Homework 3

Aasim Zahoor

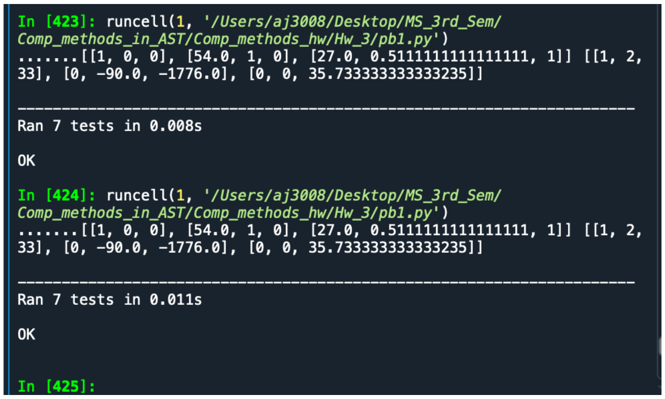
September 24, 2020

1 Problems

Link `https://github.com/AasimZahoor/Comp_methods.git`

Problem 1

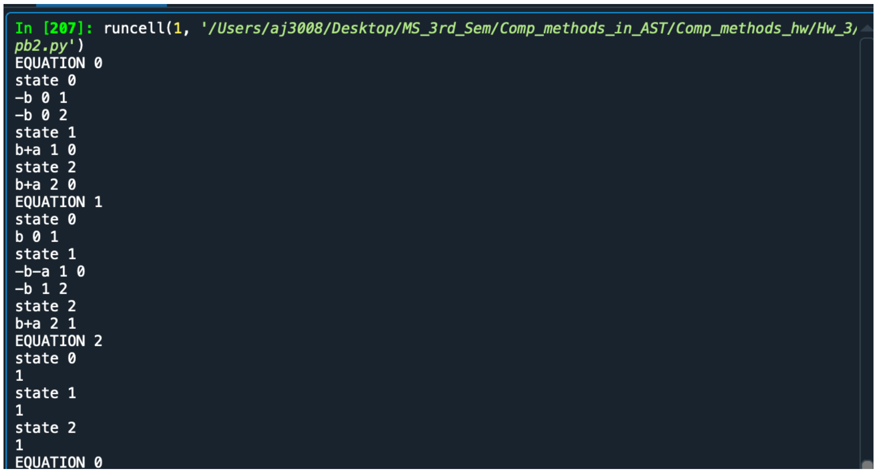
In this problem I use two random matrices to test my matrix class. I used numpy's functionality to test some of the instances of the class. There were 7 instances in my class and I tested all of them. So I tested addition, multiplication, transpose, trace, Determinant, Inverse and LU decomposition. The output is:



```
In [423]: runcell(1, '/Users/aj3008/Desktop/MS_3rd_Sem/
Comp_methods_in_AST/Comp_methods_hw/Hw_3/pb1.py')
.......[[1, 0, 0], [54.0, 1, 0], [27.0, 0.5111111111111111, 1]] [[1, 2,
33], [0, -90.0, -1776.0], [0, 0, 35.733333333333235]]

----------------------------------------------------------------------
Ran 7 tests in 0.008s

OK

In [424]: runcell(1, '/Users/aj3008/Desktop/MS_3rd_Sem/
Comp_methods_in_AST/Comp_methods_hw/Hw_3/pb1.py')
.......[[1, 0, 0], [54.0, 1, 0], [27.0, 0.5111111111111111, 1]] [[1, 2,
33], [0, -90.0, -1776.0], [0, 0, 35.733333333333235]]

----------------------------------------------------------------------
Ran 7 tests in 0.011s

OK


In [425]:
```

Problem 2

This problem asks us to plot number densities at different temperatures. I took 8 temperature values. They are $T = [273, 1000, 4000, 8000, 10000, 30000, 50000, 60000]$. For this problem I first found the pattern and write an algorithm for it. Here is a test for 3 state system. Note the coefficients are such that the RHS of equation formed is zero, meaning I took the LHS in the equations you gave to RHS.



```
In [207]: runcell(1, '/Users/aj3008/Desktop/MS_3rd_Sem/Comp_methods_in_AST/Comp_methods_hw/Hw_3/
pb2.py')
EQUATION 0
state 0
-b 0 1
-b 0 2
state 1
b+a 1 0
state 2
b+a 2 0
EQUATION 1
state 0
b 0 1
state 1
-b-a 1 0
-b 1 2
state 2
b+a 2 1
EQUATION 2
state 0
1
state 1
1
state 2
1
EQUATION 0
```

Figure: Testing the alogorithm

After finding the pattern I wrote down function for Bul, Aul, Blu and Energy. Then I extracted the data and plotted the number densities V/s Temperature.

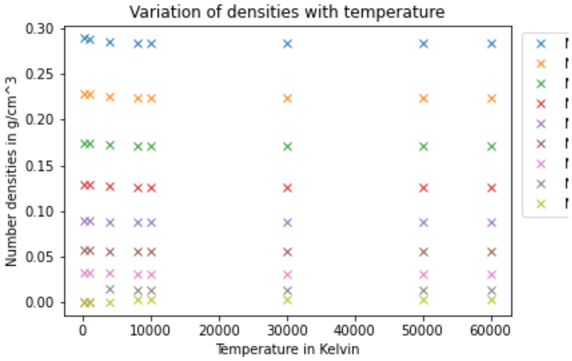


Figure: Plot of number densities v/s temperature

Problem 3

In the code for Problem 3 I have defined 3 functions. They are:

- **euler**(x_0, y_0, f, h, k) This function call euler method for solving ODE. The arguments are:
  x_0=initial x value y_0=initial y value f=function of x,y,t h=step size k=number of times
  It returns x,y,t values in that order

- **heun**$(x_0, y_0, f, h, k, p = 10)$

  This function call Heun's method for solving ODE. The arguments are: x_0=initial x value, y_0=initial y value, f=function of x,y,t, h=step size, k=number of times, p=picard's iteration (Default value 10)
  It returns x,y,t values in that order

- **rk**(x_0, y_0, f, h, k)

  This function call RK-4 method for solving ODE. The arguments are: x_0=initial x value, y_0=initial y value, f=function of x,y,t, h=step size, k=number of times
  It returns x,y,t values in that order

Problem 4

In this problem we are supposed to test the functions we wrote in problem 3 using the pendulum function. To model the function I have made a function called **f(b,c)**. This function takes two arguments, b= coefficient of y, c= coefficient of g(x,t). This function returns an array where the first element gives d(theta)/dt and the second element returns d(omega)/dt. They values of b is taken as 0.25 and c as 5, step size is 0.1 and k (number of times) is 50 (to see the difference between the three methods) and 200 (to see how damping works).

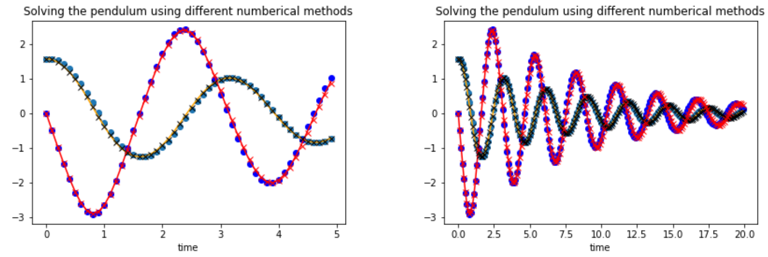


Figure: The figure on the left is for k=50 and one on the right is for k=200

Problem 5

In this problem we are supposed to test our methods on the given equation. I defined a function **f(l)** which returns the function given in pb5. It takes lambda as input and gives an array as out where first element is always 1 and second element gives dy/dt in terms of y and t. The reason it gives 1 as the first element is because my ODE solvers were made for 2 dependent variables and this has only 1.

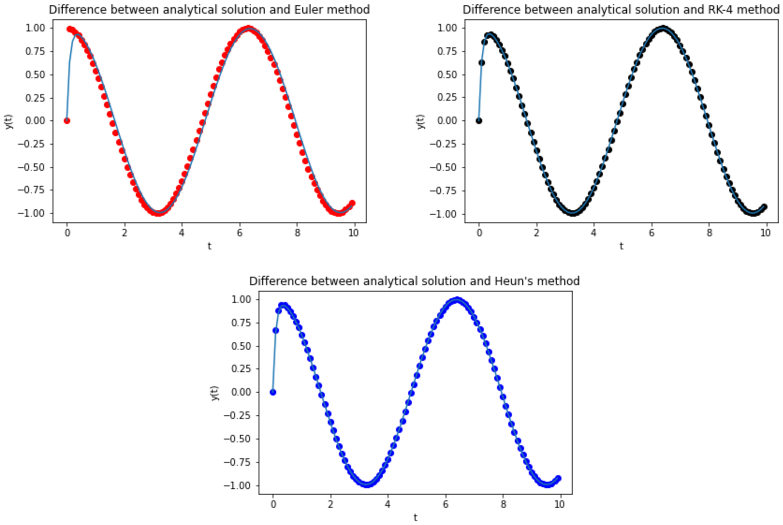


Figure: These figures shows the three methods against the analytical solution.