**EE 121 – Introduction to Digital Communication Systems** Spring 2016

# Homework 1

Assigned: Tue, 1/26/16 Due: Tue, 2/2/16

*Instructor: Omur Ozel* *GSI: Preetum Nakkiran*

Notation: $\vec{X} \sim \mathcal{N}(\vec{\mu}, \Sigma)$ denotes that random vector $\vec{X}$ has Gaussian distribution with mean $\vec{\mu}$ and covariance matrix $\Sigma$. Recall that the density of the $d$-dimensional Gaussian is:

$$f(\vec{x}) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp\left(-\frac{1}{2}(\vec{x}-\vec{\mu})^{\mathrm{T}}\Sigma^{-1}(\vec{x}-\vec{\mu})\right)$$

The $Q$ function is the tail probability of a normalized Gaussian density:

$$Q(x) = \int_x^\infty f(\tau)d\tau$$

where $x \in \mathbb{R}$ and $f$ is the pdf of the standard normal $\mathcal{N}(0,1)$. The following tail bound is useful:

$$Q(x) \le e^{-x^2/2}$$

## 1 MAP for Vector Channel

In this problem, we will find the MAP decision rule for a vector channel with vector Gaussian noise. In this channel, the transmitter sends a vector $\vec{x} \in \mathbb{R}^d$, which is corrupted by noise $\vec{n}$, and received as $\vec{y} = \vec{x} + \vec{n}$. The noise $\vec{n} \in \mathbb{R}^d$ is distributed $\vec{n} \sim \mathcal{N}(0, \Sigma)$.

The encoder sends a message $m \in \{0,1\}$ by transmitting either $\vec{\mu}_0$ or $\vec{\mu}_1$ across the channel (for some $\vec{\mu}_0, \vec{\mu}_1 \in \mathbb{R}^d$). Suppose the encoder picks $m \in \{0,1\}$ uniformly at random. We wish to determine the MAP decoder for this channel.

(a) For simplicity, suppose $\Sigma = I$, that is, each coordinate of $\vec{n}$ is distributed independently as $n_i \sim \mathcal{N}(0,1)$. Further, suppose $d = 2$ and $\vec{\mu}_0 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$ and $\vec{\mu}_1 = -\vec{\mu}_0 = \begin{bmatrix} +1 \\ +1 \end{bmatrix}$. Derive the MAP decision rule for the decoder (for picking $\hat{m}$, given the received $\vec{y}$). Plot the region of $\mathbb{R}^2$ for which the decoder decides $\hat{m} = 0$, and the region it decides $\hat{m} = 1$. Recall that for $d = 1$, the MAP decision rule was a *threshold test*, and these decision regions were half-spaces. Comment on the case $d = 2$.

(b) Relax the assumptions made in the previous part. That is, derive the MAP decoder for general $d \ge 1$, covariance matrix $\Sigma$, and $\vec{\mu}_0, \vec{\mu}_1$. Is the decision rule still a linear threshold test? *(Hint: This part should be straightforward, using the same algebra as in the previous part)*

(c) Let us consider if the message $m$ was *not* chosen to be 0 or 1 with equal probabilities. If the prior probabilities are different, is the decision rule still a linear threshold test?

(d) Consider a similar setup as part (a): $d = 2$, and each noise coordinate distributed independently $n_i \sim \mathcal{N}(0, N/2)$ for some $N \in \mathbb{R}$. Let $\vec{\mu}_0 = \begin{bmatrix} -\sqrt{P/2} \\ -\sqrt{P/2} \end{bmatrix}$ and $\vec{\mu}_1 = \begin{bmatrix} +\sqrt{P/2} \\ +\sqrt{P/2} \end{bmatrix}$. Let the message $m \in \{0,1\}$ be chosen uniformly. Notice that under this setup, the *expected noise energy* is $\mathbb{E}[||\vec{n}||^2] = N$, and the *expected signal energy* is $\mathbb{E}[||\vec{x}||^2] = P$.

Just as in the scalar case ($d = 1$), we want to show that we can drive the probability of decoding error to zero by increasing the *signal-to-noise ratio* $\frac{P}{N}$. Show that this is the case. In particular, show that the probability of decoding error (using MAP decoding), goes to zero exponentially fast in $\frac{P}{N}$.

*(Hint: Consider reducing the problem to the scalar case.)*

## 2 MAP for Audio Communication

We showed in the discussion that MAP decoding is optimal (minimizing the probability of decoding error). We also considered an audio communication system using OOK (on-off-keying), where the sender sends a message $\{0, 1\}$ by either transmitting a tone, or no tone. The transmitted waveform is corrupted by Guassian noise, and the receiver decodes by taking an inner-product with the tone waveform (equivalently, by looking at the Fourier coefficient corresponding to the tone).

In the setting of the previous problem, we can think of this audio system as communicating across a vector channel, where $\vec{\mu}_0 = \vec{0}$ and $\vec{\mu}_1 = \vec{s}$ is the [sampled] tone waveform (for some fixed number of samples $d$). Suppose that each coordinate of noise is distributed independently $n_i \sim \mathcal{N}(0, \sigma^2)$.

Recall that our candidate decoder for this channel was:

(i) Given the received vector $\vec{y}$, compute the inner-product $z := \langle \vec{y}, \vec{s} \rangle$. Notice that if $m = 0$, then $z = \langle \vec{n}, \vec{s} \rangle \sim \mathcal{N}(0, \sigma^2 ||s||^2)$. If $m = 1$, then $z \sim \mathcal{N}(||s||^2, \sigma^2 ||s||^2)$.

(ii) Perform a threshold test on the scalar $z$: If $z \geq \frac{1}{2}||\vec{s}||^2$, decode to $\hat{m} = 1$, otherwise decode to $\hat{m} = 0$.

**Is this decoder optimal?** Is it equivalent to the MAP decoder? *(Hint: Use the results of the previous problem).*

## 3 Additive Noise Channel with Fading

In this problem, we will calculate error probability for a scalar fading additive Gaussian noise channel. In particular, the channel input and output, $X$ and $Y$, are related as:

$$Y = AX + N$$

where N is a Gaussian random variable with mean 0 and unit variance. And $A \in \{0.5, 1\}$ is a binary random variable. Assume that the encoder sends a message $m \in \{0, 1\}$ by transmitting either $X = -\sqrt{P}$ or $X = \sqrt{P}$ across the channel with equal probability. The receiver observes Y and makes a decision about the input.

(a) First assume that $A = 1$ with probability one. What is the best decision rule for the receiver (i.e., the rule that minimizes the probability of error)? Now, assume $A = 0.5$ with probability one. What is the best decision rule for the receiver?

(b) What is the best decision rule when $\Pr(A = 0.5) = \Pr(A = 1) = \frac{1}{2}$? Compare it with the rules you found in the previous part.

(c) Determine the probability of error in terms of $Q$ function when (i) $A = 1$ with probability one and (ii) $\Pr(A = 0.5) = \Pr(A = 1) = \frac{1}{2}$.

(d) Plot the variation of the probability of error in the cases (i) and (ii) in part-c with respect to $P$. Comment on the plots.

## 4 Photon Counting and Direct Detection Channels

In this problem, we will consider two channels listed in the title of the question. In the photon counting channel, the channel input and output, $X$ and $Y$, are both integer valued and they are related through the following conditional probability distribution:

$$p(y|x) = \frac{x^y}{y!} e^{-x}, \ y \in \mathbb{N}$$

The direct detection channel is an additive noise channel with input dependent noise. The channel input $X$ and channel output $Y$ are both real valued and $X$ is nonnegative. The input-output relation is:

$$Y = X + N$$

where $N$ is a zero mean Gaussian random variable and the variance of $N$ is $1 + X$, i.e., the variance of the noise depends on the input $X$. Assume that the encoder sends a message $m \in \{0, 1\}$ by transmitting either $X = 0$ or $X = \sqrt{P}$ across the channel with equal probability. Note that in the photon counting channel, $\sqrt{P}$ is an integer whereas in the direct detection channel, it could be any real number.

(a) Determine the best decision rule for the photon counting channel. Obtain an expression for the probability of error.

(b) Determine the best decision rule for the direct detection channel. Obtain an expression for the probability of error.

(c) Plot the probability of error for both cases with respect to increasing $P$. Comment on the plots.

## 5 Project Warmup (optional)

This question gives some ideas for getting started on your project[1]. Note that this question is optional and you are given two weeks to complete the tasks.

(a) Write a computer program on your laptop that encodes a binary string into an sequence of tones using OOK. That is, a 1 should correspond to a 440Hz tone and a 0 should be silence. For each bit, the duration of the tone should be a quarter of a second.

(b) Write a computer program that takes samples from your microphone, divides them into quarter-second duration chunks, and then decodes each chunk separately into 0s and 1s.

(c) For testing purposes, first use a "virtual noiseless channel" that is perfectly synchronized. That is, assume that both encoder and decoder agree on exactly when to start transmitting/receiving. (Don't even go through the speaker and microphone, just simulate what would happen if you did it perfectly.) Verify that there are no errors.

(d) Test this in a virtual channel that is perfectly synchronized, but you add iid noise. How does the probability of error change with the noise power? Plot it. Try varying the duration of the tones, and the volume of the transmission. Do the error probabilities scale as you expect?

(e) Try to run the encoder/decoder in real life, either on one laptop (transmitting and receiving simultaneously on the same machine), or between two laptops. For simplicity at this point, use whatever method of synchronization you like (eg, internet-synchronized clocks, or LAN ping, etc.). How does it perform? What problems do you face?

(f) Design a test to estimate the frequency response of the audio channel. What is the usable bandwidth? Try this out in different rooms/locations. In the process, note any further non-idealities of the physical layer (are echos a problem? do you notice nonlinearities in your speakers? etc)

[1]Inspired and borrowed from EE121, Fall 2013 with Anant Sahai and K.V. Rashmi.