Inference Examples

In the remaining lectures, we consider applications that are centered around inference. In inference problems, we update probabilities of an event by conditioning on more information. We usually observe an output and want to infer the corresponding input, where the input and output are related probabilistically. An inference example we already covered is inferring whether or not a patient has a disease based on the result of a medical diagnostic. We will focus on three examples in: communications, speech recognition and tracking. We will cover the communication problem today.

Example 1: Communications

We want to send information through a medium (e.g. cable modem, the internet etc). The situation can be modeled by probabilistically. Let X be the input bit, which can be a 0 or 1. The bit goes through a communication channel that introduces some noise. The aim is to design a receiver which can infer X upon observing Y, as well as to evaluate the performance of our receiver.

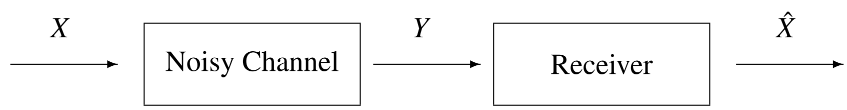

Figure 1: *A communication channel*

One of the simplest channels is the flip channel (or binary symmetric channel) which is shown in Figure 2. In that channel, a 0 input goes to a 0 output with probability $1-p$, and to a 1 with probability p (so flips occur with probability p). Assume without loss of generality that $p < \frac{1}{2}$. The probabilities that appear on the figure are conditional probabilities, such as $\mathbf{P}(Y=0|X=0) = 1-p$. Equivalently, we can represent the channel as:

$$Y = (X+Z) \mod 2,$$

where Z is 0 with probability $1-p$ and 1 with probability p, and X and Z are independent.

To complete the model of the problem, we also assume that we have some probabilities on the input values. Let $\mathbf{P}(X=0) = \alpha$, and $\mathbf{P}(X=1) = 1-\alpha$ (our prior probabilities). These probabilities are analogous to the fraction of people that have a disease in a population.

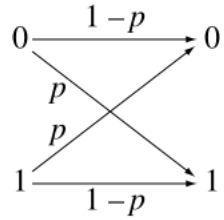

Figure 2: *Binary Symmetric Channel.*

We now want to design our receiver. We observe $Y = b$, where $b \in \{0,1\}$. What shall we guess for X? Conditioned on the event we observe $Y = b$, how do we decide what X is? We look at the posterior probabilities: $\mathbf{P}(X = 0|Y = b)$ and $\mathbf{P}(X = 1|Y = b)$. A reasonable rule would be to set our estimate $\hat{X}$ to the value that has the largest posterior probability. Therefore, the receiver computes $\mathbf{P}(X = 0|Y = b)$ and $\mathbf{P}(X = 1|Y = b)$, and using the following decoding rule:

$$\hat{X} = \begin{cases} 0 & \mathbf{P}(\text{X=0|Y=b}) > \mathbf{P}(\text{X=1|Y=b}), \\ 1 & \text{otherwise.} \end{cases}$$

These probabilities can be computed using Bayes' rule:

$$\mathbf{P}(X = 0|Y = b) = \frac{\mathbf{P}(Y = b|X = 0)\mathbf{P}(X = 0)}{\mathbf{P}(Y = b)}, \tag{1}$$

$$\mathbf{P}(X = 1|Y = b) = \frac{\mathbf{P}(Y = b|X = 1)\mathbf{P}(X = 1)}{\mathbf{P}(Y = b)}. \tag{2}$$

However since $\mathbf{P}(X = 0|Y = b)$ and $\mathbf{P}(X = 1|Y = b)$ have the same denominator $\mathbf{P}(Y = b)$, we only need to compute and compare the numerators. Therefore, our receiver rule can be simplified to comparing

$$\mathbf{P}(Y = b|X = 0)\mathbf{P}(X = 0) \underset{\hat{X}=1}{\overset{\hat{X}=0}{\gtrless}} \mathbf{P}(Y = b|X = 1)\mathbf{P}(X = 1)$$

Flipping this equation around we obtain:

$$L(b) := \frac{\mathbf{P}(Y = b|X = 0)}{\mathbf{P}(Y = b|X = 1)} \underset{\hat{X}=1}{\overset{\hat{X}=0}{\gtrless}} \frac{\mathbf{P}(X = 1)}{\mathbf{P}(X = 0)} = \frac{1-\alpha}{\alpha}.$$

The term $\frac{1-\alpha}{\alpha}$ is a constant threshold, and the left hand side term is called the likelihood ratio $L(b)$, which is a function of the observation value b. The receiver compares the likelihood to the threshold that depends on the input priors. If the threshold is very large (meaning that 1 is much more likely to be the input than 0), then we only decode to 1 if our likelihood is very large as well. Let us now compute the likelihood for both values of b.

$$L(0) = \frac{\mathbf{P}(Y = 0|X = 0)}{\mathbf{P}(Y = 0|X = 1)} = \frac{1-p}{p}$$

and

$$L(1) = \frac{\mathbf{P}(Y = 1|X = 0)}{\mathbf{P}(Y = 1|X = 1)} = \frac{p}{1-p}.$$

At the receiver, we compute $L(0)$ if a 0 was received, and $L(1)$ otherwise. We compare the values obtained to the threshold and infer $\hat{X}$.

How can we improve the performance of our channel? We can get better hardware which will give a cleaner channel with a smaller p. Alternatively, we can process X before sending it through the channel. Instead of sending the information bit directly, we can consider encoding X, as shown in Figure 3.

Figure 3: *A communication channel with an encoder.*

This code is supposed to help us improve the reliability of the system. What is the simplest possible code we can think of? We could send each bit multiple times. This is called a repetition code and is illustrated in Figure 4. The corresponding output will be n random variables $Y_1,\ldots,Y_n$.

Figure 4: *A communication channel with a repetition encoder, that sends the same input n times.*

We hope to get a better performance out of this scheme. How do we model the relationship between X and all the Y_i variables? A natural assumption is that each Y_i is the output of a binary symmetric channel with X as the input, and the Y_i's are mutually independent conditional on the input. Equivalently,

$$Y_i = (X + Z_i) \mod 2$$

where $Z_i = 1$ with probability p and $Z_i = 0$ otherwise and the Z_i's are mutually independent and independent of X.

We now have our model. How do we make our decision in this new setting? Before, we computed the posterior probabilities of the form $\mathbf{P}(X=0|Y=b)$ and $\mathbf{P}(X=1|Y=b)$ and compared them. Which posterior probabilities should we consider in this case? Since we now observe n channel outputs, we should now condition on all their values, and the rule is therefore:

$$\mathbf{P}(X=0|Y_1=b_1,\cdots,Y_n=b_n) \underset{\hat{X}=1}{\overset{\hat{X}=0}{\gtrless}} \mathbf{P}(X=1|Y_1=b_1,\cdots,Y_n=b_n).$$

We apply the same logic, use Bayes' rule and cancel the common denominator. The decoder rule is equivalent to :

$L(b_1\cdots b_n) := \frac{\mathbf{P}(Y_1=b_1,\cdots,Y_n=b_n|X=0)}{\mathbf{P}(Y_1=b_1,\cdots,Y_n=b_n|X=1)} \underset{\hat{X}=1}{\overset{\hat{X}=0}{\gtrless}} \frac{1-\alpha}{\alpha}$.

This can be easily simplified because the $Y_1,\ldots,Y_n$ are conditionally independent given the input. The overall likelihood can be decomposed as the product of the likelihoods of the individual received symbols:

$$L(b_1,\cdots,b_n) = \frac{\mathbf{P}(Y_1=b_1|X=0)\cdots\mathbf{P}(Y_n=b_n|X=0)}{\mathbf{P}(Y_1=b_1|X=1)\cdots\mathbf{P}(Y_n=b_n|X=1)} = L(b_1)L(b_2)\cdots L(b_n).$$

We want to convert this into an addition which is aesthetically nicer as it corresponds to aggregating information from each of the received output. So let as define the log-likelihood ratio as:

$$\mathrm{LLR}(b) := \log L(b) = \begin{cases} \log(\frac{1-p}{p}) & \text{if } b_i = 0, \\ -\log\left(\frac{1-p}{p}\right) & \text{if } b_i = 1. \end{cases},$$

and

$$\mathrm{LLR}(b_1,\cdots,b_n) := \log(L(b_1\cdots b_n)) = \sum_{i=1}^{n}\mathrm{LLR}(b_i) \underset{\hat{X}=1}{\overset{\hat{X}=0}{\gtrless}} \log\left(\frac{1-\alpha}{\alpha}\right).$$

The sum of the log likelihood ratios above can just be expressed in terms of 2 random variables, the number of 0s and the number of 1s that are observed at the output sequence. Let us define: U as the number of 0s received, and V as the number of 1s received. Then

$$\sum_i \mathrm{LLR}(b_i) = U\times\log\left(\frac{1-p}{p}\right) + V\times\left(-\log\left(\frac{1-p}{p}\right)\right).$$

The rule becomes

$$U - V \underset{\hat{X}=1}{\overset{\hat{X}=0}{\gtrless}} \frac{\log\left(\frac{1-\alpha}{\alpha}\right)}{\log\left(\frac{1-p}{p}\right)}.$$

The rule simplifies even further when $\alpha = \frac{1}{2}$, meaning when we are equally likely to transmit a 0 or 1. When $\alpha = \frac{1}{2}$, the threshold becomes equal to $\log(1) = 0$. So the rule becomes:

$$U \underset{\hat{X}=1}{\overset{\hat{X}=0}{\gtrless}} V,$$

which is the majority rule.

Now suppose we have a target performance for the communication system we are designing. We want to make sure that our error probability is less than 0.1%. How many times do we need to repeat each bit in order to achieve that? Let's analyze this in the case where $\alpha = \frac{1}{2}$. What is the error probability $p_e := \mathbf{P}(\hat{X} \neq X)$ of our rule when $\alpha = \frac{1}{2}$ in terms of U and V? There are two ways in which an error can happen: if 1 is transmitted and $\hat{X} = 0$, or if a 0 is transmitted and $\hat{X} = 1$.

$$\begin{aligned} p_e &= \mathbf{P}(X \neq \hat{X}) \\ &= \mathbf{P}(U > V|X=1)\mathbf{P}(X=1) + \mathbf{P}(U < V|X=0)\mathbf{P}(X=0) \\ &= \mathbf{P}(U < V|X=0) \end{aligned}$$

The receiver makes a mistake when more than half of the bits get flipped. This corresponds to the event that more than half of the Z_is are equal to 1:

$$\mathbf{P}(U < V | X = 0) = \mathbf{P}(\sum_{i=1}^{n} Z_i > \frac{n}{2}).$$

We now have a sum of random variables: $S_n := \sum_{i=1}^{n} Z_i \sim Bin(n, p)$. We want to compute $\mathbf{P}(S_n > \frac{n}{2})$ This can be computed numerically with the binomial distribution. The central limit theorem can also be used to approximate this summation. We know $\mathbb{E}[S_n] = np$ and $\text{Var}(S_n) = np(1-p)$. Therefore,

$$\mathbf{P}(S_n > \frac{n}{2}) = \mathbf{P}\left(\frac{S_n - np}{\sqrt{np(1-p)}} > \frac{\frac{n}{2} - np}{\sqrt{np(1-p)}}\right) = Q\left(\frac{n/2 - np}{\sqrt{np(1-p)}}\right) = Q\left(\frac{0.5 - p}{\sqrt{p(1-p)}} \cdot \sqrt{n}\right),$$

where $Q(x)$ is the probability that a $N(0, 1)$ random variable exceeds x. Note that this error probability decreases with n. We simply choose n so that

$$Q\left(\frac{0.5 - p}{\sqrt{p(1-p)}} \cdot \sqrt{n}\right) < 0.001.$$