Numerical Analysis
MS-C1650
Tölle/Nyman
Exercises (MATLAB), Week 21, 2022

DATE[1]

On assignments: Submit homework to your assistant electronically via the course pages. MATLAB-assignments are submitted via Peergrade.

1 Euler's Method

For $m = 1$ let us consider the problem

$$\begin{cases} y'(t) = -150y(t) + 49 - 150t, & t \in [0,1], \\ y(0) = 1/3 + \epsilon, \end{cases} \tag{1}$$

where $\epsilon \in \mathbb{R}$ is the error in the initial data.
EXERCISE 1

(a) Find the analytic solution y_ϵ.

(b) Show that $\|y_0 - y_\epsilon\|_\infty \le |\epsilon|$.

(c) Let $h > 0$. If $t, t+h \in [0,1]$, show that

$$y_0(t+h) = y_0(t) + h(-150y_0(t) + 49 - 150t).$$

(d) Let $n \in \mathbb{N}$ with $n > 0$, $h = 1/n$, and $t_i = (i-1)h$, $i = 1, \dots, n+1$. Compute the discrete solution $u_{\epsilon,i}$ for $i = 1, \dots, n+1$ using Euler's Method.

(e) Show that for $i = 1, \dots, n$,

$$u_{\epsilon,i+1} - y_0(t_{i+1}) = (1 - 150h)(u_{\epsilon,i} - y_0(t_i))$$

and

$$u_{\epsilon,i} - y_0(t_i) = (1 - 150h)^{i-1}\epsilon$$

for $i = 1, \dots, n+1$.

[1] Published on 2022-05-19

(f) If $n = 50$ and $\epsilon = 0.01$, compute the error $u_{\epsilon,n+1} - y_0(1)$ at $t = 1$.

(g) Give a condition on n to obtain

$$\max_{i=1,\ldots,n+1} |u_{\epsilon,i} - y_0(t_i)| \leq \epsilon.$$

EXERCISE 2

(a) Show how $u_{\epsilon,i+1}$ for $1 \leq i \leq n$ can be computed from $u_{\epsilon,i}$ using the backward Euler method.

(b) Show that for $i = 1, \ldots, n$

$$u_{\epsilon,i} - y_0(t_i) = \frac{1}{(1+150h)^{i-1}}\epsilon.$$

(c) Give a condition on n to obtain

$$\max_{i=1,\ldots,n+1} |u_{\epsilon,i} - y_0(t_i)| \leq \epsilon.$$

2 Heun's Method

Let us consider the same IVP assume that the RHS from above is implemented in `f1.m`.
EXERCISE 3

(a) Write a programme `Heun.m` that implements the Heun's method on an uniform partition.

(b) Write a program that plots the graphs of the approximation and of the exact solution and computes the error: $\max_{i=1,\ldots,n+1} |u_i - y(t_i)|$, where $y(t)$ is the exact solution. Test with $n = 40, 73, 75, \ldots$ and $\epsilon = 0.01$.

(c) Write a program `Heunerror.m` to study the error for different values of n. Beginning with an array `arrn` the program will compute the corresponding `arrerror` and plot $\ln(\text{arrerror})$ depending on $\ln(\text{arrn})$. What seems to be the order of the scheme?

Test with `arrn=5000:10:5100` and $\epsilon = 0.01$.

Without ϵ you should get something along these lines:

```
>> [t,u] = heun( 'f1', [0,1], 1/3, 4)

t =

         0    0.2500    0.5000    0.7500    1.0000


u =

    0.3333    0.0833   -0.1667   -0.4167   -0.6667
```

Remember that the results will change dramatically as you add ϵ to the initial value!
One outline for the graph illustration is:

```
arrn=5000:10:5100;
epsilon=0.01;

for i=1:length(arrn)
n=arrn(i);
[t,u]=heun(...);
y=...;
arrerror(i)=norm(u-y,inf);
end
plot(log(arrn),log(arrerror),'.--')
a=polyfit ( log ( arrn ) , log ( arrerror ) ,1);
title ([ 'Slope of the regression line : ', num2str(a(1))])
xlabel('log(n)')
ylabel('log(error)')
```

I'd expect the slope to be close to 2 (in absolute value).

3 MATLAB

We use the same f1.m. EXERCISE 4 Test the MATLAB tool ode23 with the initial data $u_1 = 1/3 + \epsilon$, and let $\epsilon = 0.1$ and 0.001.
Test with [t,u] = ode23('f1',[0,1],1/3 + epsilon). Plot the vector Δt, with $\Delta t_i = t_{i+1} - t_i$ in the two cases.

For instance:

```
>> [ t, u]=ode23( 'f1',[0,1],1/3); length(t)

ans =

    39
```