So far

- Basics of Python

- Numpy, Scipy & Matplotlib

4 projects

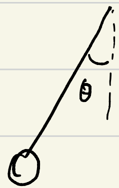
1. Classical Mechanics

2. Quantum

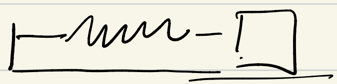
3. Stat Mech

4. Your choice

Simple Harmonic Oscillator:-



$\theta(t)$, $\ddot{\theta}^{2}(t) = -\omega^2 \theta(t)$



$m \ddot{x}(t) = -kx(t)$

2^{nd} order differential eqns.

$$p = m\dot{x} \qquad \longrightarrow \qquad \dot{x} = \frac{p}{m}$$

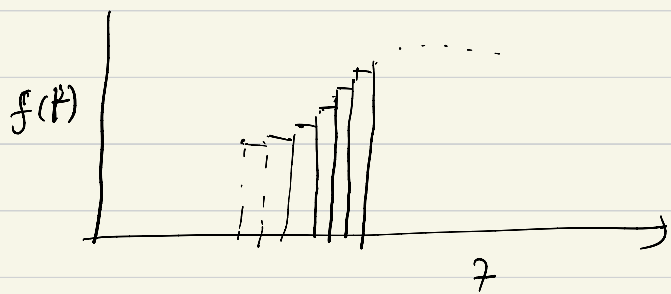
$$\dot{p} = -kx \qquad \qquad \dot{p} = -kx$$

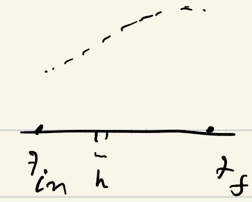
$$y = \begin{pmatrix} x \\ p \end{pmatrix} , \quad \dot{y} = f(y, t)$$

$$= \begin{pmatrix} 0 & \frac{1}{m} \\ -k & 0 \end{pmatrix} \begin{pmatrix} x \\ p \end{pmatrix}$$

$$y(t) \quad , \quad \frac{dy}{dt} = f(t)$$

$$y(t) = \int_{t_0}^{t} dt' \, f(t') + y(t_0)$$





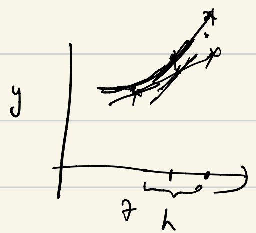
$$\dot{y} = f(y, t)$$

$$t_{n+1} = t_n + h$$

$$y_{n+1} = y_n + h f(y_n, t_n).$$

"Euler method" or "RK1"

RK2



$$k_1 = h f(y_n, t_n)$$

$$k_2 = h f\left(y_n + \tfrac{1}{2} k_1, t_1 + \tfrac{h}{2}\right)$$

$$y_n = y_n + k_2$$

Taylor series:- $\underline{\underline{y(t)}}$, $y(t_0)$, $y'(t_0)$, $y''(t_0)$

$$y(t_0) = y(t_0) + (t - t_0)\, y'(t_0) + \frac{(t - t_0)^2}{2!} y''(t_0)$$

$$+ \dots$$

$$f(t) = \sin(t)$$

A better numerical algorithm:

- Choose a 1d mesh of points:-

$$t_0, t_1 \, . \, . \, . \, t_n, t_{n+1}$$

$$y(t_n) = y_n, \qquad h = t_{n+1} - t_n$$

$$y_{n+1} = y_n + h y_n' + \frac{h^2}{2!} y_n''$$

$$y_n' = f(t_n, y_n) = \frac{dy_n}{dt}$$

$$y_n'' = \left[\frac{\partial f}{\partial t} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial t} \right]_{t_n, y_n}$$

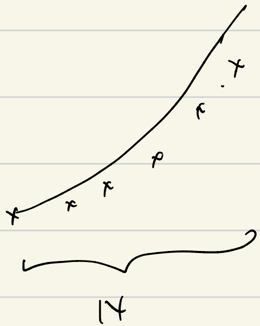
$$y_{n+1} = y_n + h f(t_n, y_n)$$

$$+ \frac{h^2}{2}\left(\frac{\partial f}{\partial t} + f\frac{\partial f}{\partial y}\right)_{t_n, y_n} + O(h^3)$$

$O(h^3)$ – error is of order h^3.

Defn:-

If an approx. soln is equivalent to the exact Taylor series upto order h^p or equivalently if error is $O(h^{p+1})$. then the method is said to be of order p.



Local Error at a step is $O(h^{p+1})$

Total number of steps is $N = \frac{\Delta}{h}$

Global error $= N \times O(h^{p+1})$

$= \frac{\Delta}{h} \times O(h^{p+1})$.

$= O(h^{p})$.

- assumes that errors are simply cumulative

Machine epsilon, η

- smallest numbers which added to 1 gives a different number.

Roundoff error $O(\eta)$

Total roundoff error $= O(\eta) \times \#$ of steps

$= O(\eta) \times O(\frac{1}{h}) = O(\eta/h)$

Total error $= O\left(\frac{\eta}{h}\right) + O(h^p)$

t_0 $\quad t$ $\quad y(t)$

Net error is minimum approximately

when $E(h) = \frac{\eta}{h} + h^p$ is minimum.

$E'(h) = 0$

$$\Rightarrow \quad -\frac{\eta}{h^2} + p\,h^{p-1} = 0$$

$$\text{or} \quad h^{p+1} \simeq \eta$$

$$\text{or} \quad h \simeq (\eta)^{\frac{1}{p+1}}$$

$$\text{Net error} \;\propto\; (\eta)^{\frac{p}{p+1}}$$

Typical value of η. $\qquad \eta = 2.22 \times 10^{-16}$.

Predictor - Corrector Euler Method:-

predictor $\quad \tilde{y}_{m+1} = y_m + h\, f(t_m, y_m).$

corrector $\quad y_{m+1} = y_m + \frac{h}{2}\left[f(t_m, y_m) + f(t_m + h, \tilde{y}_{m+1}) \right]$

<u>Order of this method</u>:-

$$y_{m+1} = y_n + \frac{h}{2}\Big[f(t_m, y_m) + f(t_m, y_m) + h\frac{\partial f}{\partial t} + h\frac{\partial f}{\partial y} f(t_m, y_m) + O(h^2) \Big]$$

$$= y_n + h f(t_n, y_n) + \frac{h^2}{2}\left[\frac{\partial f}{\partial t} + f\frac{\partial f}{\partial y}\right]$$

$$+ O(h^3).$$

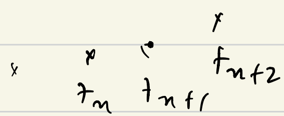
Local error $\sim O(h^3)$

Global truncation error $\sim O(h^2)$.

Runge - Kutta Methods

C. Runge (1895), W. Kutta (1901)

- self - starting, one - step methods



Basic idea :-

$$y_{n+1} = y_n + \sum_{i=1} \underbrace{w_i}_{\text{weights}} h f(\overbrace{t_n + \alpha_i h}^{\xi_i}, \overbrace{y_n + \sum_j \beta_j k_j}^{\eta_i}).$$

$$\frac{dy}{dt} = f(y, t)$$

When $f(\xi_i, \eta_i)$ are expanded around t_m, y_m upto some order, then this expression agrees with the Taylor series:-

$$y_{m+1} = y_m + h\, f(y_m, t_m) + \frac{h^2}{2}\left[\frac{\partial f}{\partial t} + f\frac{\partial f}{\partial y}\right]_{t_m, y_m}$$

$$+ \ldots$$

$$y_{m+1} = y_m + \sum_{i=1}^{p} w_i k_i$$

w_i – weights

$$k_i = h f(\xi_i, \eta_i).$$

$$\xi_i = x_m + \alpha_i h$$

$$\eta_i = y_m + \sum_{j=1}^{i-1} \beta_{ij} k_j$$

α_i, β_{ij} & w_i are adjustable parameters.

Quick review of C.M:-

$$m\ddot{q} = F$$

for conservative forces, $F = -\frac{\partial U}{\partial q}(q)$

Lagrangian:-

$$\mathcal{L}(q, \dot{q}) = T - U$$

where $T(\dot{q}) = \frac{m\dot{q}^3}{2}$

Hamiltonian :-

$$H = T + U$$

Introduce $\quad p = \dfrac{\partial L}{\partial \dot{q}}$

Rewrite H as a fn of p, q

$H(p, q)$

$$\dot{p} = -\frac{\partial H}{\partial q}$$

$$\dot{q} = \frac{\partial H}{\partial p}$$

For a S.H.O,

$$\mathcal{H}(p,q) = \frac{p^2}{2m} + \frac{1}{2} m\omega^2 x^2$$

$$\dot{x} = \frac{p}{m}$$

$$\dot{p} = -m\omega^2 x$$

with initial condns :- $p = p_0$, $x = x_0$ at $t = t_0$.

$$x(t) = x_0 \cos\omega(t-t_0) + \frac{p_0}{m\omega} \sin(\omega(t-t_0)).$$

$$p(t) = p_0 \cos\omega(t-t_0) - m\omega x_0 \sin\omega(t-t_0).$$

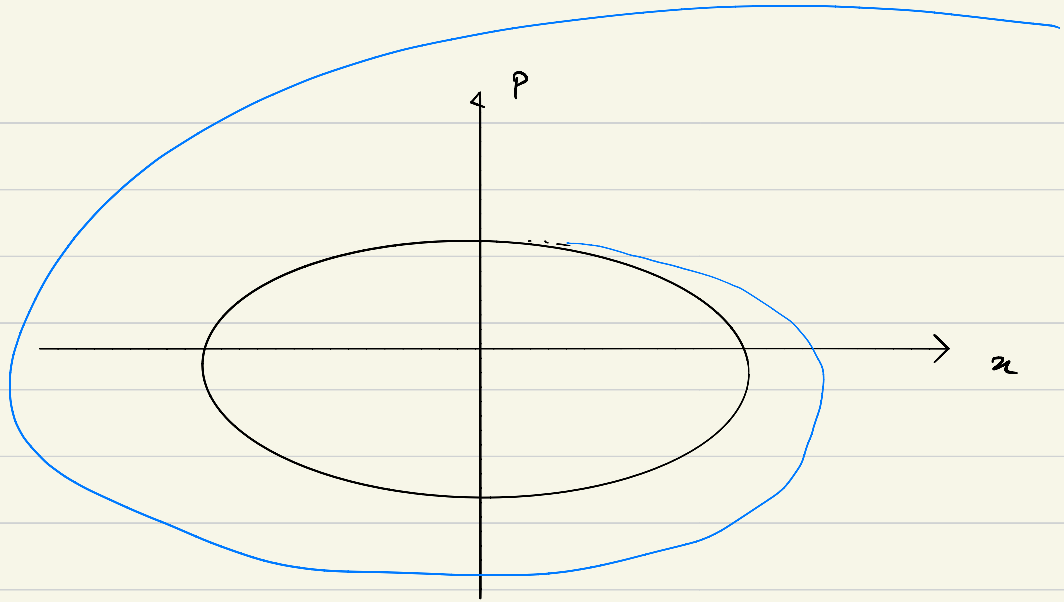
Euler's method:-

$$x_{n+1} = x_n + \frac{h p_n}{m}$$

$$p_{n+1} = p_n - h m \omega^2 x_n$$

$$H_{n+1} = \left(1 + (h\omega)^2\right) H_n$$

$$H_N = \left(1 + (\omega h)^2\right)^N H_0$$


p
x

Symplectic Euler / Euler - Cromer method :-

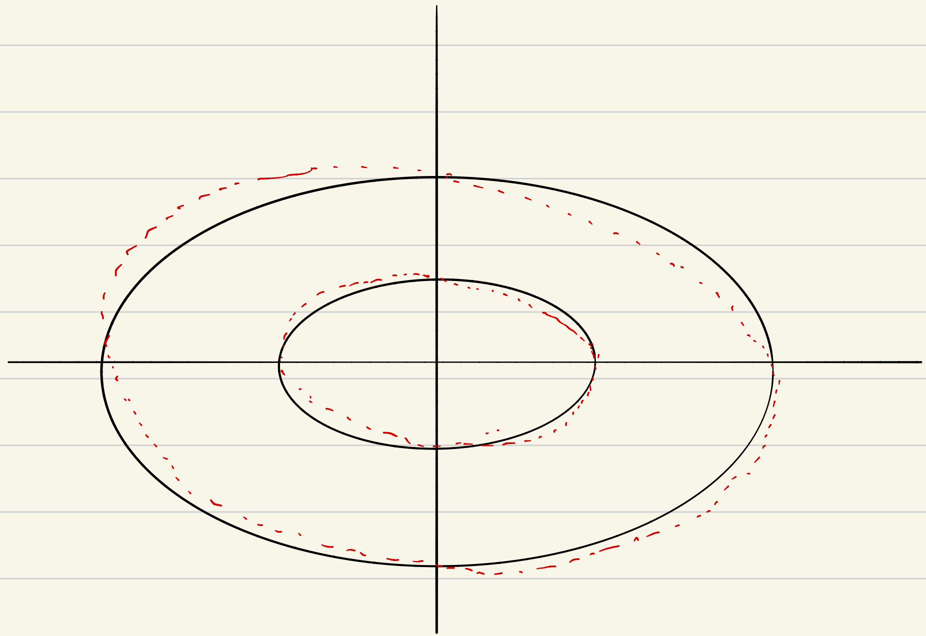
$$x_{n+1} = x_n + \frac{h p_n}{m}.$$

$$p_{n+1} = p_n - h m \omega^2 x_{n+1}$$

$$\bar{H}_h(x, p) = H(x, p) + \frac{h \omega^2 x p}{2}$$

$$\bar{H}_{h,n+1} = \bar{H}_{h,n}$$

This is not the energy, but a const. of motion!



Symplectic methods.

- conserve the energy.

1. Symplectic Euler Method:-

$$q^k_{n+1} = q^k_n + h \cdot \frac{\partial H}{\partial p_k}(q_n, p_n)$$

$$p^k_{n+1} = p^k_n - h \cdot \frac{\partial H}{\partial q^k}(q_{n+1}, p_n)$$

2. Stormer - Verlet

$$p^k_{n+1/2} = p^k_n - \frac{h}{2} \cdot \frac{\partial H}{\partial q^k}(q_n, p_n).$$

$$q^k_n = q^k_n - h \cdot \frac{\partial H}{\partial p^k}(q_n, p_{n+1/2})$$

$$p^{k}_{n+1} = p^{k}_{n+1/2} - \frac{h}{2} \cdot \frac{\partial H}{\partial q^{k}} (q_{n+1}, p_n).$$

$$A \xleftrightarrow{\quad r \quad} B$$

$$\frac{1}{r^{q}} \qquad q \neq 2$$