14. $(9x^2+y-1)-(4y-x)y'=0$ $\qquad y(1)=0$

$\vdots$

$3x^3+xy-x-2y^2=C$ $\qquad$ when $x=1$ $y=0$

$3-1=C$

$3x^3+xy-x-2y^2=2$ $\qquad y=f(x)$

$-2y^2+xy+(3x^3-x-2)=0$

$2y^2-xy-(3x^3-x-2)=0$ $\qquad$ quadratic eq. in y

$ay^2+by+c=0$

$$y=\frac{-b\pm\sqrt{b^2-4ac}}{2a}$$

$$y=\frac{x\pm\sqrt{x^2+4(2)(3x^3-x-2)}}{2(2)}$$

$y(1)=0$

$$0 = \frac{1 \pm \sqrt{1+24-8-16}}{4} = \frac{1 \pm \sqrt{1}}{4}$$ choose $-$ root

$$y = \frac{x - \sqrt{24x^3+x^2-8x-16}}{4}$$

Solution valid if $\underbrace{24x^3+x^2-8x-16 \geq 0}_{\text{done w/ a solver (Wolfram alpha)}}$

solve $24x^3+x^2-8x-16 = 0$

18. any separable is exact

$$\underbrace{M(x)}_{\text{no } y}\,dx + \underbrace{N(y)}_{\text{no } x}\,dy = 0 \longrightarrow \int M(x)\,dx = \int -N(y)\,dy$$

exact if $M_y = N_x$

$$\left.\begin{aligned} \frac{\partial M(x)}{\partial y} &= 0 \\ \frac{\partial N(y)}{\partial x} &= 0 \end{aligned}\right\} \quad M_y = N_x \quad \text{exact}$$

F. $\frac{dw}{dt} = \frac{2tw}{w^2 - t^2}$ exact ~~and separable~~

$t : x$

$w : y$

$$\underbrace{(2tw)}_{M}\,dt - \underbrace{(w^2 - t^2)}_{N}\,dw = 0$$

exact if $M_w = N_t$

$2t = 2t$ exact.

$\Psi = C$ $\Psi_t = M$ $\Psi_w = N$

$$\Psi = \int M\,dt = \int 2tw\,dt = t^2 w + h(w)$$

$$\Psi_w = t^2 + h'(w) = -w^2 + t^2$$

$h'(w) = -w^2$ $h(w) = -\frac{1}{3}w^3$

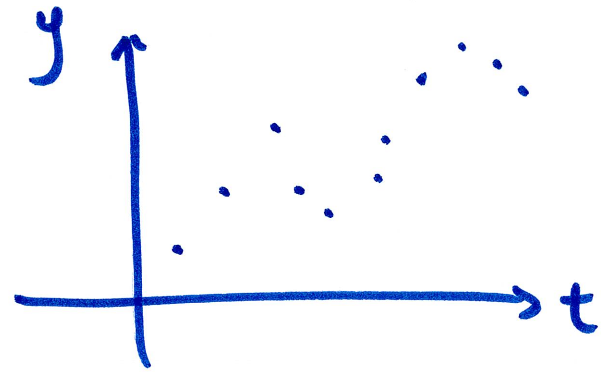
solution: $t^2 w - \frac{1}{3}w^3 = C$

so $-t^2 w + \frac{1}{3}w^3 = C$ is also ok

2.7 Numerical method: Euler's Method

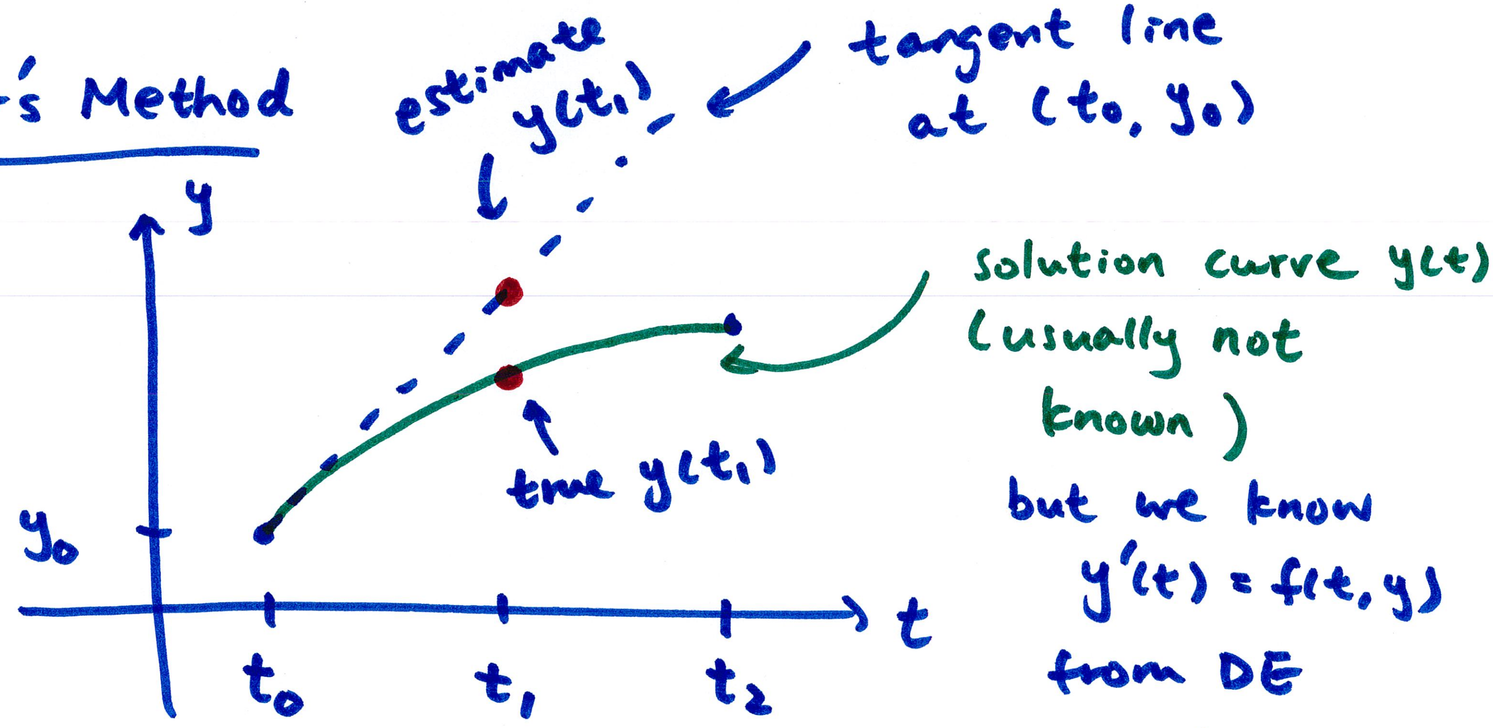
numerical method ~~d~~ solves $y' = f(t, y)$
by finding values of $y(t)$ for arbitrary t
without ~~sol~~ finding $y(t)$ explicitly.

→ find dots instead of a curve



→ normally used when eq can't be solved
by hand (analytically) or is impractical
to do so.

Euler's Method



but we know
$y'(t) = f(t,y)$
from DE
→ we know slope of y

if $h = t_1 - t_0$ is "small",
then estimate $y(t_1)$ is normally pretty good.

equation of tangent line at (t_0, y_0) is

$$y - y_0 = y'(t_0)(t - t_0)$$

$$\boxed{y = y(t_0) + y'(t_0)(t - t_0)}$$

$$\boxed{\underset{\text{new } y}{y_{n+1}} = \underset{\text{old } y}{y_n} + \underbrace{f(t_n, y_n)}_{\text{slope at old } (t,y)} \underset{\text{step size}}{h}}$$

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

h : time interval (step size)

example $y' = 0.5 - t + 2y \qquad y(0) = 1$

use step size $\underline{\underline{h = 0.1}}$ to estimate $y(0.3)$

$h=0.1$

$t=0, \quad y_0 = 1$ given

previous t, y

$t=0.1, \quad y_1 = y_0 + f(t_0, y_0)h$

$= 1 + (0.5 - 0 + 2\cdot 1)(0.1) = 1.25$

$t=0.2, \quad y_2 = y_1 + f(t_1, y_1)h$

$= 1.25 + (0.5 - 0.1 + 2\cdot 1.25)(0.1) = 1.54$

$t=0.3 \quad y_3 = 1.54 + (0.5 - 0.2 + 2\cdot 1.54)(0.1) = 1.878$

$y' = 0.5 - t + 2y$ is linear, we can solve

$y' - 2y = 0.5 - t$

$\mu = e^{\int -2\,dt} = e^{-2t}$ $\qquad y(0) = 1$

$\vdots$

$y(t) = 0.5t + e^{2t}$ $\qquad$ true $y(0.3) = 1.972$

estimat: 1.878

improve estimate, use smaller h

and more steps.

example $y' = 5 - 3\sqrt{y}$ $y(0) = 2$

use Matlab code eul.m

see Numerical Methods document on www.math.purdue.edu/ma266

use $h = 0.1$, ~~0.01, 0.005,~~ ~~0.00~~

0.01

0.001

0.0001

estimate $y(3)$

0.1 $y(3)$ 2.7352

0.01 $y(3)$ 2.7296

Numerical Methods and `.m` Files

- In order to use MATLAB routines for the Euler, Improved Euler or Runge-Kutta Methods, you will need the files `eul.m`, `rk2.m` or `rk4.m`, respectively. These files are already present on all ITaP machines as standard software. (If using your own copy of MATLAB you may need to download these files from `http://math.rice.edu/~dfield`). You may also access these files from MATLAB via the *Software Remote*:

  `http://goremote.ics.purdue.edu`

- You must first create a function file in the same directory (or folder) as your MATLAB. Here is one way. After MATLAB has been opened, pull down the **File** menu and select **New M-File**. A window will pop up for you to create your function file. For example, to create a function file for the function $f(x,y) = 6x^3 - e^{2y} + \sqrt{x}/y$, type:

  ```
  function W=fcn1(x,y)
  W=6*x^3-exp(2*y)+sqrt(x)/y;
  ```

  (Don't forget the ";" at the end.)

- Save this file as a `.m` file with the **same** name as your function. The above example would be saved as `fcn1.m`. You can check if your function has been saved by typing something like the following at a MATLAB prompt: `fcn1(0,3)`

  You should get the value of $f(0,3)$.

- Your initial value problem should have the form: $\begin{cases} y' = f(x,y) \\ y(x_0) = y_0 \end{cases}$.

  Assuming $f(x,y)$ was saved as the file `fcn1.m`, the syntax for `eul` (as well as `rk2` and `rk4`, just replace `eul`) will be:

  ```
  eul('fcn1',[x0,xf],y0,h)
  ```

  where `x0` and `xf` denote the initial and final values of x, respectively, `y0` is the initial value of y, and `h` is the step size. (Your version of MATLAB may not utilize brackets. Type `help eul` to find out.) To approximate the actual solution to the IVP at `xf`, with given `h`, using `eul`, just type the following at a MATLAB prompt:

  ```
  [x,y]=eul('fcn1',[x0,xf],y0,h);
  ```

  The approximations y_0, y_1, y_2, $\ldots$, y_n are stored in the matrix **y**

- To print them out, type: `[x,y]`

- To plot them, type: `plot(x,y)`