$

解: 已知 $x_0 = a = 0, y_0 = 1, N = 1/h = 10,$

$$f(x,y) = y - 2x/y$$

利用 后退 Euler 法求解：

$$y_1 = y_0 + hf(x_1, y_1) = y_0 + h\left(y_1 - \frac{2x_1}{y_1}\right)$$

$$\Rightarrow y_1 = \frac{y_0 + (y_0^2 - 0.72x_1)^{1/2}}{1.8} \approx 1.0907$$

$$\vdots$$

显式与隐式结合方法

$$y_{n+1} = y_n + hf\left(x_n, y_n\right) \qquad \text{前进 Euler 求解公式}$$

$$y_{n+1} = y_n + hf\left(x_{n+1}, y_{n+1}\right) \qquad \text{后退 Euler 求解公式}$$

$$\begin{cases} \bar{y}_{n+1} = y_n + hf\left(x_n, y_n\right) & \text{预测值} \\ y_{n+1} = y_n + hf\left(x_{n+1}, \bar{y}_{n+1}\right) & \text{校正值} \\ y\left(x_0\right) = y_0, \end{cases}$$

$$y_0 \rightarrow \left(\bar{y}_1\right) \rightarrow y_1 \rightarrow \left(\bar{y}_2\right) \rightarrow y_2 \rightarrow \ldots \rightarrow \left(\bar{y}_N\right) \rightarrow y_N$$

这种求解方法称为预测-校正系统。

例题

例：取 $h = 0.1$, 利用预测-校正系统求解初值问题

$$\begin{cases} y' = y - \frac{2x}{y}, \quad x \in [0,1] \\ y(0) = 1 \end{cases}$$

例题

例：取 $h=0.1$, 利用预测-校正系统求解初值问题

$$\begin{cases} y' = y - \frac{2x}{y}, & x \in [0,1] \\ y(0) = 1 \end{cases}$$

解: 已知 $x_0 = a = 0, y_0 = 1, N = 1/h = 10,$

$$f(x,y) = y - 2x/y$$

$$\bar{y}_1 = y_0 + hf(x_0, y_0) = y_0 + h\left(y_0 - \frac{2x_0}{y_0}\right) = 1.1000$$

$$y_1 = y_0 + hf(x_1, \bar{y}_1) = y_0 + h\left(\bar{y}_1 - \frac{2x_1}{\bar{y}_1}\right) = 1.0918$$

$$\bar{y}_2 = y_1 + hf(x_1, y_1) = y_1 + h\left(y_1 - \frac{2x_1}{y_1}\right) \approx 1.1827$$

$$y_2 = y_1 + hf(x_1, y_1) = y_1 + h\left(\bar{y}_2 - \frac{2x_2}{\bar{y}_2}\right) \approx 1.1763$$

基于数值积分的求解公式

因为

$$\int_{x_{n-1}}^{x_n} f(x, y(x)) \mathrm{d}x = \int_{x_{n-1}}^{x_n} y'(x) \mathrm{d}x = y\left(x_n\right) - y\left(x_{n-1}\right)$$

因此

$$y\left(x_n\right) = y\left(x_{n-1}\right) + \int_{x_{n-1}}^{x_n} f(x, y(x)) \mathrm{d}x \tag{5.2}$$

若 $y\left(x_{n-1}\right)$ 已知, 则由 (5.2)可求出 $y\left(x_n\right)$。

若取近似, 利用矩形求积得

$$\begin{aligned} y\left(x_n\right) &= y\left(x_{n-1}\right) + \int_{x_{n-1}}^{x_n} f(x, y(x)) \mathrm{d}x \\ &\approx y\left(x_{n-1}\right) + h f\left(x_{n-1}, y\left(x_{n-1}\right)\right) \end{aligned}$$

上式即为 Euler 求解公式, 故又可称为矩形法。

基于数值积分的求解公式

若在 (5.2)中, 利用梯形公式:

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

则

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

令

$$y_n=y_{n-1}+\frac{h}{2}\left(f\left(x_{n-1}, y_{n-1}\right)+f\left(x_n, y_n\right)\right)$$

上式称为梯形求解公式，简称梯形法。

问题：几何解释？
问题：梯形法是显式还是隐式？

基于数值积分的求解公式

梯形法是隐式, 故可以采用预测–校正系统

$$\begin{cases} \bar{y}_n = y_{n-1} + hf\left(x_{n-1}, y_{n-1}\right) \\ y_n = y_{n-1} + \frac{h}{2}\left[f\left(x_{n-1}, y_{n-1}\right) + f\left(x_n, \bar{y}_n\right)\right] \\ y_0 = x_0 \end{cases} \tag{5.3}$$

上式又称为改进的 Euler 求解公式, 简称改进 Euler 法。

紧凑形式:

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

例题

例：用 Euler 法、梯形法以及改进 Euler 法求解

$$\begin{cases} y' = -y + x + 1 \\ y(0) = 1 \end{cases}$$

其中,步长 $h = 0.1$, 计算至 $x = 0.5$。

例题

例：用 Euler 法、梯形法以及改进 Euler 法求解

$$\begin{cases} y' = -y + x + 1 \\ y(0) = 1 \end{cases}$$

其中,步长 $h = 0.1$, 计算至 $x = 0.5$。

解: 设

$$f(x, y) = -y + x + 1,$$
$$x_0 = a = 0, b = 0.5, y_0 = 1, n = 5,$$

1) Euler法: 计算公式为

$$\begin{aligned} y_n &= y_{n-1} + h\left(x_{n-1} - y_{n-1} + 1\right) \\ &= hx_{n-1} + (1-h)y_{n-1} + h \\ &= 0.1x_{n-1} + 0.9y_{n-1} + 0.1 \end{aligned}$$

所以

$$y_1 = 0.1x_0 + 0.9y_0 + 0.1 = 1.000$$

例题

2) 梯形法: 计算公式为

$$y_n = y_{n-1} + 0.5h\left[(x_{n-1} - y_{n-1} + 1) + (x_n - y_n + 1)\right]$$

解得

$$\begin{aligned} y_n &= \frac{(2-h)y_{n-1} + h\left(x_{n-1} + x_k\right) + 0.2}{2+h} \\ &= \frac{1.9y_{n-1} + 0.1\left(x_{n-1} + x_n\right) + 0.2}{2.1} \end{aligned}$$

所以

$$\begin{aligned} y_1 &= \frac{1.9y_0 + 0.1\left(x_0 + x_1\right) + 0.2}{2.1} \\ &= 1.004762 \end{aligned}$$

例题

3）改进Euler法：

$$
\begin{aligned}
y_n =& y_{n-1} + 0.5h\left[(x_{n-1} - y_{n-1} + 1) + (-y_{n-1} - h\,(x_{n-1} - y_{n-1} + 1) + x_n + 1\right] \\
=& 0.905 y_{n-1} + 0.045 x_{n-1} + 0.05 x_k + 0.095
\end{aligned}
$$

所以

$$
\begin{aligned}
y_1 &= 0.905 y_0 + 0.045 x_0 + 0.05 x_1 + 0.095 \\
&= 1.005000
\end{aligned}
$$

Simpson公式

$$\int_{x_{n-1}}^{x_n} f(x, y(x))\mathrm{d}x$$
$$\approx \frac{h}{6}\left[f\left(x_{n-1}, y_{n-1}\right) + 4f\left(\frac{x_{n-1}+x_k}{2}, y\left(\frac{x_{n-1}+x_n}{2}\right)\right) + f\left(x_n, y\left(x_n\right)\right)\right]$$
$$= \frac{h}{6}\left[f\left(x_{n-1}, y_{n-1}\right)+4f\left(x_{n-1}+\frac{h}{2}, y\left(x_{n-1}+\frac{h}{2}\right)\right) + f(x_{n-1}+h, y\left(x_{n-1}+h\right))\right]$$

所以

$$y_n = y_{n-1} + \frac{h}{6}\Big[f(x_{n-1}, y_{n-1}) + 4f(x_{n-1}+\frac{h}{2}, y(x_{n-1}+\frac{h}{2})) \\ + f(x_{n-1}+h, y(x_{n-1}+h))\Big] \tag{5.4}$$

上式是隐式, 不便计算, 因此需要显化。

Simpson公式

将每个小区间 $[x_{n-1}, x_n]$ 再进行对分，令对分后的小区间长度为 $h_1 = \frac{h}{2}$，于是有

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

则 (5.4)化简为

$$\begin{aligned} y_{n+1} &= y_{n-1} + \frac{h_1}{3}[f(x_{n-1}, y_{n-1}) + 4f(x_n, y_n) + f(x_{n+1}, y_{n+1})] \\ \Rightarrow y_n &= y_{n-2} + \frac{h_1}{3}[f(x_{n-2}, y_{n-2}) + 4f(x_{n-1}, y_{n-1}) + f(x_n, y_n)] \end{aligned}$$

上式称为 Simpson 求解公式。

Runge-Kutta法

基本思想： 用不同点的函数值作线性组合，构造近似公式，把近似公式和解的 Taylor 展开比较，使前面的若干项吻合，从而使近似公式达到一定的阶数。

一般的显式 Runge-Kutta 方法，可以写成

$$\begin{cases} y_1 = y_{n-1} + c_1 k_1 + c_2 k_2 + \cdots + + c_N k_N \\ k_1 = hf(x_{n-1}, y_{n-1}) \\ k_2 = hf(x_{n-1} + a_2 h, y_{n-1} + b_{21} k_1) \\ k_3 = hf(x_{n-1} + a_3 h, y_{n-1} + b_{31} k_1 + b_{32} k_2) \\ \vdots \\ k_N = hf(x_{n-1} + a_N h, y_{n-1} + b_{N1} k_1 + b_{N2} k_2 + b_{N,N-1} k_N) \end{cases}$$

其中，a_i, b_i, c_i为常数，选取这些常数的原则是，要求第一式的右端在 (x_n, y_n) 处 Taylor 展开后，按 h 的幂次重新整理得到

$$y_n = y_{n-1} + \gamma_1 h + \frac{1}{2!}\gamma_2 h^2 + \frac{1}{3!}\gamma_3 h^3 + \cdots$$

与微分方程的解的 Taylor 展开式

$$y(x_n) = y(x_{n-1}) + f_{n-1}h + \frac{1}{2!}f'_{n-1}h^2 + \frac{1}{3!}f''_{n-1}h^3 + \cdots$$

有尽可能多的项重合，即要求

$$\gamma_1 = f_{n-1}, \gamma_2 = f'_{n-1}, \gamma_3 = f''_{n-1}, \cdots$$

其中， f_{n-1}，f'_{n-1}，f''_{n-1} 分别表示

$$y'(x_{n-1}) = f(x_{n-1}, y_{n-1}), y''(x_{n-1}), y'''(x_{n-1}), \cdots$$

上述公式叫做 N 阶的 Runge-Kutta 方法，其局部截断误差为 $\mathcal{O}(h^{N+1})$。

显然，Euler 法是一阶 R-K 方法。

Runge-Kutta法

下面以二阶 R-K 公式为例，来说明 R-K 方法的推导过程。

复习 $f(x,y)$ 在 (a,b) 的 Taylor 展开：

$$
\begin{aligned}
f(x,y) \approx & f(a,b) + f_x(a,b)(x-a) + f_y(a,b)(y-b) \\
& + \frac{f_{xx}(a,b)}{2}(x-a)^2 + f_{xy}(a,b)(x-a)(y-b) + \frac{f_{yy}(a,b)}{2}(y-b)^2
\end{aligned}
$$

要求下式中

$$
\begin{cases}
y_n = y_{n-1} + c_1 k_1 + c_2 k_2 \\
k_1 = hf(x_{n-1}, y_{n-1}) \\
k_2 = hf(x_{n-1} + a_2 h, y_{n-1} + b_{21} k_1)
\end{cases}
$$

系数 c_1, c_2, a_2, b_2，使得 $y_n = y(x_n)$ 时，局部截断误差为 $\mathcal{O}(h^3)$

Runge-Kutta法

将 k_2 在 (x_{n-1}, y_{n-1}) 处展开，得

$$
\begin{aligned}
k_2 =& hf(x_{n-1} + a_2 h, y_{n-1} + b_{21} k_1) \\
=& h\left[f(x_{n-1}, y_{n-1}) + a_2 h f_x(x_{n-1}, y_{n-1}) + b_{21} k_1 f_y(x_{n-1}, y_{n-1})\right. \\
& + \frac{1}{2}[(a_2 h)^2 f_{xx}(x_{n-1}, y_{n-1}) + 2a_2 b_{21} h k_1 f_{xy}(x_{n-1}, y_{n-1}) \\
& \left. + (b_{21} k_1)^2 f_{yy}(x_{n-1}, y_{n-1})]\right] \\
=& h\left[y'(x_{n-1}) + a_2 h f_x(x_{n-1}, y_{n-1}) + b_{21} h y'(x_{n-1}) f_y(x_{n-1}, y_{n-1})\right. \\
& + \frac{1}{2}[a_2^2 h^2 f_{xx}(x_{n-1}, y_{n-1}) + 2a_2 b_{21} h^2 y'(x_{n-1}) f_{xy}(x_{n-1}, y_{n-1}) \\
& \left. + b_{21}^2 h^2 y'(x_{n-1})^2 f_{yy}(x_{n-1}, y_{n-1})]\right]
\end{aligned}
$$

Runge-Kutta法

将k_1, k_2 代入 y_n 得

$$
\begin{aligned}
y_n =& y_{n-1} + c_1 k_1 + c_2 k_2 \\
=& y(x_{n-1}) + c_1 h y'(x_{n-1}) + c_2 h y'(x_{n-1}) + a_2 c_2 h^2 f_x(x_{n-1}, y_{n-1}) \\
& + b_{21} c_2 h^2 y'(x_{n-1}) f_y(x_{n-1}, y_{n-1}) + \frac{1}{2} a_2^2 c_2 h^3 f_{xx}(x_{n-1}, y_{n-1}) \\
& + a_2 b_{21} c_2 h^3 y'(x_{n-1}) f_{xy}(x_{n-1}, y_{n-1}) + \frac{1}{2} b_{21}^2 c_2 h^3 (y'(x_{n-1}))^2 f_{yy}(x_{n-1}, y_{n-1}) \\
=& y(x_{n-1}) + (c_1 + c_2) y(x_{n-1})' h + [a_2 c_2 f_x + c_2 b_{21} f f_y]_{(x_{n-1}, y_{n-1})} h^2 \\
& + [\frac{1}{2} a_2^2 c_2 f_{xx} + a_2 b_{21} c_2 f f_{xy} + \frac{1}{2} b_{21}^2 c_2 f_{yy}]_{(x_{n-1}, y_{n-1})} h^3
\end{aligned}
$$

Runge-Kutta法

y_{x_n} 在 x_{n-1} 处的 Taylor 展开为

$$\begin{aligned}y(x_n) =& y(x_{n-1}) + y'(x_{n-1})h + \frac{1}{2}y''(x_{n-1})h^2 + \frac{1}{6}y'''(x_{n-1})h^3 \\ =& y(x_{n-1}) + y'(x_{n-1})h + \frac{1}{2}[f_x + ff_y]_{(x_{n-1},y_{n-1})}h^2 \\ &+ \frac{1}{6}[f_{xx} + 2ff_{xy} + f^2 f_{yy} + f_y(f_x + ff_y)]_{(x_{n-1},y_{n-1})}h^3\end{aligned}$$

所以

$$\begin{aligned}R_n =& y(x_n) - y_n \\ =& (1 - c_1 - c_2)y'(x_{n-1})h \\ &+ \left[(\frac{1}{2} - c_2 a_2)f_x + (\frac{1}{2} - c_2 b_{21})ff_y\right]_{(x_{n-1},y_{n-1})} h^2 + \Bigg[(\frac{1}{6} - \frac{1}{2}a_2^2 c_2)f_{xx} \\ &+ (\frac{1}{3} - a_2 b_{21} c_2)ff_{xy} + (\frac{1}{6} - \frac{1}{2}b_{21}^2 c_2)f_{yy} + \frac{1}{6}f_y(f_x + ff_y)\Bigg]_{(x_{n-1},y_{n-1})} h^3\end{aligned}$$

Runge-Kutta法

要使 R_n 尽量小，应先让 h, h^2 项的系数为零：

$$c_1 + c_2 = 1, c_2 a_2 = \frac{1}{2}, c_2 b_{21} = \frac{1}{2}$$

由于要求二阶方法，$c_2 \neq 0$, 所以

$$c_1 = 1 - c_2, a_2 = \frac{1}{2c_2}, b_{21} = \frac{1}{2c_2}$$

此时 R_n 为

$$R_n = [(\frac{1}{6} - \frac{1}{8c_2})(f_{xx} + 2ff_{xy} + f^2 f_{yy}) + \frac{1}{6} f_y (f_x + ff_y)]_{(x_{n-1}, y_{n-1})} h^3$$

要使 R_n 尽量小，应令 $\frac{1}{6} - \frac{1}{8c_2} = 0$，得

$$c_2 = \frac{3}{4}, c_1 = \frac{1}{4}, a_2 = \frac{2}{3}, b_{21} = \frac{2}{3}$$

局部截断误差为： $R_n = \frac{h^3}{6}[f_y(f_x + ff_y)]_{(x_{n-1}, y_{n-1})}$

Runge-Kutta法

所以二阶 R–K 方法为

$$\begin{cases} y_n = y_{n-1} + \dfrac{1}{4}(k_1 + 3k_2) \\ k_1 = hf(x_{n-1}, y_{n-1}) \\ k_2 = hf(x_{n-1} + \dfrac{2}{3}h, y_{n-1} + \dfrac{2}{3}k_1) \end{cases}$$

该方法又称为二阶 Heun （休恩）方法。

若取 $c_1 = c_2 = \frac{1}{2}$, 则 $a_2 = b_{21} = 1$

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

得 Euler 预测-校正格式

Runge-Kutta法

Remark 1: 我们可以构造无穷多个二阶 R–K 方法，这些方法的截断误差均为 $\mathcal{O}(h^3)$, 即都是二阶方法。其中二阶 Heun 方法是截断误差项数最少，且允许 f 任意变化的情况下截断误差最小的二阶方法。

Remark 2: 二阶 R–K 方法不可能达到三阶。

Remark 3: 同样可构造其他阶的 R–K 方法它们都有无穷多组解且三阶 R–K 方法阶数不超过3, 四阶 R–K 方法阶数不超过4。

Remark 4: 更高阶的方法由于计算量较大一般不再采用。

Runge-Kutta法

三阶 R–K 法。

$$\begin{cases} y_n = y_{n-1} + \frac{1}{6}(k_1 + 4k_2 + k_3) \\ k_1 = hf\left(x_{n-1}, y_{n-1}\right) \\ k_2 = hf\left(x_{n-1} + \frac{1}{2}h, y_{n-1} + \frac{1}{2}k_1\right) \\ k_3 = hf\left(x_{n-1} + h, y_{n-1} + (2k_2 - k_1)\right) \end{cases}$$

Runge-Kutta法

标准（经典）四阶 R–K 公式

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

Runge-Kutta法

关于R–K方法计算量的讨论

二阶R–K方法需计算两个函数值，四阶R–K方法需计算四个函数值，但精度要比二阶方法高出两阶。因此，要达到同样的精度，用低阶方法需步长取得比较小，但若用高阶方法则可以将步长取得大一些，从而降低计算量。

例题

例: 使用三阶、四阶R-K法计算初值问题:

$$\begin{cases} y' = y^2, x \in [0, 0.5] \\ y(0) = 1 \end{cases}$$

的部分数值解 y_1, y_2, y_3, 其中 $h = 0.1$

例题

例: 使用三阶、四阶R-K法计算初值问题:

$$\begin{cases} y' = y^2, x \in [0, 0.5] \\ y(0) = 1 \end{cases}$$

的部分数值解 y_1, y_2, y_3, 其中 $h = 0.1$

解: (1) 三阶R-K法

$$\begin{cases} y_n = y_{n-1} + \frac{h}{6}\left(k_1 + 4k_2 + k_3\right) \\ k_1 = f\left(x_{n-1}, y_{n-1}\right) = y_{n-1}^2 \\ k_2 = f\left(x_{n-1} + 0.5h, y_{n-1} + 0.5hk_1\right) = \left(y_{n-1} + 0.5hy_{n-1}^2\right)^2 \\ k_3 = f\left(x_{n-1} + h, y_{n-1} + h\left(2k_2 - k_1\right)\right) = \left(y_{n-1} + h\left(2k_2 - k_1\right)\right)^2 \end{cases}$$

例题

当 $n=1$

$$\begin{cases} k_1 = f\left(x_0, y_0\right) = y_0^2 = 1 \\ k_2 = f\left(x_0 + 0.5h, y_0 + 0.5hk_1\right) = \left(y_0 + 0.5hy_0^2\right)^2 = 1.102500 \\ k_3 = f\left(x_0 + h, y_0 + h\left(2k_2 - k_1\right)\right) = \left(y_{n-1} + h\left(2k_2 - k_1\right)\right)^2 \approx 1.255520 \\ y_1 = y_0 + \frac{h}{6}\left(K_1 + 4k_2 + k_3\right) \approx 1.111092 \end{cases}$$

当 $n=2$

$$\begin{cases} k_1 = f\left(x_1, y_1\right) = y_1^2 = 1.111092^2 \approx 1.234525 \\ k_2 = f\left(x_1 + 0.5h, y_1 + 0.5hk_1\right) = \left(y_1 + 0.5hy_1^2\right)^2 = 1.375502 \\ k_3 = f\left(x_1 + h, y_1 + h\left(2k_2 - k_1\right)\right) = \left(y_1 + h\left(2k_2 - k_1\right)\right)^2 \approx 1.594512 \\ y_2 = y_1 + \frac{h}{6}\left(K_1 + 4k_2 + k_3\right) \approx 1.249943 \end{cases}$$

当 $n=3$：略

例题

(2) 四阶R-K法

$$\begin{cases} K_1 = f\left(x_{n-1}, y_{n-1}\right) = y_{n-1}^2 \\ K_2 = f\left(x_{n-1} + 0.5h, y_{n-1} + 0.5hK_1\right) = \left(y_{n-1} + 0.5hK_1\right)^2 \\ K_3 = f\left(x_{n-1} + 0.5h, y_{n-1} + 0.5hK_2\right) = \left(y_{n-1} + 0.5hK_2\right)^2 \\ K_4 = f\left(x_{n-1} + h, y_{n-1} + hK_3\right) = \left(y_{n-1} + hK_3\right)^2 \\ y_n = y_{n-1} + \frac{h}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right) \end{cases}$$

当 $n = 1$

$$\begin{cases} k_1 = f\left(x_0, y_0\right) = y_0^2 = 1.000000 \\ k_2 = f\left(x_0 + 0.5h, y_0 + 0.5hk_1\right) = \left(y_0 + 0.5hk_1\right)^2 = 1.102500 \\ k_3 = f\left(x_0 + 0.5h, y_0 + 0.5hk_2\right) = \left(y_0 + 0.5hk_2\right)^2 \approx 1.113289 \\ k_4 = f\left(x_0 + h, y_0 + hk_3\right) = \left(y_0 + hk_3\right)^2 \approx 1.235052 \\ y_1 = y_0 + \frac{h}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right) \approx 1.111111 \end{cases}$$

例题

当 $n=2$

$$\begin{cases} k_1 = f\left(x_1, y_1\right) = y_1^2 = 1.234568 \\ k_2 = f\left(x_1 + 0.5h, y_1 + 0.5hk_1\right) = \left(y_1 + 0.5hk_1\right)^2 = 1.375551 \\ k_3 = f\left(x_1 + 0.5h, y_1 + 0.5hk_2\right) = \left(y_1 + 0.5hk_2\right)^2 \approx 1.392138 \\ k_4 = f\left(x_1 + h, y_1 + hk_3\right) = \left(y_1 + hk_3\right)^2 \approx 1.563313 \\ y_2 = y_1 + \frac{h}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right) \approx 1.249999 \end{cases}$$

当 $n=3$

$$\begin{cases} k_1 = f\left(x_2, y_2\right) = y_2^2 = 1.562498 \\ k_2 = f\left(x_2 + 0.5h, y_2 + 0.5hk_1\right) = \left(y_2 + 0.5hk_1\right)^2 = 1.763913 \\ k_3 = f\left(x_2 + 0.5h, y_2 + 0.5hk_2\right) = \left(y_2 + 0.5hk_2\right)^2 \approx 1.790766 \\ k_4 = f\left(x_2 + h, y_2 + hk_3\right) = \left(y_2 + hk_3\right)^2 \approx 2.042258 \\ y_3 = y_2 + \frac{h}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right) \approx 1.428568 \end{cases}$$