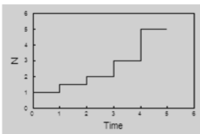
Exponential Growth Models

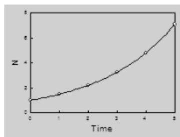
FDE model

$$N_{t+1} = N_t + rN_t$$

ODE model

$$\frac{dN}{dt} = rN$$





Errors in Euler's Method – Exponential Model

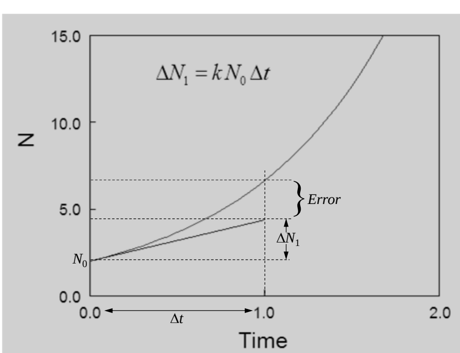
$k = 0.4$
$N_0 = 2$

$$\frac{dN}{dt} = kN$$

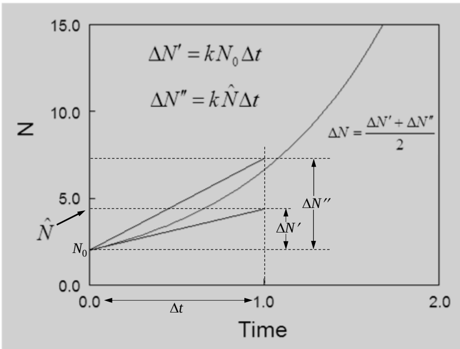
$$N = N_0 e^{kt}$$

Time	Obs.	Euler's			Exact
		1.0	0.1	0.01	
0	2	2.0	2.0	2.0	2.0
2	5	3.92	4.38	4.44	4.45
6	20	15.06	21.04	21.94	22.05
10	109	57.85	101.0	108.3	109.2

How does Euler's method work ?



Better methods – Runge-Kutta second order



Euler's Method and errors – logistic model

$$\frac{dN}{dt} = rN\left(1 - \frac{N}{K}\right)$$

Analytical solution (from Calculus)

$$N = \frac{rN_0}{\frac{rN_0}{K} + \left(r - \frac{rN_0}{K}\right)e^{-rt}}$$

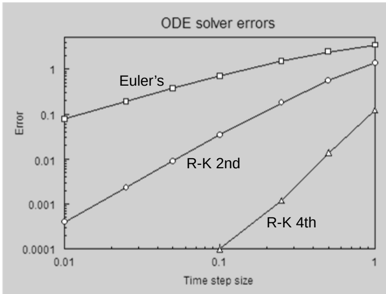
$$N = \frac{rN_0}{\frac{rN_0}{K} + \left(r - \frac{rN_0}{K}\right)e^{-rt}}$$

r = 1.5, K = 100,
N_0 = 2

At t = 4.0
N_{exact} = 89.1696

Smaller
Δt → more accurate

Δt	N
2.0	29.657
1.0	57.917
0.5	78.888
0.1	87.895
0.01	89.053

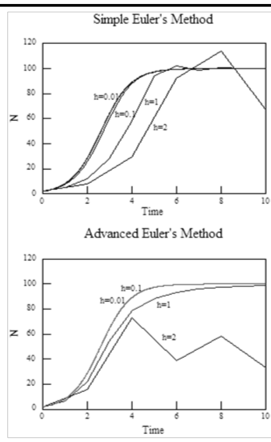


Better ODE solvers have…
- Smaller errors
- Errors that decrease faster

Logistic ODE
r = 0.5
K = 100
t = 3.0

Large step sizes lead to large errors but may also lead to other behaviors, like oscillating solutions

Advanced methods can provide greater accuracy but with large step size, solution may still oscillate.



Programming ideas…

```
#include <stdheaders.h>

int main( void )
{
        int i, steps;
        float k, dn, dt, tfinal;

        k = 0.4;
        tfinal = 10;
        dt = 1.0;
        steps = tfinal / dt;
        float N[steps+1], t[steps+1];

        t[0] = 0;
        N[0] = 2;

        for( i=0; i<steps; i++ )
                {
                        dn = k * N[i] * dt;
                        N[i+1] = N[i] + dn;
                        t[i+1] = t[i] + dt;
                }

        for( i=0; i<=steps; i++ )
                printf( "At time %4.2f, N = %6.4f\n", t[i], N[i] );

        return 0;
}
```

Run lcc with expSteps program...

Errors in Euler's Method – Exponential Model

$k = 0.4$
$N_0 = 2$

$$\frac{dN}{dt} = kN$$

$$N = N_0 e^{kt}$$

Time	Obs.	Euler's			Exact
		1.0	0.1	0.01	
0	2	2.0	2.0	2.0	2.0
2	5	3.92	4.38	4.44	4.45
6	20	15.06	21.04	21.94	22.05
10	109	57.85	101.0	108.3	109.2

Programming Errors

1. Syntax errors
2. Logical errors

View compiler error statements, correct

Debugger -
A program that can run and control your program

- Start and stop your program wherever you want
- Monitor the value of all variables while the program is running