Homework 10

Ch En 263 – Numerical Tools

Due: 27 Mar. 2023

Instructions

- Complete the problems below and submit the following files to Learning Suite:
  - Handwritten portion: scan each page (or take a picture) and combine them into a single pdf named: `LastName_FirstName_HW10.pdf`
  - Excel portion: submit a workbook named `LastName_FirstName_HW10.xlsx` where each worksheet tab is named "Problem_1", "Problem_2", etc.
  - Python portion: submit a separate file for each problem named `LastName_FirstName_HW10_ProblemXX.py` where XX is the problem number.

Problems

1. The kinetic rates r in the table on the right were collected as a function of temperature, T. We want to fit the model

$$r = kT^m \exp(\frac{-E_a}{RT})$$

to this data where k, m and (E_a/R) are adjustable parameters.

T	r
500.0	105.598
571.43	89.700
642.86	70.768
714.29	66.996
785.71	60.711
857.14	58.992
928.57	55.8328
1000.0	53.420

   (a) Use Solver in Excel to find the best-fit coefficients k, m and E_a/R by minimizing the sum-of-squared error (SSE).

   (b) Use `scipy.optimize.curve_fit` in Python to find the best-fit coefficients and print them to the console. In addition, calculate the value of R^2 in Python and print it to the console.

   (c) Plot the data and the model with the best-fit parameters from part (b) together on the same graph in Python.

2. The data in `HW10_P2_Data.csv` contains data for the response of a "first-order system with time delay". Such data is used to create control devices to keep operations running correctly. (You will learn about this in ChEn 436, Process Control and Dynamics.)

   The model for a first-order system with time delay is given by

$$y(t) = 5\left[1 - \exp\left(\frac{-(t-\theta)}{\tau}\right)\right] S(t-\theta)$$

   where

$$S(t-\theta) = \begin{cases} 0, & \text{when } t < \theta \\ 1, & \text{when } t \geq \theta \end{cases}$$

   where τ is a characteristic time of the process and θ is the time delay.

(a) Use *either Python or Excel* to find the constants, τ and θ, that best fit the model to the process data.

(b) Determine the R^2-value of the fit and report it.

(c) Make a plot showing the data (as points) and the fit (as a line).

Hint: You will need to use an `if` statement for this problem.

3. In this problem we are going to explore how noisy data can affect an interpolation. The data set `HW10_P3_Data.csv` contains three sets of x and y pairs based on an underlying function

$$y = \exp(-2x)\sin(4\pi x). \tag{1}$$

One of the data sets has no noise, one has a small amount of noise and one has a large amount of noise.

(a) Use Python functions to generate a cubic spline for each data set (no noise, small noise, large noise). Using the original function in Eq. 1, calculate the relative error of your interpolation at $x = 1.37$ for each data set. (See Lab 18 for the definition of the relative error.) Print the values of the relative error to the console.

(b) Make three separate plots—one for each data set—and compare your cubic spline to the original function in Eq. 1. Each plot should be formatted using:

- A solid line for the original function.
- Points for the data.
- A dashed line for the cubic spline.
- A legend labelling the interpolating function as "no noise", "small noise" or "large noise".

(c) What happens to the interpolation as the data gets noisier? Is the interpolation better or worse for noisy data when the y is close to zero? Speculate about why. Print your comments as a string to the console.