Initial Value Problems of ODEs

Differential Equations

Objective: Solve an equation containing an unknown function and one or more of its derivatives

Problem: Find $y(t)$ satisfying:

$$\begin{aligned} y'(t) &= f(t, y(t)), \quad t \geq t_0 & (1) \\ y(t_0) &= y_0 & (2) \end{aligned}$$

- (1)–(2) is called an initial value problem of the ODE $y'(t) = f(t, y)$.
- $y(t)$ is a scalar valued function of t.

Examples

$$\boxed{\begin{aligned} y'(t) &= \sin(t) \\ y(\frac{\pi}{3}) &= 2 \end{aligned}}$$

- $y'(t) = sin(t) \Longrightarrow y(t) = -\cos(t) + C$ (general solution)

Examples

$$y'(t) = \sin(t)$$
$$y(\frac{\pi}{3}) = 2$$

- $y'(t) = sin(t) \implies y(t) = -\cos(t) + C$ (general solution)
- Using initial condition $y(\frac{\pi}{3}) = 2$, we have a particular solution by solving for C

$$y(t) = 2.5 - \cos(t)$$

Examples

$$y'(t) = \lambda y(t) + b(t), \quad t \geq t_0 \text{ and } \lambda \text{ constant}$$

Examples

$$\boxed{y'(t) = \lambda y(t) + b(t), \quad t \geq t_0 \text{ and } \lambda \text{ constant}}$$

Solve using the method of **integrating factors!**

$$\begin{aligned} y'(t) - \lambda y(t) &= b(t) \\ (y'(t) - \lambda y(t))e^{-\lambda t} &= e^{-\lambda t} b(t) \end{aligned}$$

Recognize that $\boxed{(y'(t) - \lambda y(t))e^{\lambda t} = \frac{d}{dt}\left(y(t)e^{-\lambda t}\right)}$ then

$$\frac{d}{dt}\left(y(t)e^{-\lambda t}\right) = e^{-\lambda t} b(t)$$

Examples

$$\frac{d}{dt}\left(y(t)e^{-\lambda t}\right) = e^{-\lambda t}b(t)$$

Integrating both sides:

$$e^{-\lambda t}y(t) = \int_{t_0}^{t} e^{-\lambda s}b(s)\,ds + C$$

so the general solution is

$$\boxed{y(t) = e^{\lambda t}\left[C + \int_{t_0}^{t} e^{-\lambda s}b(s)\,ds\right] = Ce^{\lambda t} + \int_{t_0}^{t} e^{\lambda(t-s)}b(s)\,ds}$$

Second order IVP (Hooke's Law - spring-mass oscillation)

If the displacement is not too large, the force exerted on the mass is proportional to the displacement from the origin

$$\begin{aligned} y''(t) &= -ky, \quad k > 0 \\ y(0) &= y_0 \\ y'(0) &= 0 \end{aligned}$$

- $y(t) = c_1 \sin(\sqrt{k}t) + c_2 \cos(\sqrt{k}t)$

Euler's method

$$\boxed{\begin{aligned} y'(t) = f(t, y(t)), \\ y(a) = y_0 \quad a \le t \le b \end{aligned}}$$

- **Step 1:** Divide $[a, b]$ into N subintervals of size $h = \dfrac{b-a}{N}$

$$\boxed{a = t_0 < t_1 < \cdots < t_{N-1} < t_N = b}$$

Euler's method

$$\boxed{\begin{array}{c} y'(t) = f(t, y(t)), \\ y(a) = y_0 \quad a \le t \le b \end{array}}$$

- **Step 1:** Divide $[a, b]$ into N subintervals of size $h = \dfrac{b-a}{N}$

$$\boxed{a = t_0 < t_1 < \cdots < t_{N-1} < t_N = b}$$

- **Step 2:** Replace $y'(t)$ by an approximation (from calc I)

$$\boxed{y'(t) = \lim_{h \to 0} \frac{y(t+h) - y(t)}{h}}$$

so starting with $[t_0, t_1]$,

$$y'(t_0) \approx \frac{y(t_1) - y(t_0)}{h} = f(t_0, y_0)$$

Euler's method for $y'(t) = f(t, y(t)) \quad a \leq t \leq b$

t grid: $\boxed{a = t_0 < t_1 < \cdots < t_{N-1} < t_N = b}$

- On subinterval $[t_0, t_1]$: replace $y'(t_0)$ by a finite difference:

$$\boxed{y'(t_0) \approx \frac{y(t_1) - y(t_0)}{h} = f(t_0, y_0)} \Longrightarrow \boxed{y_1 = y_0 + hf(t_0, y_0)}$$

Euler's method for $y'(t) = f(t, y(t)) \quad a \leq t \leq b$

t grid: $a = t_0 < t_1 < \cdots < t_{N-1} < t_N = b$

- On subinterval $[t_0, t_1]$: replace $y'(t_0)$ by a finite difference:

$$y'(t_0) \approx \frac{y(t_1) - y(t_0)}{h} = f(t_0, y_0) \implies y_1 = y_0 + hf(t_0, y_0)$$

- On subinterval $[t_1, t_2]$: replace $y'(t_1)$ by a finite difference:

$$y'(t_1) \approx \frac{y(t_2) - y(t_1)}{h} = f(t_1, y_1) \implies y_2 = y_1 + hf(t_1, y_1)$$

Euler's method for $y'(t) = f(t, y(t)) \quad a \le t \le b$

t grid: $a = t_0 < t_1 < \cdots < t_{N-1} < t_N = b$

- On subinterval $[t_0, t_1]$: replace $y'(t_0)$ by a finite difference:

$$y'(t_0) \approx \frac{y(t_1) - y(t_0)}{h} = f(t_0, y_0) \Longrightarrow y_1 = y_0 + hf(t_0, y_0)$$

- On subinterval $[t_1, t_2]$: replace $y'(t_1)$ by a finite difference:

$$y'(t_1) \approx \frac{y(t_2) - y(t_1)}{h} = f(t_1, y_1) \Longrightarrow y_2 = y_1 + hf(t_1, y_1)$$

In general on subinterval $[t_k, t_{k+1}]$: replace $y'(t_k)$ by a finite:

$$y'(t_k) \approx \frac{y(t_{k+1}) - y(t_k)}{h} = f(t_k, y_k) \Longrightarrow y_{k+1} = y_k + hf(t_k, y_k)$$

Euler's method - quick derivation

Brook Taylor (1685–1731)[1]

Theorem (Taylor)

Let $f, f', \cdots f^{(n)}$ be continuous on $[a, b]$ and let $f^{(n+1)}$ exist for all $t\ in (a, b)$. Then there is a number ξ between t and a such that

$$f(t) = f(a) + (t-a)f'(a) + \frac{(t-a)^2}{2!}f''(a) + \cdots + \frac{(x-a)^n}{n!}f^{(n)}(a) + \frac{(x-a)^{n+1}}{(n+1)!}f^{(n+1)}(\xi)$$

[1] http://www-history.mcs.st-andrews.ac.uk/PictDisplay/Taylor.html

Euler's method - quick derivation

Leonard Euler (1707–1783)[2]

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

[2] http://www-history.mcs.st-andrews.ac.uk/PictDisplay/Euler.html

Euler's method - quick derivation

Taylor's Theorem

Given the (IVP) $y'(t) = f(t, y(t))$, expand $y(t)$ about t_k as

$$y(t) \;\; = \;\; y(t_k) + (t - t_k)y'(t_k) + \frac{(t - t_k)^2}{2} y''(\xi_k), \quad \xi_k \in [t_k, t_{k+1}]$$

Euler's method - quick derivation

Taylor's Theorem

Given the (IVP) $y'(t) = f(t, y(t))$, expand $y(t)$ about t_k as

$$
\begin{aligned}
y(t) &= y(t_k) + (t - t_k)y'(t_k) + \frac{(t - t_k)^2}{2} y''(\xi_k), \quad \xi_k \in [t_k, t_{k+1}] \\
&= y(t_k) + (t - t_k) f(t_k, y(t_k)) + \frac{(t - t_k)^2}{2} y''(\xi_k), \quad \xi_k \in [t_k, t_{k+1}]
\end{aligned}
$$

Euler's method - quick derivation

Taylor's Theorem

Given the (IVP) $y'(t) = f(t, y(t))$, expand $y(t)$ about t_k as

$$\begin{aligned} y(t) &= y(t_k) + (t - t_k)y'(t_k) + \frac{(t-t_k)^2}{2} y''(\xi_k), \quad \xi_k \in [t_k, t_{k+1}] \\ &= y(t_k) + (t - t_k)f(t_k, y(t_k)) + \frac{(t-t_k)^2}{2} y''(\xi_k), \quad \xi_k \in [t_k, t_{k+1}] \end{aligned}$$

Evaluating at $t = t_{k+1}$ and recalling that $h = t_{k+1} - t_k$ yields:

$$y(t_{k+1}) = y(t_k) + hf(t_k, y(t_k)) + \frac{h^2}{2} y''(\xi_k)$$

Dropping the error term $\frac{h^2}{2} y''(\xi_k)$

$$\boxed{y_{k+1} = y_k + hf(t_k, y_k), \quad k = 0, 1, 2, \cdots}$$

Euler's method - works well in some cases

Leonard Euler (1707–1783)[3]

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

Euler's method is simple and works for some problems BUT is prone to errors and can be unstable for *stiff problems* (MA 428) [Hidden Figures]

[3] http://www-history.mcs.st-andrews.ac.uk/PictDisplay/Euler.html

Euler's method - modeling the spread of an infection

Modeling the spread of an epidemic (Kermack & McKendrick, *Proc Roy, Soc. (1927)*)

Assumptions

- Population divided into **healthy individuals** (H),**infected individuals** (I) and the **dead** (D)

Euler's method - modeling the spread of an infection

Modeling the spread of an epidemic (Kermack & McKendrick, *Proc Roy, Soc. (1927)*)

Assumptions

- Population divided into **healthy individuals** (H),**infected individuals** (I) and the **dead** (D)
- The epidemic spreads so quickly that changes in population due to birth, death or migration can be ignored.

Euler's method - modeling the spread of an infection

Modeling the spread of an epidemic (Kermack & McKendrick, *Proc Roy, Soc. (1927)*)

Assumptions

- Population divided into **healthy individuals** (H),**infected individuals** (I) and the **dead** (D)
- The epidemic spreads so quickly that changes in population due to birth, death or migration can be ignored.
- The disease is transmitted to healthy individuals at a rate proportional to the product of **healthy** and **infected** people

Euler's method for systems - modeling the spread of an infection

- Turning our assumptions into equations:

$$\frac{dH}{dt} = -cHI, \quad \frac{dI}{dt} = cHI - mI, \quad \frac{dD}{dt} = mI$$

where c is the **transmission rate** and m is the **mortality** rate of infected individuals.

- The model can be reduced to a single equation. First divide the H equation by the D equation to get

$$\frac{dH}{dD} = -\frac{c}{m}H$$

whose solution is:

Euler's method for systems - modeling the spread of an infection

- Turning our assumptions into equations:

$$\frac{dH}{dt} = -cHI, \quad \frac{dI}{dt} = cHI - mI, \quad \frac{dD}{dt} = mI$$

where c is the **transmission rate** and m is the **mortality** rate of infected individuals.

- The model can be reduced to a single equation. First divide the H equation by the D equation to get

$$\frac{dH}{dD} = -\frac{c}{m}H$$

whose solution is:

$$\boxed{H = H_0 e^{-\frac{c}{m}D}}$$

where H_0 is the number of healthy individuals.

Euler's method - modeling the spread of an infection

- If N is the size of the population then

$$H + I + D = N \Longrightarrow I = N - H - D$$

- $$\frac{dD}{dt} = mI \Longrightarrow \frac{dD}{dt} = m(N{-}H{-}D) \Longrightarrow \boxed{\frac{dD}{dt} = m[N - H_0 e^{-\frac{c}{m}D} - D]}$$

- Once D is determined, we can solve for H and I using

$$\boxed{H = H_0 e^{-\frac{c}{m}D}} \quad \boxed{I = N - H - D}$$

Exercise

Write a function
`[t_vals, y_vals] = approx_ode(f,a,b,inital_value,n)` that takes as input a function defined in a file `yprime = f(t,y)`, defined in `f.m` and two values a and b indicating the left and right intervals on which the initial value problem is defined and `n`, the number of Euler steps. Your function should return 2 vectors (`t_vals` contains the values $t_0, t_1, \cdots, t_N$ and `y_vals` is the solution at those points $y_0, y_1, \cdots, y_N$) Test your code on the initial value problem

$$\frac{dy}{dt} = 1 + \frac{y}{t} \quad 1 \leq t \leq 6, \quad y(1) = 1$$

whose exact solution is $y(t) = t(1 + \ln t)$.