Python Programming for Engineers

Assignment # 4

Due date: Monday, Jan. 4th, 2021

IMPORTANT!

1. Submit your HWs **ONLINE** before the due date
2. HW reports should contain:
    a. The description of the problem and proposed solution
    b. The program code
    c. Any program outputs
3. Submitted codes should be well-commented.

Problem 1:

Design a Python class named **ImageFilt** for processing a digital image. The class should contain:

- The data fields for **filename**, image array (**im**), **height** and **width** of the image.
- A constructor with the argument for **filename**.
- A method named **load_im()** that loads the image, sets height and width and returns true if the file exists; else returns false and prints a warning that says "Cannot find the image file".
- The method **filter_im(f)** that filters the image using a blur filter if f=0, or using a sharpening filter if f =1.
- The method **plot_im()** that plots the original and filtered images as subplots in a single figure window.

Write a test program that prompts the user to enter the **filename** (i.e. "lena.pgm") and displays the original image and its blurred and sharpened versions. Test your program for two different colored images.

Problem 2:

Using the function *f(t)*:

$$f(t) = 3e^{-3t}\cos(\frac{2\pi}{3}t) - 5te^{-t} - 2e^{-t}$$

Perform the following and provide the resulting plots:

a. Evaluate *f(t)* for $0 \le t \le 10$ - with steps of 0.001.

b. Plot *f(t) vs. t.*

c. Plot *g(t) vs. t* **(using a single Python command),** where

$$g(t) = \begin{cases} f(t), & 0 \le t \le 4 \\ -f(t), & \text{else} \end{cases}$$

d. Filter *f(t)* using the **median filter** and plot the filtered and original functions together on the same plot.

Problem 3:
Load the data from "TimeSeries.xlsx" into a numpy array, say *numdat*, and perform the following analysis:

a. Assign first 150 samples of the "*numdat*" into a new vector called "*numdat_1*".
b. Assign every $(3k+1)^{st}$ $\left(k \in 0, Z^{+}\right)$ sample into a new vector called "*numdat_2*". (*numdat_2* provides only the values of samples no: *1,4,7,...,148*).

TASK 1: Linear Interpolation

c. By using **linear** and **cubic spline interpolation** methods of *interpolate.interp1d* method of SciPy package, estimate the missing sample values of *numdat_2* and compare predicted time-series with original time-series (i.e. compute the mean squared error (MSE) between true samples and estimated samples).

TASK 2: Polynomial Regression

d. By using polymonial regression methods *polyfit* and *polyval* of NumPy package, represent the characteristics of the *numdat_2* time-series with a polynomial.

   Use **fourth** and **fifth** order polynomial regression to estimate the sample values of *numdat_2* and compare predicted time-series with original time-series (i.e. compute the mean squared error (MSE) between true samples and estimated samples).

(Hint: First model the given samples of *numdat_2* by using *polyfit* method. Once you obtain a polynomial which models the data, evaluate the value of the polynomial for the all samples of *numdat_1* using *polyval* method.)

e. Compare the estimation results of TASK 1 and TASK 2. Plot original time-series *numdat_1* and the estimated results of TASK 1 and TASK 2. Discuss which method performs better and why.