ode_solve_test

June 7, 2021

[1]:
```
#First define the function f(t,u) in the ODE u' = f(t,u). Here u is treated
#as a vector (u[0],u[1],...) so for a scalar ODE use u[0].
def f(t,u):
    return [t+u[0]]
```

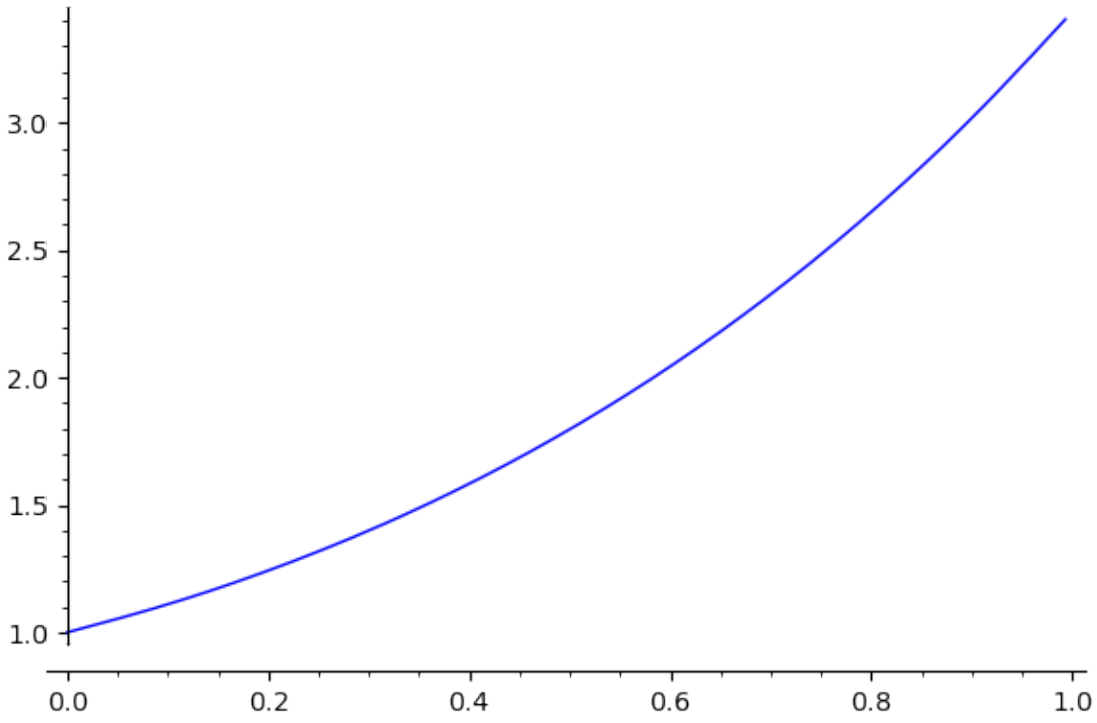
[2]:
```
T = ode_solver() #Set up data structure "T" for handling solution process
```

[3]:
```
T.function = f #Define the right side of the ODE
```

[4]:
```
T.ode_solve(y_0=[1],t_span=[0,1],num_points=10) #Call the solver with initial␣
↪condition u(0)=1 (must label "y0")
```

[5]:
```
usol = T.interpolate_solution() #Set up a function "usol" to interpolate the␣
↪solution
```

[6]:
```
plot(usol,0,1).show() #Plot the solution
```



[7]:
```
usol(0.5) #Evaluate the solution at a certain time
```

[7]:
```
1.7974425413610653
```

[ ]:
```

```