Due: 23:59, November 10, 2024

ME 278: A Practical Introduction to Data Analysis
Assignment 4

22-11-2024

Honor Code

Each student should solve problems in this assignment independently. Specifically, *code sharing in any form is strictly prohibited*. However, you are permitted to discuss aspects of code syntax/theory/algorithms/implementation with your peers. You can list names of your colleagues (below) with whom you collaborated in solving this assignment.

Collaborators

1.
2.
3.
4.

Students found plagiarising will lose all credit for this assignment.

Submission Instructions

Submit the code for the assignment in .zip file under the name 'SRNo_firstname_lastname.zip'. The zip file must contain the code for all the problems along with all the necessary files to run the code. Also, include in the .zip file, a .pdf file containing all results, comments and explanations requested in the assignment.

1 Problem 1 - Sampling and Differentiation

Consider the function $f(x) = sin(x) + cos(x)$.

- Use non-uniform sampling (use random generator by specifying seed $rng(4)$ and sorting in ascending order for x) to generate 1000 samples over the interval $[0, 2\pi]$.
- Approximate the derivative of $f(x)$ using the following methods.
  1. Forward difference
  2. Backward difference
  3. Three point, two-sided difference formula
- Plot and compare the numerical approximations with the analytical derivative and quantify the error. (Exclude the derivatives at first and last points for the comparison)

2 Numerical Integration

2.1 Lorenz system

Consider the following system dynamics

$$\frac{dx}{dt} = \sigma(y - x) \tag{1}$$

$$\frac{dy}{dt} = x(\rho - z) - y \tag{2}$$

$$\frac{dz}{dt} = xy - \beta z \tag{3}$$

- Implement a numerical integration scheme through MATLAB/Python that uses the implicit Crank-Nicolson method for the linear terms and the explicit Adams-Bashforth method (AB2) for the nonlinear terms.
- Solve the Lorenz system using this scheme for a given set of initial conditions $[x, y, z]|_{t=0} = [0.9, 0, 0]$ and parameters $(\rho = 28, \sigma = 10, \beta = 8/3)$ and plot the state-space solution (x, y, z) in 3D.
- Compare the results obtained using the Crank-Nicolson-Adams-Bashforth scheme with those obtained using the built-in MATLAB function ode45.

3 Fourier Analysis

The discrete Fourier transform (DFT) of an N point signal is defined as

$$F(\omega) = \sum_{k=0}^{N-1} f_k e^{-i\omega x_k} \tag{4}$$

3.1 Matrix product representation of DFT

1. What are the values of ω permitted in the discrete spectrum?
2. Re-formulate the above equation solely in terms of N and the indices k, j for x, ω respectively.
3. Represent the DFT as a matrix product: $\mathbf{F} = \mathbf{Mf}$, where $\mathbf{f} = \begin{bmatrix} f_0, & f_1 & \cdots & f_{N-1}, & f_N \end{bmatrix}^\top$ is a column vector representation of the N point signal. Note: Your code should take N as the only input argument, and the matrix M should be computed.

3.2 DFT vs FFT

A time series data for a signal is provided at $sample\_Q3.mat$. Using the given uniformly sampled data,

- Compute DFT matrix and plot the magnitude spectrum.
- Compare the result with fft function from MATLAB/Python
- Obtain an analytical approximation of the form $\sum_i a_i sin(w_i t)$ (where $\omega_i = 2\pi f_i$) by obtaining suitable dominant frequencies. (Round the dominant frequencies to nearest integers.)
- Using the above $obtained$ expression, sample N point signal over $[0, 2\pi)$ for N=512, 1024, 2048 and compare the computational times of DFT and FFT. How does computational time scale with N in each case? Will there be any difference in this observation if we sample N=500 points? Justify.

3.3 Aliasing and spectral leakage

Consider the following signal defined over the interval [0, 1)

$$\begin{aligned} x_1(t) &= \sin(144\pi t) \\ x_2(t) &= \sin(144.4\pi t) \end{aligned} \tag{5}$$

- Create the discrete signal $x_1(t)$ using N = 256 points; compute the Nyquist frequencies f_N and the FFT. Plot the Fourier spectrum versus frequency and confirm the peak is at the correct location.
- Create the signals $x_1^2(t)$ and $x_1^3(t)$ using the same number of points N and recompute the FFTs. At what frequencies do you observe the peaks in each case?
- Use trigonometric identities and explain the results.
- Discretize $x_2(t)$ with N=256 and compute FFT.
- The peaks in the Fourier spectrum of this signal aren't as sharp as $x_1(t)$. Using a curve fitting tool, fit a Gaussian of form

  $$e^{-(\frac{f-f_0}{\sigma_0})^2} \tag{6}$$

  to points close to one of the FFT peaks and identify the mean frequency f_0. Does it match with that of the input signal $x(t)$?

4 Noise filtering and Convolution

Consider the following signal defined over the interval $[0, 2\pi)$

$$x(t) = \sin(4t) + 2\sin(11t)\cos(7t) + 4\cos(t) + \epsilon(t) \tag{7}$$

where $\epsilon(t)$ represents noise.

- *Case: 1* $\epsilon(t)$ follows Gaussian with $(\mu, \sigma^2) = (0, 1)$.
- *Case: 2* $\epsilon(t)$ is sampled from Uniform distribution in the interval $(-0.4, 0.4)$.

4.1 Filters

Use the following noise filtering techniques

- Moving Average filter
- Savitzky-Golay filter
- Median filter

Tweak the parameters of each filter to obtain the best performance.

1. Quantify the performance of each filtering technique by using comparison plots and RMSE error from the noiseless signal.
2. Justify the performance of each technique for different cases of noise mentioned above.

4.2 Convolution

Given a noisy image $image\_with\_noise.png$, identify 2D Sharpening and Gaussian kernels (5×5).

1. Apply the above gaussian kernel to $image\_with\_noise.png$.
2. Subsequently, apply the Sharpening kernel and comment on the effects of each kernel.
3. Tweak the parameters to obtain the best possible result. Compare and comment using peak signal-to-noise ratio (PSNR) and mean squared error (MSE) with the $original\_image.png$