Numerical Method

National Cheng Kung University Department of Engineering Science

Instructor: Chi-Hua Yu

Lab 8

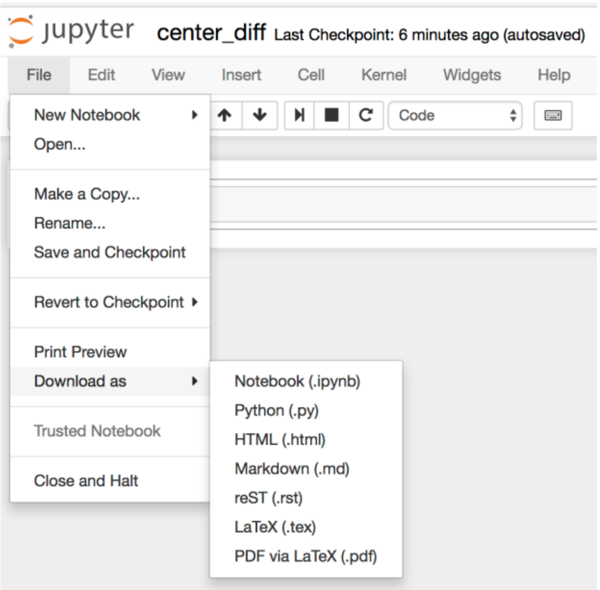
Programming, Due 10:00, Wednesday, April 27th, 2022

注意事項：

1. Lab 的時間為授課結束(Lab 當天 10:00)。
2. Lab 的分數分配：出席 20%，Lab 分數 100%，Bonus 20%。
3. 請儘量於 Lab 時段完成練習，完成後請找助教檢查，經助教檢查後沒問題者請用你的學號與 Lab number 做一個檔案夾 (e.g., N96091350_Lab8, 將你的全部 ipynb 檔放入檔案夾，壓縮後上傳至課程網站 (e.g., N96091350_Lab8.zip)。
4. 上傳後即可離開。
5. 未完成者可於隔日 11:55 pm 前上傳至 Moodle，惟補交的分數將乘以 0.8 計，超過期限後不予補交。
6. Bouns 只需要在每週四的 11:55 pm 上傳即可。

<u>Lab Submission Procedure (請仔細閱讀)</u>

1. You should submit your Jupyter notebook and Python script (`*.py`, in Jupyter, click File, Download as, Python (`*.py`)).



2. Name a folder using your student id and lab number (e.g., n96081494_lab1), put all the python scripts into the folder and zip the folder (e.g., n96081494_lab1.zip).

3. Submit your lab directly through the course website.

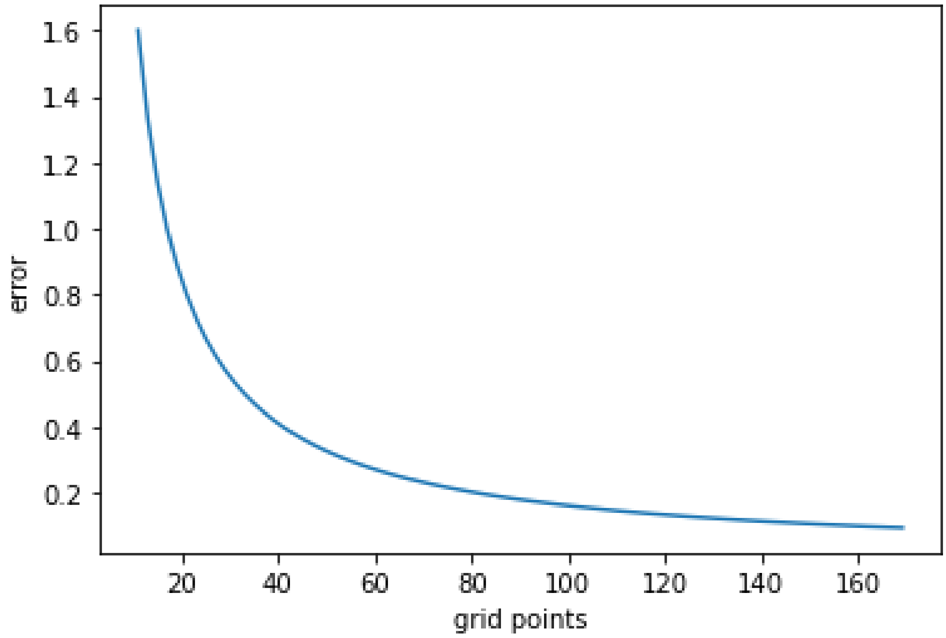
1. (50%) Name your Jupyter notebook `Simpson_rule.ipynb` and Python script `Simpson_rule.py`. Use Simpson's rule to approximate $\int_{-2}^{4} x^3 + x^2 dx$ and compare this value to the exact value of 84. Compare the error of the approximation with the exact value using grid points in the range 11 to 171 according to **Simpson's rule**. Please print the minimum error and plot the error of grid points

```
def function(x):
    f = (x**3) + (x**2)
    return f

a = -2
b = 4
h = (b - a) / (n - 1)
x = np.linspace(a, b, n)
f = function(x)
```

Below is the running example:

```
minimum error: 0.09523809523813043
```



2. (50%) Name your Jupyter notebook `Euler_method.ipynb` and Python script `Euler_method.py`. The differential equation $\frac{df(t)}{dt} = e^{-2t}$ with initial condition $f_0 = -\frac{1}{2}$ has the exact solution $f(t) = -\frac{1}{2}e^{-2t}$. Approximate the solution to this initial value problem between zero and 1 in increments of 0.1 using the explicit Euler formula. Please plot the difference between the approximated solution and the exact solution.

Below is the running example: