

1.7 Slope fields and stability

In this section we introduce techniques for studying solutions for differential equations that doesn't require us to write down explicit formulas for the solutions. These methods are examples of "qualitative analyses" because they provide qualitative (as opposed to quantitative) information about the behavior of solutions. The lack of quantitative information might give one the impression that these qualitative results are somehow lacking in rigor. Do not be fooled! Not only are the qualitative results we obtain mathematically rigorous in the sense that they are provable statements, but often in applications such information is just as useful as quantitative information, if not more useful!

Slope fields

Consider the differential equation

$$\frac{dy}{dt} = y(1 - y). \quad (1.7.1)$$

If at some time t we happen to know the value of a solution $y(t)$, then the differential equation tells us the derivative of the solution at that time. For example, if we have a solution $y(t)$ such that $y(1) = 0.3$, then the differential equation tells us that

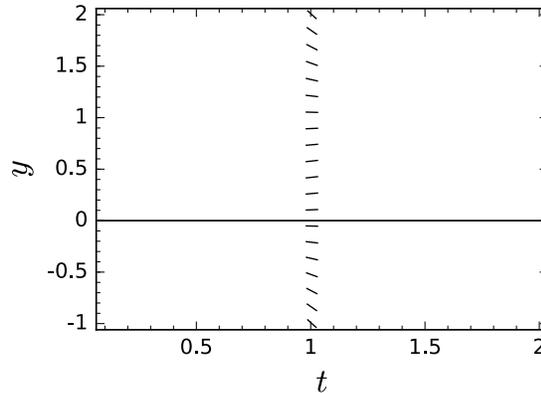
$$y'(1) = y(1)(1 - y(1)) = 0.3(1 - 0.3) = 0.21.$$

This tells us the "direction" in the ty plane that a solution starting out at $t = 1$ and $y(1) = 0.3$ will go.

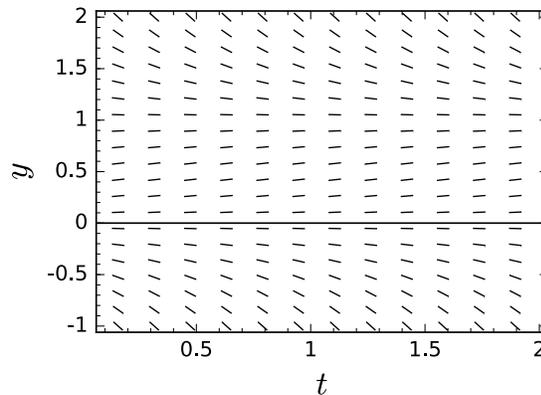
Consider now some other solution $y(t)$, say with $y(1) = 0.1$. The differential equation tells us that for this solution we have $y'(1) = 0.1(1 - 0.1) = .09$. We can continue on this way, considering a variety of possible solutions, and end up with the following table:

t	$y(t)$	$y'(t)$
1	0	0
1	0.1	0.09
1	0.2	0.16
1	0.3	0.21
1	0.4	0.24
1	0.5	0.25
	etc.	

What do all these numbers tell us? One way to understand them is to interpret them in terms of the graphs of the solutions. The derivative tells us the slope of the graph in the ty plane, so this list of numbers tells us the slopes of various solutions at time $t = 1$ and with various values of y . If we plot these little slopes, we end up with the following picture.



Of course, there is no reason to focus only on $t = 1$. For any value of t and y we can use the differential equation to compute the slope of the solution passing through that point. Plotting these slopes at each point for the differential equation (1.7.1) yields the following picture.



The plot above is called the **slope field plot** for the differential equation (1.7.1). Since the slope field plot shows the slopes of any solutions that exist, we can use the plot to easily imagine how solutions to the differential

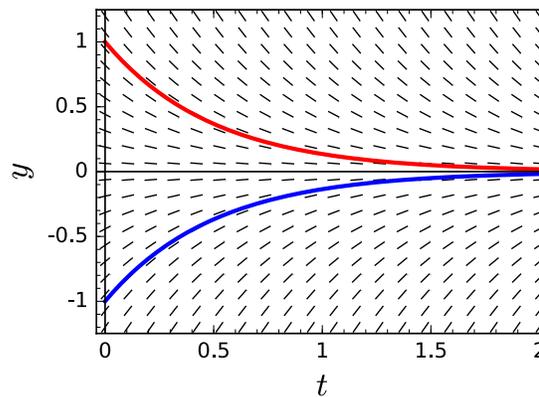
equation behave – they “follow” the slope field. The next example illustrates this in a situation where we know the exact formula for the solutions.

Example 1.7.1. Consider the differential equation

$$\frac{dy}{dt} = -2y. \quad (1.7.2)$$

We know from previous work that solutions take the form $y(t) = Ce^{-2t}$.

The following picture shows the slope field for (1.7.2) together with the plots of two different solutions. Notice that the solution $y(t) = e^{-2t}$ (in red) and the solution $y(t) = -e^{-2t}$ (in blue) both follow the slope field lines.

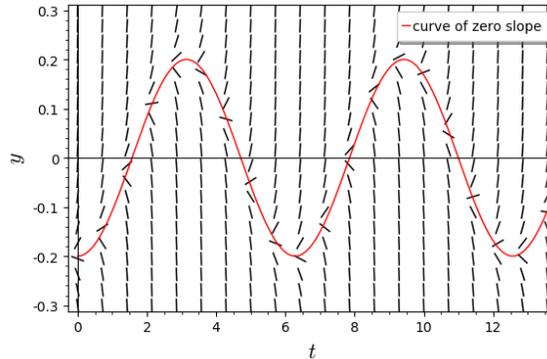


Example 1.7.2. Consider the differential equation

$$\frac{dy}{dt} = 5y + \cos(t).$$

Note that the slope is zero along the curve $y = -\frac{1}{5} \cos(t)$. For y values below the curve, the slope is negative, while above the curve the slope is positive.

The plot slope for this ODE is as follows/



What happens to solutions that start off on the curve of zero slope?

Equilibrium solutions

Notice that the slopefield for the equation (1.7.1) has two special values of y . At $y = 0$ and $y = 1$, the slope is zero. These correspond to two constant solutions to the differential equation (1.7.1):

$$y(t) = 0 \quad \text{and} \quad y(t) = 1.$$

Constant solutions are called *equilibrium solutions*, and represent states of whatever is being modeled that are unchanging. For example, in a population model, an equilibrium solution represents a population size that is always the same.

In order to find an equilibrium solution to an equation of the form

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

we simply assume that the solution takes the form $y(t) = y_*$ for some constant y_* , and solve algebraically for

$$0 = f(t, y_*)$$

for y_* .

Example 1.7.3. *The differential equation*

$$\frac{dy}{dt} = y(1 - y) \tag{1.7.3}$$

has equilibrium solutions $y(t) = 0$ and $y(t) = 1$.

Example 1.7.4. Consider the differential equation

$$\frac{dy}{dt} = (5 - y)(\cos(t) - y).$$

There is one equilibrium solution, $y(t) = 5$.

Why isn't the function $y(t) = \cos(t)$ an equilibrium solution?

- The function $y(t) = \cos(t)$ is not a constant function – an equilibrium solution is, by definition, a constant function!
- Also, the function $y(t) = \cos(t)$ is not even a solution to the differential equation!

Equilibrium solutions are important in applications because they represent states that persist over long periods of time. In many applications, however, it may not be the case that equilibrium solutions can be attained exactly. For this reason, we also look at the behavior of solutions nearby to equilibrium solutions.

Example 1.7.5. Consider the differential equation

$$\frac{dy}{dt} = y(1 - y).$$

Let us study the equilibrium solution $y(t) = 1$. From the slope field, and also from the differential equation, we can see any solution that is slightly less than 1 will be increasing. Similarly, any solution that is slightly greater than 1 will be decreasing. Thus nearby solutions flow towards the equilibrium solution $y(t) = 1$. In applications, this means that small “errors” decrease nearby to the solution $y(t) = 1$. Due to this feature, we say that the equilibrium $y(t) = 1$ is *stable*.

Analogous analysis shows that solutions close by to the equilibrium solution $y(t) = 0$ move away from that equilibrium. Thus we say that $y(t) = 0$ is *unstable*.

We make the following definitions:

- If $y(t) = y_*$ is an equilibrium solution such that all nearby solutions move towards y_* , then we say that $y(t) = y_*$ is **stable**.
- If $y(t) = y_*$ is an equilibrium solution such that all nearby solutions move away from y_* , then we say that $y(t) = y_*$ is **unstable**.

- If $y(t) = y_*$ is an equilibrium solution such that some nearby solutions move towards y_* and some move away, then we say that $y(t) = y_*$ is **semi-stable**.

In practice, unstable equilibrium solutions are rarely seen “in nature”, while stable equilibrium solutions are often seen.

Activity 1.7.1. *Then plot the slope field for the equation*

$$\frac{dy}{dt} = (y - 2)(y + 3).$$

Find the equilibrium solutions. Determine whether they are stable, unstable, or semistable.

Linearization stability

We now introduce another tool that can be used to deduce stability of equilibrium solutions: linearization. We focus attention on differential equation of the form

$$\frac{dy}{dt} = f(y). \quad (1.7.4)$$

*Note: Equations of this form, where the right side depends only on the unknown y and not on t , are called **autonomous**. Our discussion generalizes to the case where the right side depends on both t and y , but we work with autonomous equations of the form (1.7.4) for simplicity.*

Suppose that $y(t) = y_*$ is an equilibrium solution for (1.7.4). This means that $f(y_*) = 0$. Let’s examine the behavior of solutions that are nearby. This means we look at solutions of the form $y(t) = y_* + u(t)$ where we assume $u(t)$ is small. (Note: There might not be such a solution that lasts very long, but if $u(0)$ is small, then $u(t)$ will be small for at least some short amount of time.) Plugging $y(t) = y_* + u(t)$ in to the differential equation yields

$$\frac{du}{dt} = f(y_* + u).$$

So long as f is differentiable at y_* , we can use Taylor’s theorem to write

$$f(y_* + u) = f(y_*) + f'(y_*)u + \text{remainder},$$

where the remainder is of size $|u(t)|^2$. Since $y(t) = y_*$ is an equilibrium solution we have $f(y_*) = 0$. Suppose also that $f'(y_*) \neq 0$. Hence as long as u is very small, the linear term is larger than the error term and we have

$$f(y_* + u) \approx f'(y_*)u.$$

In other words, we have

$$\frac{du}{dt} \approx f'(y_*)u.$$

Example 1.7.6. Consider differential equation

$$\frac{dy}{dt} = y(1 - y)$$

and the equilibrium $y(t) = 1$. The right side of the differential equation is given by the function $f(y) = y(1 - y)$. We have $f'(1) = -1$. Thus when u is small, solutions of the form $y(t) = y_* + u(t)$ satisfy

$$\frac{du}{dt} \approx -u.$$

The equation

$$\frac{du}{dt} = f'(y_*)u \tag{1.7.5}$$

is called the **linearization (or linearized equation)** of (1.7.4) at y_* . The linearized equation tells us what happens to solutions to the main equation (1.7.4) nearby to the equilibrium. We use the linearized equation because it is very easy to solve: the general solution to (1.7.5) is

$$u(t) = Ce^{f'(y_*)t}.$$

Thus the sign of $f'(y_*)$ tells us whether nearby solutions move towards the equilibrium (ie, u decreases in size) or whether they move away from the equilibrium (ie, u increases in size). From this discussion we conclude the following:

- If $f(y_*) = 0$ and $f'(y_*) > 0$, then $y(t) = y_*$ is an unstable equilibrium solution.
- If $f(y_*) = 0$ and $f'(y_*) < 0$, then $y(t) = y_*$ is a stable equilibrium solution.
- If $f(y_*) = 0$ and $f'(y_*)$ is either zero or not defined, then we cannot deduce anything from the linearization and need to examine the situation more carefully.

Stability deduced from the linearization is sometimes called **linearization stability**.

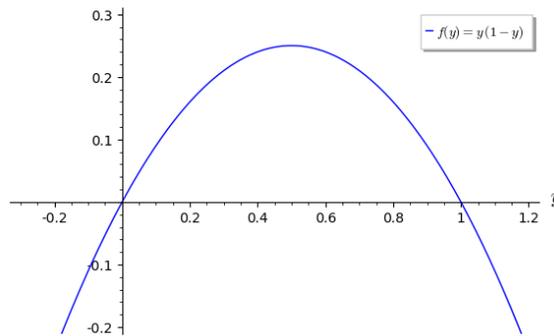
All of this tells us that we can often deduce much about solutions to (1.7.4) by plotting the function f . The roots of the function tell us the

equilibrium solutions, the slope at those roots tells us the stability of those equilibrium points. We can also deduce the ranges of increasing/decreasing from the plot of f .

Example 1.7.7. Consider the differential equation

$$\frac{dy}{dt} = y(1 - y),$$

so that the right side is given by the function $f(y) = y(1 - y)$. Here is a plot of f :



From the plot we deduce the following:

- Since $f(0) = 0$ and $f'(0) > 0$ we have that $y(t) = 0$ is an unstable equilibrium solution.
- Since $f(1) = 0$ and $f'(1) < 0$ we have that $y(t) = 1$ is a stable equilibrium solution.
- If a solution $y(t)$ is between 0 and 1, then it increases towards the equilibrium at 1.
- If a solution $y(t)$ is greater than 1, then it decreases towards the equilibrium at 1.
- If a solution $y(t)$ is less than 0, then it decreases away the equilibrium at 0.

What is the connection between the plot of f and the slope field plot?

Summary

- By plotting the slope field, we can gain a visual understanding of how solutions to the differential equation behave.
- One important issue is the stability of equilibrium solutions.
- Slope fields can be especially helpful for determining the stability of equilibrium solutions.
- We can also use linearization to study stability of equilibrium solutions.
- For autonomous equations $\frac{dy}{dt} = f(y)$ we can relate the plot of f to the slope field, and to the stability of equilibrium solutions.

Activity 1.7.2. Consider the differential equation

$$\frac{dy}{dt} = \cos(y).$$

Find the equilibrium solutions and determine their stability.

Exercise 1.7.1. Perform a qualitative analysis (determine equilibrium solutions, sketch slope field, determine stability of equilibrium points) of the ODE

$$\frac{dy}{dt} = 1 - y^2$$

Exercise 1.7.2. Perform a qualitative analysis (determine equilibrium solutions, sketch slope field, determine stability of equilibrium points) of the ODE

$$\frac{dy}{dt} = y^2 - 1$$

Exercise 1.7.3. Perform a qualitative analysis (determine equilibrium solutions, sketch slope field, determine stability of equilibrium points) of the ODE

$$\frac{dy}{dt} = y^3 + y^2 - 6y$$

Exercise 1.7.4. Perform a qualitative analysis (determine equilibrium solutions, sketch slope field, determine stability of equilibrium points) of the ODE

$$\frac{dy}{dt} = \frac{1}{1 + y^2}$$

Exercise 1.7.5. Perform a qualitative analysis (determine equilibrium solutions, sketch slope field, determine stability of equilibrium points) of the ODE

$$\frac{dy}{dt} = (2y - y^2)e^{-y}$$

Exercise 1.7.6. Perform a qualitative analysis (determine equilibrium solutions, sketch slope field, determine stability of equilibrium points) of the ODE

$$\frac{dy}{dt} = \cos(y)$$

Exercise 1.7.7.

For each differential equation below you need to

- Find a formula for several “typical” solutions. This requires using methods from previous sections!
- Make plots (using technology) of these solutions,
- Generate a slope field plot (using technology), and
- Superimpose your solution plots on top of the slope field plot.

Turn in printouts of the final, combined plot. You should also write a sentence or two describing the long-term behavior of solutions.

1. $\frac{dy}{dt} = -y + t + 1$;
2. $\frac{dy}{dt} = y(9 - y^2)$;
3. $\frac{dy}{dt} = y + 2t$.

Exercise 1.7.8. The size of the fish population of a lake is modeled by a logistic equation. It is known that under **ideal** conditions the per capita growth rate of fish is 80%. It is also known that the lake has the carrying capacity of 20 000 fish. It is estimated that currently this lake has 15 000 fish. The owners of the lake are considering harvesting.

1. Write down the logistic equation which models the size of the fish population. What is the current per capita growth rate of the population?

2. Modify the logistic equation to accommodate for (gradual) annual harvesting at the relative rate of 60%. Perform the qualitative analysis of your modified equation without any use of “technology”. Then check your predictions using technology.

Exercise 1.7.9. Consider the differential equation

$$\frac{dy}{dt} = y \left(\frac{y}{2e^t} - 1 \right)$$

1. Verify that $y_1(t) = 0$ and $y_2(t) = 4e^t$ are both solutions of this differential equation.
2. Produce a rough sketch of the slope field corresponding to this equation **without** using “technology”. Pay particular attention to the slope field along the curve $y = 2e^t$, between the curves $y = 2e^t$ and $y = 4e^t$, between $y = 2e^t$ and $y = 0$, above the curve $y = 4e^t$ and below the line $y = 0$. Check your answer using technology.
3. A certain variable quantity $y(t)$ satisfies the differential equation in this problem. It is known that $y(0) \approx 4$, but the exact value of $y(0)$ is not known. What can you say about the long term behavior of this quantity i.e. about $\lim_{t \rightarrow +\infty} y(t)$?

1.8 Introduction to numerical methods

For “generic” ODEs, there is no hope of writing down an explicit formula for a typical solution. Fortunately, we can use computers in order to obtain numerical approximations of solutions.

Suppose we have a differential equation of the form

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

We can compare the values of a solution y at times t and $t + \Delta t$ by integrating – the Fundamental Theorem of Calculus tells us that

$$y(t + \Delta t) = y(t) + \int_t^{t+\Delta t} f(\tau, y(\tau)) d\tau. \quad (1.8.1)$$

Suppose now that we know the value $y(t)$, but do not know the value of the function y for times larger than t . If Δt is not too large, then we can approximate the integral in (1.8.1) using the left endpoint approximation

$$\int_t^{t+\Delta t} f(\tau, y(\tau)) d\tau \approx f(t, y(t)) \Delta t; \quad (1.8.2)$$

see Figure 1.8.1.

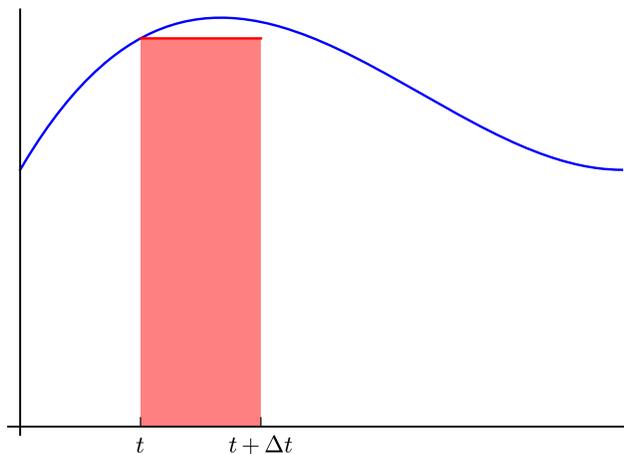


Figure 1.8.1: If Δt is small then the approximation (1.8.2) is “reasonably good.”

Using (1.8.2) and (1.8.1) we obtain the approximation

$$y(t + \Delta t) \approx y(t) + f(t, y(t)) \Delta t, \quad (1.8.3)$$

which is “reasonably good” if Δt is small.

We now show how to iteratively use the approximation (1.8.3) in order to construct a numerical approximation of the solution to the initial value problem

$$\frac{dy}{dt} = f(t, y) \quad y(t_0) = y_0. \quad (1.8.4)$$

We begin by choosing a fixed number Δt and constructing a list of times t_k , indexed by counter k , that each differ by Δt :

k	t_k
0	t_0
1	$t_1 = t_0 + \Delta t$
2	$t_2 = t_0 + 2(\Delta t)$
3	$t_3 = t_0 + 3(\Delta t)$
\vdots	\vdots
k	$t_k = t_0 + k(\Delta t)$

Notice that for each k we have $t_{k+1} = t_k + \Delta t$.

We now construct a list of values y_k that approximate the values of the solution $y(t)$ to (1.8.4) at the times t_k . The initial condition in (1.8.4) provides the value y_0 . The approximation (1.8.3) states that

$$y(t_1) \approx y_0 + f(t_0, y_0)\Delta t.$$

Thus we define

$$y_1 = y_0 + f(t_0, y_0)\Delta t.$$

Similarly, (1.8.3) implies that

$$y(t_2) \approx y(t_1) + f(t_1, y(t_1))\Delta t.$$

We don't know what $y(t_1)$ is, but we do know that $y(t_1) \approx y_1$. Using this approximation we have

$$y(t_2) \approx y_1 + f(t_1, y_1)\Delta t.$$

Thus we set

$$y_2 = y_1 + f(t_1, y_1)\Delta t.$$

Repeating this procedure, we see that the numbers y_k , defined by

$$y_k = y_{k-1} + f(t_{k-1}, y_{k-1})\Delta t, \quad (1.8.5)$$

approximate the value of $y(t_k)$. The approximation scheme (1.8.5) is called **Euler's method**.

We can summarize this in the following table:

k	t_k	y_k
0	t_0	y_0
1	$t_1 = t_0 + \Delta t$	$y_1 = y_0 + f(t_0, y_0)\Delta t$
2	$t_2 = t_0 + 2(\Delta t)$	$y_2 = y_1 + f(t_1, y_1)\Delta t$
3	$t_3 = t_0 + 3(\Delta t)$	$y_3 = y_2 + f(t_2, y_2)\Delta t$
\vdots	\vdots	
k	$t_k = t_0 + k(\Delta t)$	$y_k = y_{k-1} + f(t_{k-1}, y_{k-1})\Delta t$

Let's now look at how this approximation works for an equation for which we know the exact solution.

Example 1.8.1. Consider the initial value problem

$$\frac{dy}{dt} = y^2 \quad y(0) = 1.$$

We know that the solution to the initial value problem is

$$y(t) = \frac{1}{1-t}. \quad (1.8.6)$$

Let's now construct an approximation using Euler's method.

We choose $\Delta t = 0.1$ and decide to approximate the solution $y(t)$ on the interval $0 \leq t \leq 0.5$. Since $t_0 = 0$, we are interested in the times

k	t_k
0	$t_0 = 0$
1	$t_1 = 0.1$
2	$t_2 = 0.2$
3	$t_3 = 0.3$
4	$t_4 = 0.4$
5	$t_5 = 0.5$

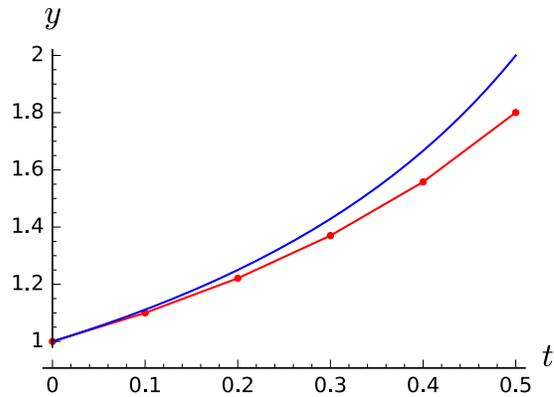
We now use (1.8.5) in order to construct the list of approximate values. Since $f(t, y) = y^2$ and $y_0 = 1$, we compute

$$y_1 = y_0 + f(t_0, y_0)\Delta t = 1 + (0.1)^2(0.1) = 1.001.$$

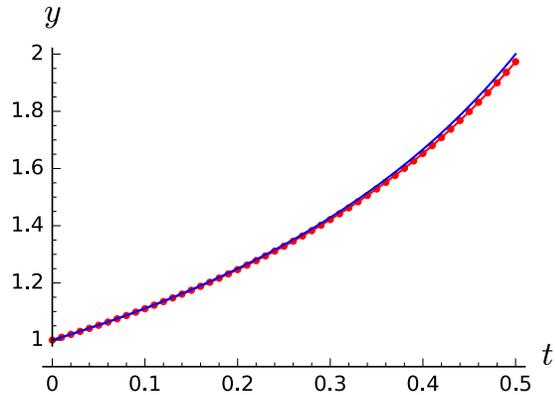
Proceeding iteratively, we obtain the following table:

k	t_k	y_k
0	$t_0 = 0$	$y_0 = 1$
1	$t_1 = 0.1$	$y_1 = 1.100$
2	$t_2 = 0.2$	$y_2 = 1.221$
3	$t_3 = 0.3$	$y_3 = 1.370$
4	$t_4 = 0.4$	$y_4 = 1.558$
5	$t_5 = 0.5$	$y_5 = 1.800$

Here is a plot of the approximating points (t_k, y_k) , as computed with $\Delta t = 0.1$, in red and a plot of the actual solution $y(t)$ in blue. Notice that the error accumulates as time progresses.



As evidenced by this plot, the approximation scheme with $\Delta t = 0.1$ is not bad, but is not great. We can obtain a better approximation by choosing $\Delta t = 0.01$, in which case we must plot 50 points. Here is a plot of the approximating points (t_k, y_k) in red, now with $\Delta t = 0.01$, together with a plot of the actual solution $y(t)$ in blue. Notice that the approximation here is much better than the previous approximation.



Activity 1.8.1. Use Euler's method to construct an approximate solution to the initial value problem

$$\frac{dy}{dt} = 1 - y, \quad y(0) = 3$$

where $\Delta t = 0.5$. Do this by hand, so that you get a sense of how the method works.

Exercise 1.8.1. Use Euler's method with a step size of $\Delta t = 0.5$ to approximate the solution to

$$\frac{dy}{dt} = y(1 - y) \quad y(0) = 0.2$$

on the time interval $[0, 3]$.

Exercise 1.8.2. Create an approximate table of values for the solution of the IVP

$$\frac{dy}{dt} = \cos(y), \quad y(0) = 0$$

over the interval $0 \leq t \leq 1$. Please use the step-size of $\Delta t = 0.25$.

1.9 Euler's method with Sage

We can easily make Sage do the computations in Euler's method. Here I explain how to build the code used in Example 1.8.1. First we define variable y and also the function $f(y) = y^2$ that determines the right hand side:

```
var('y')
f(y) = y^2
```

Then we define Δt and set it equal to 0.1. We also define y_0 , which we set equal to 1:

```
var('y')
f(y) = y^2

y0 = 1
deltaT = 0.1
```

Next we construct a data structure that will hold the various values of t_k and y_k . To do this we make a variable called `steps` that tells us how many time steps we will take. In this example, we set `steps` equal to 5. We call that data structure `eulerData`. For now, we define all the t_k values to be $t_k = k(\Delta t)$ and all the y_k values to be the same as y_0 .

```
var('y')
f(y) = y^2

y0 = 1
deltaT = 0.1
steps=5

eulerData = [[k*deltaT,y0] for k in range(0,steps+1)]

eulerData
```

The variable `eulerData` is organized as follows: the quantity `eulerData[2]` tells us the entries in the row corresponding to $k = 2$.

```
var('y')
f(y) = y^2

y0 = 1
deltaT = 0.1
steps=5

eulerData = [[k*deltaT,y0] for k in range(0,steps+1)]

eulerData[2]
```

If we want only to see the time t_2 , then we need to ask Sage to show us `eulerData[2][0]`, while if we want Sage to show us y_2 , we need to ask to be shown `eulerData[2][1]`. Try it:

```
var('y')
f(y) = y^2

y0 = 1
deltaT = 0.1
steps=5

eulerData = [[k*deltaT,y0] for k in range(0,steps+1)]

eulerData[2][0]
```

What we want to do now is systematically go through and update all the y_k values, starting at $k = 1$ and ending at $k = \text{steps}$. We can do this with a loop:

```
var('y')
f(y) = y^2

y0 = 1
deltaT = 0.1
steps=5

eulerData = [[k*deltaT,y0] for k in range(0,steps+1)]

for k in [1..steps]:
    eulerData[k][1]=eulerData[k-1][1]+deltaT*f(eulerData[k-1][1])

eulerData
```

Finally, we can plot the resulting approximate solution:

```
var('y')
f(y) = y^2

y0 = 1
deltaT = 0.1
steps=5

eulerData = [[k*deltaT,y0] for k in range(0,steps+1)]

for k in [1..steps]:
    eulerData[k][1]= eulerData[k-1][1] + deltaT*f(eulerData[k-1][1])
```

```
eulerPlot = list_plot(eulerData, color="red", plotjoined=
    true, marker='.', axes_labels=['t$', '$y$'])
eulerPlot.show()
```

Built-in ODE solvers

In principle, you can code up Euler's method by hand (as is done above) any time you need a numerical approximation for a differential equation. However, there are much better algorithms available for doing this. The "industry standard" algorithm is a procedure called *Runge-Kutta*. Sage has a built-in Runge-Kutta function. The inputs to the Runge-Kutta function are

- the right side of the differential equation ($f(y)$),
- the unknown variable (y),
- the initial conditions (t_0, y_0),
- the independent variable (usually t)
- the endpoint of the time interval you are interested in,
- the number of time steps you want.

Here is an example that applies the built-in Runge-Kutta function to the problem

$$\frac{dy}{dt} = y^2, \quad y(0) = 1$$

with $\Delta t = 0.25$ on the time interval $0 \leq t \leq 1$. Notice that the output is a list of times and values, just as we got from the Euler's method.

```
var('y','t')

f(y) = y^2 # rhs of ODE

t0, y0 = 0, 1 # initial conditions

numsoln = desolve_rk4( f(y), y, ics=[t0,y0], ivar=t,
    end_points=1, step=0.25)

numsoln
```

Here is an application of the Runge-Kutta function that generates a plot.

```

var('y','t')

f(y) = y^2 # rhs of ODE

t0, y0 = 0, 1 # initial conditions

numsoln = desolve_rk4( f(y), y, ics=[t0,y0], ivar=t,
                      end_points=1, step=0.25)

numplot = list_plot(numsoln, marker=".", plotjoined=true)
numplot.show(figsize=[4,3])

```

Important: For the exercises in this section, I want you to use the Euler's method code that we've constructed up by hand. But after this assignment is done, it is better to just use the built-in Runge-Kutta algorithm.

Exercise 1.9.1. As you know, the function $y(t) = e^t$ is the unique solution of the IVP

$$\frac{dy}{dt} = y, \quad y(0) = 1.$$

In this problem you get to investigate the power and the limitation of the Euler method.

1. Make a table which shows the actual values of $y(0)$, $y(0.25)$, $y(0.5)$, $y(0.75)$ and $y(1)$.
2. What does the Euler method for step-size of $\Delta t = 0.25$ predict for the values of $y(0)$, $y(0.25)$, $y(0.5)$, $y(0.75)$ and $y(1)$?
3. What does the Euler method for step-size of $\Delta t = 0.05$ predict for the values of $y(0)$, $y(0.25)$, $y(0.5)$, $y(0.75)$ and $y(1)$?
4. Make a table that lists the errors that the two approximation methods make. Then write a sentence or so interpreting what you see.
5. Represent your findings graphically in the t - y plane. Then write a sentence or so interpreting what you see.

Exercise 1.9.2. Consider the initial value problem

$$\frac{dy}{dt} = y^3, \quad y(0) = 1.$$

1. Using the Euler's Method with $\Delta t = 0.5$, graph an approximate solution over the interval $0 \leq t \leq 1$.

2. What happens if you make Δt considerably smaller than above (e.g. $\Delta t = 0.05$)?
3. Verify that the function

$$y(t) = \frac{1}{\sqrt{1-2t}}$$

is a solution to the IVP in this problem. Use this knowledge to interpret the findings in parts a) and b) of this problem.

Exercise 1.9.3. Construct a numerical solution to the initial value problem

$$\frac{dy}{dt} = -y^2 + t, \quad y(2) = 2.$$

Use your solution to estimate $y(10)$.

Exercise 1.9.4. Consider the differential equation

$$\frac{dy}{dt} = y(y-3)(y-6).$$

1. Perform qualitative analysis of this equation. What is the expected behavior for solutions with $y(0) = 1$?
2. In addition to the differential equation above consider the initial condition $y(0) = 1$. Perform Euler method with the step size $\Delta t = 0.5$ in order to understand $y(t)$ for $0 \leq t \leq 2$. Make sure your answer includes the table of values and the corresponding piece-wise linear graph.
3. In light of part (a) please discuss the reliability of your answer in part (b).

1.10 Optional Topic: Trapezoid and Runge-Kutta algorithms

In this section we consider two numerical approximation schemes that are more sophisticated than the Euler method presented above.

For simplicity we work with autonomous equations of the form

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

Our goal is to approximate the value of $y(t_* + \Delta t)$ using information about the value of $y(t_*)$. From the fundamental theorem of calculus we know that

$$y(t_* + \Delta t) = y(t_*) + \int_{t_*}^{t_* + \Delta t} f(y(t)) dt. \quad (1.10.1)$$

Thus if we can construct an approximation of the integral, we have the desired approximation.

Let's take a minute to discuss some ways to approximate an integral. The Euler method is to approximate the integral by the left-endpoint approximation:

$$\int_a^b g(s) ds \approx g(a) (b - a).$$

Another possible approximation is the trapezoid approximation

$$\int_a^b g(s) ds \approx \frac{1}{2} [g(a) + g(b)] (b - a). \quad (1.10.2)$$

This can be generalized to the approximation

$$\int_a^b g(s) ds \approx [\alpha g(a) + \beta g(b)] (b - a), \quad (1.10.3)$$

where α and β are constants such that $\alpha + \beta = 1$. One should interpret trapezoidal approximations (1.10.2) and (1.10.3) as a weighted average of the left and right endpoint approximations.

Let's now apply the trapezoidal approximation to (1.10.1). For notational convenience, we set $y_* = y(t_*)$. Then the trapezoidal approximation (1.10.2) becomes

$$y(t_* + \Delta t) \approx y_* + \frac{1}{2} (f(y_*) + f(y(t_* + \Delta t))) \Delta t.$$

1.10. OPTIONAL TOPIC: TRAPEZOID AND RUNGE-KUTTA ALGORITHMS 57

But wait! In order for this to be useful, we need to be able to compute the right side solely in terms of y_* and Δt . Thus we insert another approximation, the Euler approximation $y(t_* + \Delta t) \approx y_* + f(y_*)\Delta t$, in to the right side in order to obtain

$$y(t_* + \Delta t) \approx y_* + \frac{1}{2} \left(f(y_*) + f(y_* + f(y_*)\Delta t) \right) \Delta t. \quad (1.10.4)$$

This gives us the *trapezoidal approximation scheme*

$$y_{k+1} = y_k + \frac{1}{2} \left(f(y_k) + f(y_k + f(y_k)\Delta t) \right) \Delta t. \quad (1.10.5)$$

In practice, it is common to compute the quantities

$$K_0 = f(y_k) \quad \text{and} \quad K_1 = f(y_k + f(y_k)\Delta t)$$

so that (1.10.5) becomes

$$y_{k+1} = y_k + \frac{1}{2} \left(K_0 + K_1 \right) \Delta t. \quad (1.10.6)$$

Example 1.10.1. *Here is some code that uses the trapezoidal approximation scheme to approximate the solution the initial value problem*

$$\frac{dy}{dt} = y^2 \quad y(0) = 1 \quad (1.10.7)$$

with $\Delta t = 0.5$; see Figure 1.10.1.

```

var('y','t')
f(y) = y^2

y0 = 1
deltaT = 0.1
steps=5

trapData = [[k*deltaT,y0] for k in range(0,steps+1)]

for k in [1..steps]:
    K0 = f(trapData[k-1][1])
    K1 = f(trapData[k-1][1] + (1/2)*deltaT*(K0 + K1))
    trapData[k][1]= trapData[k-1][1] + (1/2)*deltaT*(K0 + K1)

realPlot = plot(1/(1-t),(t,0,.5))
trapPlot = list_plot(trapData, color="red", plotjoined=true,
    marker='.', axes_labels=['t','$y$'])
mainPlot = realPlot + trapPlot
mainPlot.show(figsize=[4,3])

```

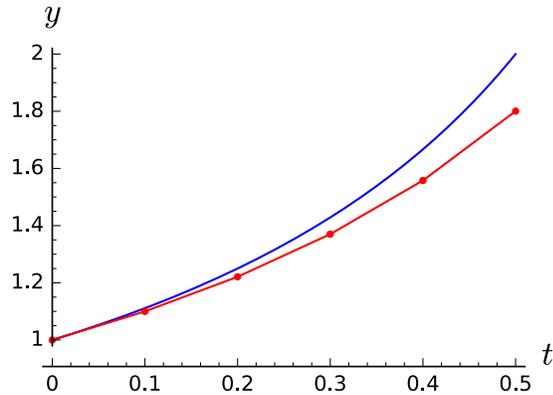


Figure 1.10.1: The trapezoidal approximation of the solution to the initial value problem (1.10.7) is in red; the exact solution is in blue.

While the previous example illustrates that the trapezoidal approximation scheme is a reasonable one, it is not immediately obvious that it is better than the Euler approximation scheme from the previous section. To see that the trapezoidal scheme is indeed better, we need to trace through the size of the errors involved in (1.10.4). Here is a rough accounting. From the mean value theorem we have

$$f(y_2) = f(y_1) + O(\Delta y)$$

and from Taylor's theorem, together with the ODE, we have

$$\begin{aligned} y(t_* + \Delta t) &= y_* + y'(t_*)\Delta t + O(\Delta t^2) \\ &= y_* + f(y_*)\Delta t + O(\Delta t^2). \end{aligned}$$

Thus

$$f(y(t_* + \Delta t)) = f(y_* + f(y_*)\Delta t) + O(\Delta t^2).$$

Since the trapezoidal rule for the integral (1.10.2) has error $O(|b - a|^2)$, we see that the error in the trapezoidal approximation scheme is $O(\Delta t^2)$. This is better than the Euler method, which has error $O(\Delta t)$.

We now briefly present another, more sophisticated scheme called the **Runge-Kutta approximation scheme**. The main idea behind this scheme is to choose an intermediate time between t_* and $t_* + \Delta t$ at which to approximate the function y . For convenience, we choose the intermediate point to be the midpoint $t_* + \Delta t/2$. The Runge-Kutta scheme is as follows. Given the value y_k , compute four different approximations of $f(y)$:

1.10. OPTIONAL TOPIC: TRAPEZOID AND RUNGE-KUTTA ALGORITHMS 59

- Left endpoint: $K_0 = f(y_k)$
- Midpoint, Euler: $K_1 = f(y_k + \frac{\Delta t}{2}K_0)$
- Midpoint, Trapezoid: $K_2 = f(y_k + \frac{\Delta t}{2}K_1)$
- Right endpoint, using midpoint: $K_3 = f(y_k + \Delta t K_2)$

We then use the numerical scheme

$$y_{k+1} = y_k + \left(\frac{1}{6}K_0 + \frac{1}{3}K_1 + \frac{1}{3}K_2 + \frac{1}{6}K_3 \right) \Delta t. \quad (1.10.8)$$

It turns out that the error in (1.10.8) is $O(\Delta t^4)$, which makes it much more efficient for long-time computations.

Exercise 1.10.1. *In this problem we study the differential equation $\frac{dy}{dt} = y - y^2$, which is a simplified version of the logistic growth model.*

1. *This differential equation has two equilibrium solutions. Determine what they are and, after sketching the slope field, determine their stability.*
2. *Choose two numbers y_{stable} and $y_{not\ stable}$ such that when we consider initial value problem*

$$\frac{dy}{dt} = y - y^2 \quad y(0) = y_0$$

we expect the solution $y(t)$ to tend towards the stable equilibrium for $y_0 = y_{stable}$ and the solution $y(t)$ to diverge for $y_0 = y_{not\ stable}$.

3. *Run an Euler method numerical simulation with both initial conditions, generating a plot of the approximate solution. Do the solutions behave the way which you expected them to?*
4. *Implement the trapezoid numerical scheme for this equation, and construct approximate solutions for both of your initial conditions. Plot these approximate solutions on the same axes as your Euler solutions. Can you see any improvement?*
5. *Implement the Runge-Kutta numerical scheme for this equation, and construct approximate solutions for both of your initial conditions. Plot these approximate solutions on the same axes as your Euler and trapezoid solutions. Can you see any improvement?*

Exercise 1.10.2. *In this problem we study the initial value problem*

$$\frac{dy}{dt} = y^2 \quad y(0) = 1$$

1. *Verify that $y(t) = (1 - t)^2$ is the unique solution to this initial value problem. On what time domain does this solution exist? Make a plot of the solution on this domain. What happens at the “boundaries” (both left and right) of the domain?*
2. *Construct an approximate (numerical) solution using the Euler method on as much of the domain as is feasible. Does the numerical approximation share the same behavior as the actual solution near the boundaries of the domain? Plot the approximate solution on the same axes as the actual solution.*
3. *Repeat the previous part for both the trapezoid and Runge-Kutta methods. Do you see any improvement? Comment.*