EXAMPLE OF ONE-STEP METHOD

Consider solving

$$y' = y\ \cos x, \quad y(0) = 1$$

Imagine writing a Taylor series for the solution $Y(x)$, say initially about $x = 0$. Then

$$Y(h) = Y(0) + hY'(0) + \frac{h^2}{2}Y''(0) + \frac{h^3}{6}Y'''(0) + \cdots$$

We can calculate $Y'(0) = Y(0)\cos(0) = 1$. How do we calculate $Y''(0)$ and higher order derivatives?

$$Y'(x) = Y(x)\cos(x)$$

$$Y''(x) = -Y(x)\sin(x) + Y'(x)\cos(x)$$

$$Y'''(x) = -Y(x)\cos(x) - 2Y'(x)\sin(x) + Y''(x)\cos x$$

Then $Y(0) = 1$, $Y'(0) = 1$,and

$$Y''(0) = -Y(0)\sin(0) + Y'(0)\cos(0) = 1$$

$$Y'''(0) = -Y(0)\cos(0) - 2Y'(0)\sin(0) + Y''(0)\cos 0 = 0$$

Thus

$$\begin{aligned} Y(h) &= Y(0) + hY'(0) + \tfrac{h^2}{2}Y''(0) \\ &\qquad + \tfrac{h^3}{6}Y'''(0) + \cdots \\ &= 1 + h + \tfrac{h^2}{2} + \cdots \end{aligned}$$

We can generate as many terms as desired, obtaining added accuracy as we do so. In this particular case, the true solution is $Y(x) = \exp(\sin x)$. Thus

$$Y(h) = 1 + h + \frac{h^2}{2} - \frac{h^4}{8} + \cdots$$

We can truncate the series after a particular order. Then continue with the same process to generate approximations to $Y(2h), Y(3h), \ldots$ Letting $x_n = nh$, and using the order 2 Taylor approximation, we have

$$Y(x_{n+1}) = Y(x_n) + hY'(x_n) + \frac{h^2}{2}Y''(x_n) + \frac{h^3}{6}Y'''(\xi_n)$$

with $x_n \leq \xi_n \leq x_{n+1}$. Drop the truncation error term, and then define

$$y_{n+1} = y_n + hy_n' + \frac{h^2}{2}y_n'', \quad n \geq 0$$

with

$$y_n' = y_n \cos(x_n)$$

$$y_n'' = -y_n \sin(x_n) + y_n' \cos(x_n)$$

We give a numerical example of computing the numerical solution with Taylor series methods of orders 2, 3, and 4. For a Taylor series of degree r, the global error will be $O\left(h^r\right)$. The numerical example output is given in a separate file.

A 4^{th}-ORDER EXAMPLE

Consider solving

$$y' = -y, \quad y(0) = 1$$

whose true solution is $Y(x) = e^{-x}$. Differentiating the equation

$$Y'(x) = -Y(x)$$

we obtain

$$Y'' = -Y' = Y$$

$$Y''' = Y' = -Y, \quad Y^{(4)} = Y$$

Then expanding $Y(x_n + h)$ in a Taylor series,

$$\begin{aligned} Y(x_{n+1}) = Y_n + hY_n' + \frac{h^2}{2}Y_n'' + \frac{h^3}{6}Y_n''' \\ + \frac{h^4}{24}Y_n^{(4)} + \frac{h^5}{120}Y^{(4)}(\xi_n) \end{aligned}$$

Dropping the truncation error, we have the numerical method

$$\begin{aligned} y_{n+1} &= y_n + hy_n' + \tfrac{h^2}{2}y_n'' + \tfrac{h^3}{6}y_n''' + \tfrac{h^4}{24}y_n^{(4)} \\ &= \left(1 - h + \tfrac{h^2}{2} - \tfrac{h^3}{6} + \tfrac{h^4}{24}\right) y_n \end{aligned}$$

with $y_0 = 1$. By induction,

$$y_n = \left(1 - h + \frac{h^2}{2} - \frac{h^3}{6} + \frac{h^4}{24}\right)^n, \quad n \geq 0$$

Recall that

$$e^{-h} = 1 - h + \frac{h^2}{2} - \frac{h^3}{6} + \frac{h^4}{24} - \frac{h^5}{120}e^{-\xi}$$

with $0 < \xi < h$. Then

$$\begin{aligned} y_n &= \left(e^{-h} + \tfrac{h^5}{120}e^{-\xi}\right)^n \\ &= e^{-nh}\left(1 + \tfrac{h^5}{120}e^{h-\xi}\right)^n \\ &\doteq e^{-x_n}\left(1 + \tfrac{x_n h^4}{120}e^{h-\xi}\right) \end{aligned}$$

Thus

$$Y(x_n) - y_n = e^{-x_n} + O(h^4)$$

We can introduce the Taylor series method for the general problem

$$y' = f(x, y), \quad y(x_0) = Y_0$$

Simply imitiate what was done above for the particular problem $y' = y \cos x$.

In general,

$$\begin{aligned} Y'(x) &= f\left(x, Y(x)\right) \\ Y''(x) &= f_x\left(x, Y(x)\right) + f_y\left(x, Y(x)\right) Y'(x) \\ &= f_x\left(x, Y(x)\right) + f_y\left(x, Y(x)\right) f\left(x, Y(x)\right) \\ Y'''(x) &= f_{xx} + 2 f_{xy} f + f_{yy} f^2 + f_y f_x + f_y^2 f \end{aligned}$$

and we can continue on this manner. Thus we can calculate derivatives of any order for $Y(x)$; and then we can define Taylor series of any desired order.

This used to be considered much too arduous a task for practical problems, because everything had to be done by hand. But with symbolic programs such as *Mathematica* and *Maple*, Taylor series can be considered a serious framework for numerical methods. Programs that implement this in an automatic way, with varying order and stepsize, are available.

RUNGE-KUTTA METHODS

Nonetheless, most researchers still consider Taylor series methods to be too expensive for most practical problems (a point contested by others). This leads us to look for other one-step methods which imitate the Taylor series methods, without the necessity to calculate the higher order derivatives. These are called *Runge-Kutta methods*. There are a number of ways in which one can approach Runge-Kutta methods, and I will describe a fairly classical approach.

We begin by considering *explicit Runge-Kutta methods of order 2*. We want to write

$$y_{n+1} = y_n + hF(x_n, y_n, h; f)$$

with $F(x_n, y_n, h; f)$ some carefully chosen approximation to $f(x, y)$ on the interval $[x_n, x_{n+1}]$. In particular, write

$$F(x, y, h; f) = \gamma_1 f(x, y) + \gamma_2 f(x + \alpha h, y + \beta h f(x, y))$$

$F(x, y, h; f) = \gamma_1 f(x, y) + \gamma_2 f(x + \alpha h, y + \beta h f(x, y))$

This is some kind of “average” derivative. Intuitively, we should restrict α so that $0 \le \alpha \le 1$. In addition to this, how should the constants $\gamma_1, \gamma_2, \alpha, \beta$ be chosen?

Introduce the <u>truncation error</u>

$$T_n(Y) = Y(x + h) - [Y(x) + hF(x, Y(x), h; f)]$$

and choose the constants to make $T_n(Y) = O(h^p)$ with p as large as possible.

By choosing

$$\alpha = \beta = \frac{1}{2\gamma_2}, \quad \gamma_1 = 1 - \gamma_2$$

we can show

$$T_n(Y) = O(h^3)$$

EXAMPLES

$$
\begin{aligned}
y_{n+1} &= y_n + hF(x_n, y_n, h; f) \\
F(x, y, h; f) &= \gamma_1 f(x, y) + \gamma_2 f(x + \alpha h, y + \beta h f(x, y))
\end{aligned}
$$

Case: $\gamma_2 = \frac{1}{2}$. This leads to the *trapezoidal* Runge-Kutta method.

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

Case: $\gamma_2 = 1$. This leads to the *midpoint* Runge-Kutta method.

$$y_{n+1} = y_n + hf(x_n + \tfrac{h}{2}, y_n + \tfrac{h}{2} f(x_n, y_n)) \tag{2}$$

We can derive other second order formulas by choosing other values for γ_2.

We illustrate these two methods by solving

$$Y'(x) = -Y(x) + 2\cos x, \qquad Y(0) = 1$$

with true solution $Y(x) = \sin x + \cos x$. Observe the rate at which the error decreases when h is halved.

NUMERICAL EXAMPLE. Solve

$$Y'(x) = -Y(x) + 2\cos x, \qquad Y(0) = 1$$

The true solution is $Y(x) = \sin x + \cos x$. We give numerical results for the Runge-Kutta method in (1) with stepsizes $h = 0.05$ and 0.1.

h	x	$y_h(x)$	*Error*
0.1	2.0	0.491215673	1.93E − 3
	4.0	−1.407898629	−2.55E − 3
	6.0	0.680696723	5.81E − 5
	8.0	0.841376339	2.48E − 3
	10.0	−1.380966579	−2.13E − 3
0.05	2.0	0.492682499	4.68E − 4
	4.0	−1.409821234	−6.25E − 4
	6.0	0.680734664	2.01E − 5
	8.0	0.843254396	6.04E − 4
	10.0	−1.382569379	−5.23E − 4

Observe the ratio by which the error decreases when h is halved, for each fixed x. This is consistent with an error formula

$$Y(x_n) - y_h(x_n) = O\left(h^2\right)$$

FOUR-STAGE FORMULAS

We have just studied 2-stage formulas. To obtain a higher rate of convergence, we must use more derivative evaluations. A 4-stage formula looks like

$$y_{n+1} = y_n + hF(x_n, y_n, h; f)$$
$$F(x, y, h; f) = \gamma_1 V_1 + \gamma_2 V_2 + \gamma_3 V_3 + \gamma_4 V_4$$

$$\begin{aligned}
V_1 &= f(x, y) \\
V_2 &= f(x + \alpha_2 h, y + \beta_{2,1} h V_1) \\
V_3 &= f(x + \alpha_3 h, y + \beta_{3,1} h V_1 + \beta_{3,2} h V_2) \\
V_4 &= f(x + \alpha_4 h, y + h(\beta_{4,1} V_1 + \beta_{4,2} V_2 + \beta_{4,3} V_3)
\end{aligned}$$

Again it can be analyzed by expanding the truncation error

$$T_n(Y) = Y(x + h) - [Y(x) + hF(x, Y(x), h; f)]$$

in powers of h.

We attempt to choose the unknown coefficients so as to force $T_n(Y) = O(h^p)$ with p as large as possible. In the above case, this can be done so that $p = 5$. The algebra becomes very complicated, and we omit it here.

The truncation error indicates the new error introduced in each step of the numerical method. The total or *global error* for this case will be of size $O\left(h^4\right)$.

The classical 4^{th}-order formula follows.

$$y_{n+1} = y_n + \frac{h}{6}\left[V_1 + 2v_2 + 2V_3 + V_4\right]$$

$$\begin{array}{rcl} V_1 & = & f(x, y) \\ V_2 & = & f(x + \frac{1}{2}h, y + \frac{1}{2}hV_1) \\ V_3 & = & f(x + \frac{1}{2}h, y + \frac{1}{2}hV_2) \\ V_4 & = & f(x + h, y + hV_3) \end{array}$$

The even more accurate 4^{th}-order Runge-Kutta-Fehlberg formula is given in the text, along with a numerical example.

ERROR DISCUSSION

If a Runge-Kutta method satisfies

$$T_n(Y) = O(h^p)$$

with $p \geq 2$, then it can be shown that

$$|Y(x_n) - y_n| \leq ch^{p-1}, \qquad x_0 \leq x_n \leq b$$

when solving the initial value problem

$$y' = f(x, y), \quad x_0 \leq x_n \leq b, \qquad y(x_0) = Y_0$$

We can also go further and show that

$$Y(x_n) - y_h(x_n) = D(x_n)h^{p-1} + O\left(h^p\right), \quad x_0 \leq x_n \leq b$$

This can then be used to justify Richardson's extrapolation.

For example, if $T_n(Y) = O(h^3)$, then Richardson's error estimate is

$$Y(x_n) - y_h(x_n) \approx \frac{1}{3}\left[y_h(x_n) - y_{2h}(x_n)\right]$$