Statistics and inferencing Activity 08

Name: Basil khowaja bk08432

Introduction

For this task, I wrote a MATLAB code to compute and plot the detection performance for different probabilities of false alarm (P_{FA}) and varying energy-to-noise ratio (ENR). The goal was to replicate the detection performance graph for a DC level in white Gaussian noise (WGN), as shown in the provided figure.

Explanation of the Code

MATLAB Code

The following MATLAB code was written to compute and plot the detection performance:

```
clc;
clear;
PFA = [1e-1, 1e-2, 1e-3, 1e-4, 1e-5, 1e-6, 1e-7]; % Specific PFA levels
ENR = linspace(0, 20, 100); % Energy-to-Noise Ratio (dB)
sigma2 = 1; % Noise variance

% Convert ENR to linear scale
ENR_linear = 10.^(ENR / 10);

% Initialize PD matrix
PD = zeros(length(PFA), length(ENR));

% Compute Q function
Q = @(x) 0.5 * erfc(x / sqrt(2));

% Calculate PD for each PFA and ENR
```

```
for i = 1:length(PFA)
    for j = 1:length(ENR)
        threshold = sqrt(sigma2) * norminv(1 - PFA(i));
        PD(i, j) = Q((threshold - sqrt(ENR_linear(j))) / sqrt(sigma2));
    end
end

% Plotting without color fill
figure;
hold on;
for i = 1:length(PFA)
    plot(ENR, PD(i, :), 'LineWidth', 1.5); % Plot each PFA as a separate line
    text(ENR(end) - 1, PD(i, end), sprintf('10^{%.0f}', log10(PFA(i))), 'Horizon
end
hold off;

grid on;
xlabel('Energy-to-noise-ratio (dB)');
ylabel('Probability of detection, P_D');
title('Detection performance for DC level in WGN');
axis([0 20 0 1]);
```

1. Parameter Definition

- P_{FA} was defined as a specific set of probabilities $(10^{-1}, 10^{-2}, \ldots, 10^{-7})$, matching the example plot.
- ENR (Energy-to-Noise Ratio) was varied between 0 and 20 dB using `linspace`, providing sufficient resolution for the plot.
- σ^2 was set to 1, representing the noise variance in the white Gaussian noise scenario.

2. ENR Conversion

Since energy-to-noise ratio in decibels (dB) is logarithmic, it was converted to a linear scale using the formula:

$$\text{ENR\_linear} = 10^{\text{ENR}/10}.$$

3. Probability of Detection Calculation

- The Q-function was defined as:

$$Q(x) = 0.5 \cdot \operatorname{erfc}\left(\frac{x}{\sqrt{2}}\right).$$

- For each combination of P_{FA} and ENR, the detection threshold was calculated using the inverse of the Q-function (`norminv` in MATLAB).
- The probability of detection (P_D) was determined using the formula:

$$P_D = Q\left(\frac{\text{threshold} - \sqrt{\text{ENR\_linear}}}{\sqrt{\sigma^2}}\right).$$

4. Plotting

- For each P_{FA}, P_D was plotted against ENR using `plot`, creating distinct curves for each P_{FA} level.
- Labels ($10^{-1}, 10^{-2}, \ldots, 10^{-7}$) were added at the end of each curve using `text` for clear identification.
- A grid was added for better visual clarity.

5. Axis Customization

- The x-axis was labeled as *Energy-to-noise-ratio (dB)*, and the y-axis was labeled as *Probability of detection, P_D*.
- Axis limits were set explicitly to $[0, 20]$ for ENR and $[0, 1]$ for P_D to align with the provided figure.

6. Final Presentation

The result is a clean plot with:

- Distinct curves for each P_{FA},
- Gridlines for better readability,
- Properly labeled axes and title.

This approach ensures the graph closely resembles the provided figure, capturing the relationship between P_{FA}, ENR, and P_D accurately.

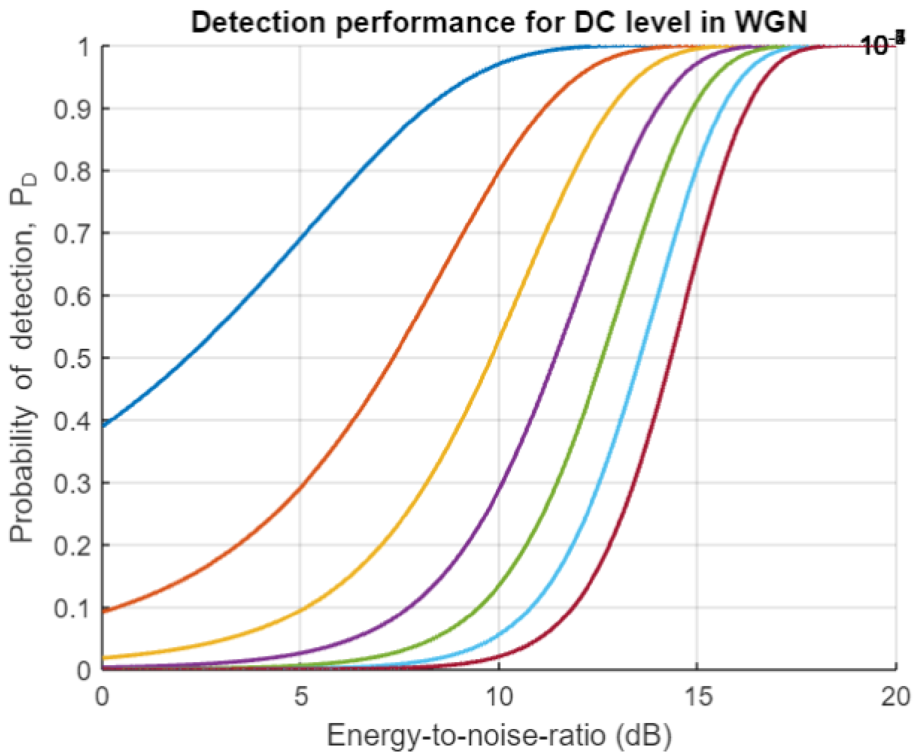

Figure 1: Generated Plot of Detection Performance for DC Level in WGN

Conclusion

The MATLAB code successfully computes and plots the detection performance for a DC level in WGN, producing a graph that matches the reference figure. The distinct curves for P_{FA} values and clean labeling make the visualization intuitive and informative.

Task 2: Determining the Necessary Number of Samples N

Given:

- Probability of false alarm $P_{FA} = 10^{-4}$
- Probability of detection $P_D = 0.99$
- Signal-to-noise ratio $10 \log_{10}\left(\frac{A^2}{\sigma^2}\right) = -32\,\text{dB}$

Solution:

1. **Converting SNR from dB to linear scale:**

$$\text{SNR (linear)} = 10^{\frac{-32}{10}} = 10^{-3.2} \approx 0.000631$$

2. **Calculating the inverse Q-function values:** Using standard tables or approximations:

$$Q^{-1}(P_{FA}) \approx 3.719, \quad Q^{-1}(1 - P_D) \approx -2.326$$

3. **Use the relationship:**

$$\sqrt{\frac{NA^2}{\sigma^2}} = Q^{-1}(P_{FA}) - Q^{-1}(P_D)$$

Substituting values:

$$\sqrt{\frac{NA^2}{\sigma^2}} = 3.719 - (-2.326) = 3.719 + 2.326 = 6.045$$

Since:

$$\frac{A^2}{\sigma^2} = \text{SNR (linear)} = 0.000631$$

We have:

$$\sqrt{N \cdot 0.000631} = 6.045$$

Squaring both sides:

$$N \cdot 0.000631 = 6.045^2 = 36.54$$

Solving for N:

$$N = \frac{36.54}{0.000631} \approx 57,926$$

Final Answer: The necessary number of samples N is approximately **57,926**.