

# Lab 1

---

Rajdeep Gill 7934493

ECE 4830 B03

January 21, 2025

## Contents

1	Problem 1	3
2	Problem 2	4
3	Problem 3	4

## 1 Problem 1

- The provided matlab code generates a signal of 181 samples. The signal consists of a line, unit pulse, sine wave and a random segment. Once the main signal is created, a second signal is created by adding some noise to the main signal. The noise is generated with the `normrnd` function. The two signals, main and noise, are then plotted on the same graph.
- Using ChatGPT to describe the code we get the following response:

This MATLAB script generates a synthetic signal composed of linear, constant, sinusoidal, and noisy segments to simulate a real-world signal. It introduces anomalies, such as spikes and plateaus, and adds Gaussian noise to mimic measurement imperfections. The original and noisy signals are plotted side by side, showcasing the impact of noise and modifications.

- Asking an LLM to convert the matlab code to python we get the following response:

```

1  import numpy as np
2  import matplotlib.pyplot as plt
3
4  # Initialize signal
5  sig = np.zeros(181)
6
7  # Signal construction
8  sig[:51] = np.arange(0, 51)           # Linearly increasing section
9  sig[51:100] = 50                     # Constant value section
10 sig[100:125] = 100                    # Higher plateau
11 sig[125:130] = 0                     # Zero section
12 sig[130:181] = 50 + 5 * np.random.randn(50) # Random noisy section
13
14 # Append sinusoidal signals
15 n = np.arange(0, 51)
16 s1 = 50 + 5 * np.sin(2 * np.pi * 0.1 * n) # Sine wave 1
17 s2 = 50 + 5 * np.sin(2 * np.pi * 0.2 * n) # Sine wave 2
18 sig = np.concatenate([sig, s1, s2])
19
20 # Add baseline offset and anomalies
21 sig += 50
22 sig[54:57] = 150 # Spike
23 sig[58:62] = 125 # Small plateau
24
25 # Add noise
26 xx = np.arange(1, 284)
27 noisy_sig = sig + np.random.normal(0, 5, size=sig.shape)
28
29 # Plot the signals
30 plt.figure(figsize=(10, 5))
31 plt.plot(xx, noisy_sig.astype(np.uint8), '--', label='Noisy', linewidth=1.5)
32 plt.plot(xx, sig.astype(np.uint8), label='Original', linewidth=2)
33 plt.axis([0, 283, 30, 170])
34 plt.legend()
35 plt.xlabel('Index')
36 plt.ylabel('Signal Value')
37 plt.title('Original and Noisy Signal')
38 plt.grid(True)

```

```
39 plt.show()
```

Code Snippet 1: Python code generated by LLM

- d. To highlight the transitory parts, we can find the standard deviation of the 30 signals at each sample and plot the results. This will show the parts of the signal that are changing the most.

## 2 Problem 2

## 3 Problem 3

3.1-6 We have:

$$x[n] = \begin{cases} \left(\frac{1}{3}\right)^n & \text{if } n \geq 0 \\ A^n & \text{if } n < 0 \end{cases}$$

- a. Determine the energy  $E_x$  and power  $P_x$  of  $x[n]$  if  $A = 1/2$ .

The energy can be found as follows:

$$\begin{aligned} E_x &= \sum_{n=-\infty}^{\infty} |x[n]|^2 \\ &= \sum_{n=-\infty}^{-1} \left(\frac{1}{2}\right)^{2n} + \sum_{n=0}^{\infty} \left(\frac{1}{3}\right)^{2n} \\ &= \sum_{n=-\infty}^{-1} \left(\frac{1}{4}\right)^n + \sum_{n=0}^{\infty} \left(\frac{1}{9}\right)^n \\ &= \frac{1}{1 - \frac{1}{4}} + \frac{1}{1 - \frac{1}{9}} \\ &= \frac{4}{3} + \frac{9}{8} \\ &= \frac{59}{24} \end{aligned}$$

The power can be found as follows:

$$\begin{aligned} P_x &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N |x[n]|^2 \\ &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=-N}^{-1} \left(\frac{1}{2}\right)^{2n} + \sum_{n=0}^N \left(\frac{1}{3}\right)^{2n} \right) \\ &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=-N}^{-1} \left(\frac{1}{4}\right)^n + \sum_{n=0}^N \left(\frac{1}{9}\right)^n \right) \\ &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{m=1}^N 4^m + \sum_{n=0}^N \left(\frac{1}{9}\right)^n \right) \end{aligned}$$

We can use the following two formulas to find the sum of the two series:

$$\sum_{m=1}^N 4^m = 4 + 4^2 + 4^3 + \dots + 4^N = \frac{4(4^N - 1)}{4 - 1} = \frac{4^{N+1} - 4}{3}$$

$$\sum_{n=0}^N \frac{1}{9^n} = \frac{1 - \frac{1}{9^{N+1}}}{1 - \frac{1}{9}} = \frac{9}{8} \left( 1 - \frac{1}{9^{N+1}} \right)$$

We now have the following:

$$P_x = \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \frac{4^{N+1} - 4}{3} + \frac{9}{8} \left( 1 - \frac{1}{9^{N+1}} \right) \right)$$

Now as  $N \rightarrow \infty$ , the term  $\frac{1}{9^{N+1}}$  goes to 0, and the  $4^{N+1}$  term grows very large. This means that the power of the signal is infinite, that is:

$$P_x \rightarrow \infty$$

- b. Determine the energy  $E_x$  and power  $P_x$  of  $x[n]$  if  $A = 1$ .

The energy can be found as follows:

$$\begin{aligned} E_x &= \sum_{n=-\infty}^{\infty} |x[n]|^2 \\ &= \sum_{n=-\infty}^{-1} 1 + \sum_{n=0}^{\infty} \left( \frac{1}{3} \right)^{2n} \\ &= \infty \end{aligned}$$

The power can be found as follows:

$$\begin{aligned} P_x &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N |x[n]|^2 \\ &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=-N}^{-1} 1 + \sum_{n=0}^N \left( \frac{1}{3} \right)^{2n} \right) \\ &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=-N}^{-1} 1 \right) + \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=0}^N \left( \frac{1}{9} \right)^n \right) \end{aligned}$$

The second term, as  $N$  grows large will approach 0, so we can disregard and focus on the first term.

$$\begin{aligned}
 P_x &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=-N}^{-1} 1 \right) \\
 &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=1}^N 1 \right) \\
 &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} (N) \\
 &= \lim_{N \rightarrow \infty} \frac{N}{2N+1} \\
 &= \frac{1}{2}
 \end{aligned}$$

c. Determine the energy  $E_x$  and power  $P_x$  of  $x[n]$  if  $A = 2$ . The energy can be found as follows:

$$\begin{aligned}
 E_x &= \sum_{n=-\infty}^{\infty} |x[n]|^2 \\
 &= \sum_{n=-\infty}^{-1} 2^{2n} + \sum_{n=0}^{\infty} \left(\frac{1}{3}\right)^{2n} \\
 &= \infty
 \end{aligned}$$

The power can be found as follows:

$$\begin{aligned}
 P_x &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N |x[n]|^2 \\
 &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=-N}^{-1} 2^{2n} + \sum_{n=0}^N \left(\frac{1}{3}\right)^{2n} \right) \\
 &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=-N}^{-1} 2^{2n} \right) + \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=0}^N \left(\frac{1}{9}\right)^n \right)
 \end{aligned}$$

Similar to the previous part, the second term will approach 0 as  $N$  grows large, so we can disregard it.

$$\begin{aligned}
 P_x &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=-N}^{-1} 4^n \right) \\
 &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \sum_{n=1}^N 4^{-n} \right)
 \end{aligned}$$

We can use the following formula to find the sum of the series:

$$\sum_{n=1}^N 4^{-n} = \frac{1}{4} + \frac{1}{4^2} + \frac{1}{4^3} + \dots + \frac{1}{4^N} = \frac{1}{3} \left( 1 - \frac{1}{4^N} \right)$$

We can now substitute this back into the equation for  $P_x$ :

$$\begin{aligned} P_x &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \frac{1}{3} \left( 1 - \frac{1}{4^N} \right) \right) \\ &= \lim_{N \rightarrow \infty} \frac{1}{2N+1} \left( \frac{1}{3} - \frac{1}{3 \cdot 4^N} \right) \\ &= 0 \end{aligned}$$

### 3.3-3

### 3.6-1

We have the following difference equation:

$$\begin{aligned} y_0[-1] &= 3, \quad y_0[-2] = -1 \\ y[n] + \frac{1}{6}y[n-1] - \frac{1}{6}y[n-2] &= \frac{1}{3}x[n] + \frac{2}{3}x[n-2] \end{aligned}$$

The following code can be used to output the first 10 values of  $y[n]$ , assuming  $x[n] = u[n]$ :

```
1 import numpy as np
2
3 # Initialize y[n] array
4 y = np.zeros(10)
5
6 # Initialize x[n] array
7 x = np.ones(10)
```

Code Snippet 2: Python code to solve difference equation